උසස් පෙළ

තොරතුරු හා සන්නිවේදන තාක්ෂණය

ICT

දෙවන කොටස

උසස් පෙළ විෂය නිර්දේශයේ
නවවන ඒකකයේ සිට දහතුන්වන ඒකකය දක්වා
විෂය කරුණු
මෙම ග්‍රන්ථයේ ඇතුලත් වේ.

ICT ගුරුවරුන් පමණයි Facebook සමූහයේ ප්‍රකාශනයකි

උසස් පෙළ
තොරතුරු හා සන්නිවේදන තාක්ෂණය
දෙවන කොටස

2017 අපොස උසස් පෙළ විෂය නිර්දේශයට අනුකූලව සැකසී ඇත

ICT ගුරුවරුන් පමණයි Facebook සමූහයේ ප්‍රකාශනයකි

කර්තෘ මණ්ඩලය

ඒකකය 9 - ක්‍රමලේඛකරණය

අරුණ හේරත්

Bsc (BIT -University of Colombo), DICS,DISE,DIM,DiTEC,DIW

ඒකකය 10 - වෙබ් අඩවි සංවර්ධනය

සුනාලි නිෂිතා ප්‍රනාන්දු

BIT, Dip in Teaching ICT

අංජන කුමාර

Dip in Teaching ICT

තණුජ බණ්ඩාර

BSc.

සමිත දිල්ශාන්

B.Sc(BIT)SP, MIE, DIP IN WEB, YOUTUBER, Government School ICT Teacher

ඒකකය 11 - සාර්ව අන්තර්ජාලය

කිෂාන් අජන්ත රාජමන්ත්‍රී

BCS,CCNA,Dip in ICT, Writer in Wijaya Pariganaka Magazine, Author , Professional ICT Teacher

ඒකකය 12 - ව්‍යාපාර තුළ තොරතුරු සන්නිවේදන තාක්ෂණය

සුනාලි නිෂිතා ප්‍රනාන්දු

BIT, Dip in Teaching ICT

ඒකකය 13 - තොරතුරු සන්නිවේදන තාක්ෂණයේ නව නැඹුරු හා අනාගත දිශානති

මහේෂ් කොඩිතුවක්කු

BTech(ICT) MCTS,MIEEE,Cert in Digital Marketing(Google),Cert in ICT(AIT-Thailand)
National Dip in Teaching (Merit)

පෙරවදන

උසස් පෙළ තොරතුරු හා සන්නිවේදන විෂය සඳහා පරිපූර්ණ ග්‍රන්ථයක් නොමැති වීම එම විෂය ඉගන්වන ගුරුවරන්ට මෙන්ම විෂය හදාරණ සිසුන්ට ද ඇති විශාල ගැටලුවකි. ගුරු අත්පොත් වල ඇති ඇතැම් දුරාබෝධ තැන් මෙන්ම සවිස්තරාත්මකව දක්වා නොමැති කොටස් මෙම ගැටලුව තවදුරටත් තීව්‍ර කරයි. විෂයක් ජනප්‍රිය කරවීම සඳහා ඒ ආශ්‍රිත ප්‍රකාශන ද සුලබ විය යුතුය. මෙම උක්ත කරුණු සැලකිල්ලට ගනිමින් පොතක් සැකසීම පිළිබඳ ව අදහස අප ICT ගුරුවරුන් සඳහා ෆේස්බුක් සමූහයෙහි විටින් විට පල විය. කෙසේ නමුත් මේ අදහස වරින් වර මතු වෙමින් නැවත යටපත් වෙමින් පැවති අතර ඒ අරමුණ යථාර්ථයක් කිරීමට ගත් උත්සාහයක් ලෙස මෙම ග්‍රන්ථය සැලකිය හැක.

මෙය කිසිදු ආර්ථික අරමුණකින් තොරව අප විසින් ගන්නා ලද සාමූහික ප්‍රයත්නයකි. මෙම කාර්යය සාක්ෂාත් කර ගැනීම සඳහා සහය වූ අප ගුරුමහත්ම මහත්මීන් ට නොමඳ ස්තුතිය හිමිවේ.
මුල දී ඒකක 13 සඳහා ම පොත සැකසිය යුතුය යන අදහසෙහි අප සිටියත් ආරම්භයක් ලෙස 1 සිට 8 දක්වා වූ ඒකක එක් කොට මෙහි පළමුවන පොත නිර්මාණය කළෙමු. මේ එම කෘතියෙහි දෙවැනි කොටසයි. ඒ තුළින් ඉතිරි ඒකක 5 සම්පූර්ණයෙන් ආවරණය කර තිබේ.

මෙම ග්‍රන්ථයෙහි දක්වා ඇති එක් එක් ඒකක ගුරුභවතෙකු හෝ කිහිපදෙනෙකු එක් ව ලියන ලද ඒවා බැවින් ඒ අයගේ කර්තෘ අයිතිය ආරක්ෂා කරන මෙන් මෙම කෘතිය පරිශීලනය කරන අප සහෘදයන්ගෙන් මෙන්ම ආදරණීය දරුවන්ගෙන් ද කාරුණිකව ඉල්ලමි.

යම් නිශ්චිත කාලපරාසයක් තුළ කළ නිර්මාණයක් බැවින් මෙහි යම් අඩුපාඩුවක් හෝ එක් විය යුතු යමක් තිබේ නම් කරුණාකර සංස්කාරක මණ්ඩලය විද්‍යුත් තැපෑල හරහා අමතන මෙන් ඉල්ලමි.

සංස්කාරක මණ්ඩලය
kkmaheshkodit@gmail.com

පටුන

ක්‍රමලේඛකරණය	01
වෙබ් අඩවි සංවර්ධනය	63
සාර්ව අන්තර්ජාලය	121
ව්‍යාපාර තුළ තොරතුරු සන්නිවේදන තාක්ෂණය	137
තොරතුරු සන්නිවේදන තාක්ෂණයේ නව නැඹුරු හා අනාගත දිශානති	142

පිටුම

තොරතුරු හා සන්නිවේදන තාක්ෂණය විෂය හදාරණ මෙන්ම
එහි නවමං සොයමින් ඉදිරියට යන සැමට පිදෙයි

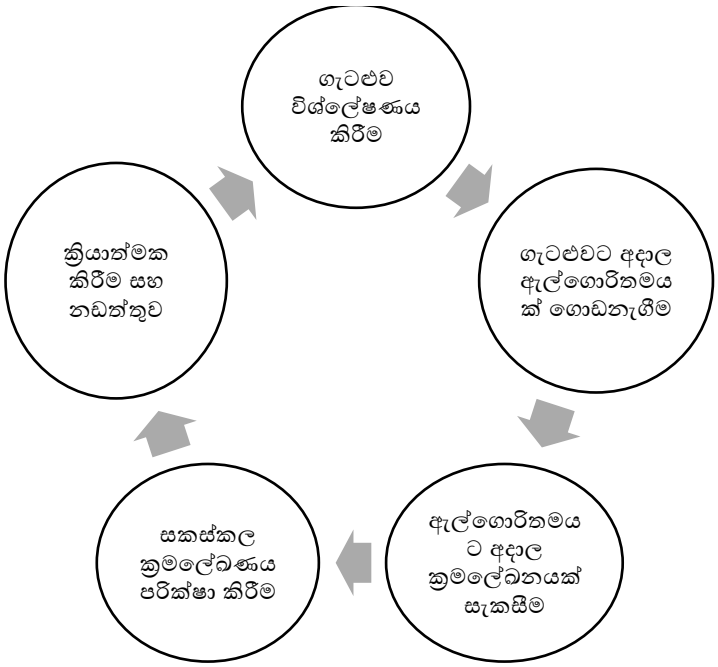
නිපුණතාව 9 ගැටලු විසඳීමට ඇල්ගොරිතම සංවර්ධනය කර ඒවා ආකේතනය කිරීම සඳහා පයිතන් ක්‍රමලේඛ භාෂා භාවිතා කිරීම.

ගැටලු විසඳීමේ ක්‍රියාවලිය භාවිතා කිරීම.

පරිගණක ක්‍රමලේඛනයක් යනු කුමක්ද ? (What is a Computer Program?)

කිසියම් කාර්යයක් පරිගණකයෙන් කර ගැනීම සඳහා පරිගණක භාෂාවක් අනුසාරයෙන් ගොනුකල ප්‍රකාශනයන්ගේ එකතුවක් පරිගණක ක්‍රමලේඛයක් (Computer Program) ලෙස හැඳින්වේ.

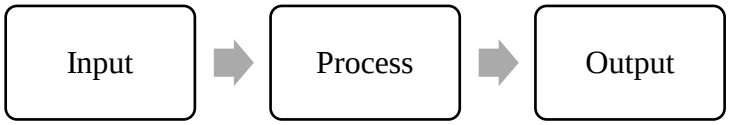
ක්‍රමලේඛන සංවර්ධන ජීවන චක්‍රය (Program Development Life-Cycle)



ගැටළුවක් විසඳීමේ අදියර (Analyzing the problem)

ගැටළුව විශ්ලේෂණය කිරීම

ගැටළුවක් විශ්ලේෂනය කිරීම යනු ගැටළුවට අදාල ආදාන, ප්‍රතිදාන සහ ක්‍රියාවලිය හඳුනාගැනීමයි.



විකල්ප විසඳුම

ගැටලුවක් විසඳීම සඳහා ක්‍රමවේද කිහිපයක් තිබිය හැක. ඒවා විකල්ප විදුම් ලෙස හඳුන්වයි. මෙය ගැටලුවේ ස්වභාවය අනුව වෙනස් වේ.

උදාහරණ: සෘජුකෝණාස්‍රයක පරිමිතිය සෙවීම සඳහා පහ ක්‍රමවේද භාවිතා කළ හැක.

පරිමිතිය=දිග+දිග+පළල+පළල

පරිමිතිය=**2**xදිග **+2**xපළල

පරිමිතිය=(දිග+ පළල) x **2**

ගැටලු විසඳීම සඳහා ඇල්ගොරිතමික ප්‍රවේශය යොදාගැනීම.

ඇල්ගොරිතමක් යනු කුමක්ද? **(What is an algorithm)**

ඇල්ගොරිතමක් යනු ගැටළුවක් විසඳීම සඳහා අනුගමනය කරන ක්‍රමවේදය අනුපිළිවෙලින් ලියා දැක්වීමයි.

මෙහිදී ගැටළුව විසඳීම සඳහා අනුගමනය කරන ක්‍රමවේදය අනුපිලිවෙලින් ලියා දක්වයි. ඇල්ගොරිතම ගොඩනැඟීම රූපමය ආකාරයෙන් (Graphically) හෝ ලිඛිත ආකාරයෙන් (Textual) දැක්විය හැක. රූපමය ආකාරයෙන් දැක්වීමට ගැලීම් සටහන් ද ලිඛිත ආකාරයෙන් දැක්වීමට ව්‍යාජ කේතද භාවිතා කල හැක.

ක්‍රමලේඛ ලිවීම **(Writing a Computer Program)**

ඇල්ගොරිතම පරිගණක භාෂාවක් ආධාරයෙන් ලියා දැක්වීම මෙහිදී සිදුකරයි. මෙය “Coding” ලෙස හඳුන්වයි.

1. Java
2. C#
3. VB
4. Python

Computer Languages

ක්‍රමලේඛ ක්‍රියාත්මක කර බැලීම **(Testing a Computer Program)**

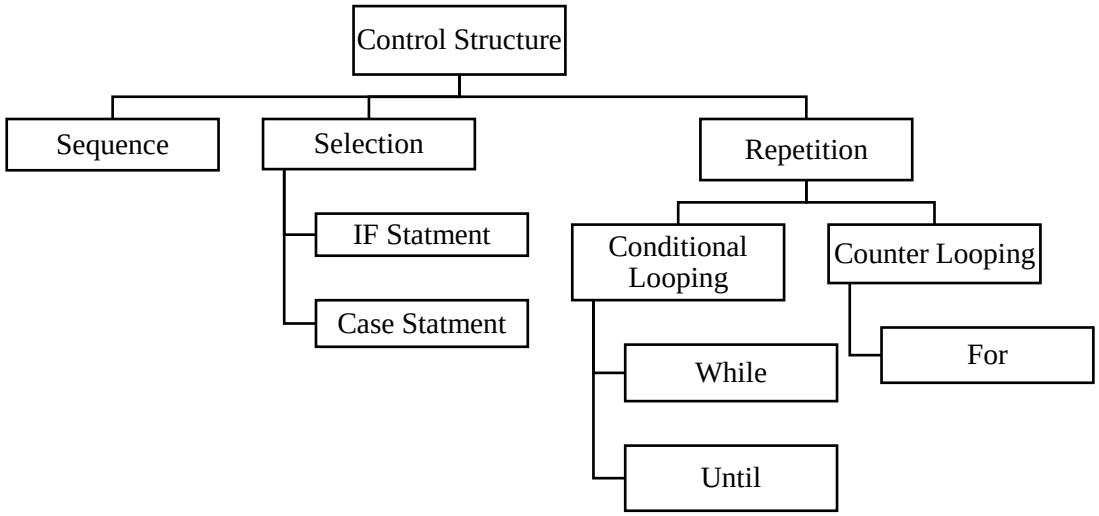
ලියන ලද ක්‍රමලේඛය නිවැරදි දැයි මෙහිදී පරීක්ෂා කරනු ලබයි. වැරදි (Bugs) තිබෙනම් ඒවා “ Programmer” විසින් නිවැරදි කරනු ලැබේ. මෙය “debugging ” ලෙස හඳුන්වයි.

ක්‍රමලේඛ අදාල දත්ත මත ක්‍රියාත්මක කිරීම **(Execute the program on the input data)**

මෙහිදී නිදොස්කල ක්‍රමලේඛ සත්‍ය ආදාන මත ක්‍රියාකර ප්‍රතිදානය ලබාගනු ලැබේ.

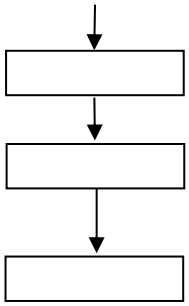
පාලන ව්‍යුහය **(Control Structure)**

පරිගණක ක්‍රමලේඛ තුල ඇති වගන්ති ක්‍රියාත්මක කරන අනුපිළිවෙල පාලන ව්‍යුහයක් මගින් පාලනය කරනු ලැබේ. ප්‍රධාන පාලන ව්‍යුහ තුනක් ඇත. අනුක්‍රමය, වරණය සහ පුණරාවර්තනය එම පාලන ව්‍යුහ වේ.



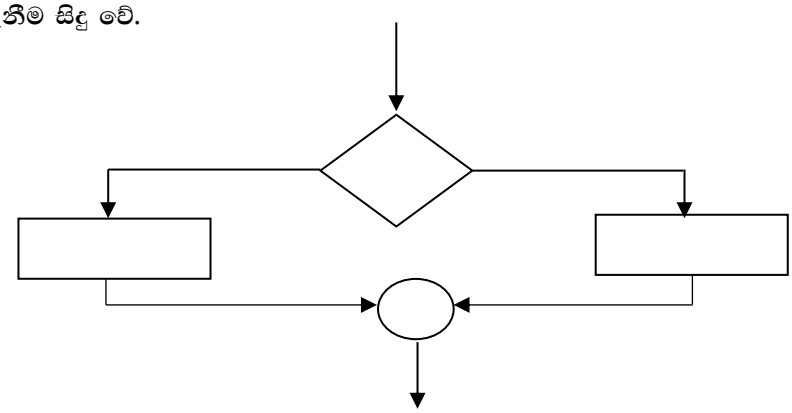
අනුක්‍රමය **(Sequence)**

ක්‍රමලේඛයක ඇති වගන්ති එකිනෙක අනුපිළිවෙලින් දී ඇති ආකාරයට ක්‍රියාත්මක කිරීම අනුක්‍රමය නමින් හැඳින්වේ.



වරණය **(Selection)**

මෙහිදී තෝරාගැනීම සඳහා විකල්ප ප්‍රකාශන සමූහයක් ඇතිවිට දෙනලද කොන්දේසියක් පරීක්ෂා කර එක් විකල්පයක් පමණක් තෝරාගැනීම සිදු වේ.

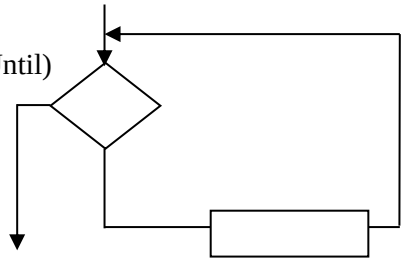


Conditional
(IF – else)
(Switch – Case)

පුනර්කරණය **(Repetition / Iteration / Looping)**

වගන්ති එකක් හෝ කිහිපයක් නැවත නැවත ක්‍රියාත්මක කිරීම පුනර්කරණය ලෙස හඳුන්වයි.

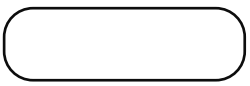
Loop (While, For, Repeat- Until)



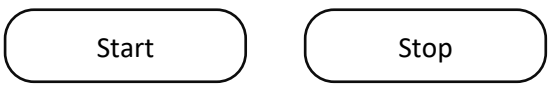
ගැලීම් සටහන් **(Flow Chart)**

ක්‍රමලේඛයක ක්‍රියාකාරීත්වය අනුපිළිවෙලින් දක්වෙන රූපමය නිරූපණය ගැලීම් සටහනක් ලෙස හඳුන්වයි. ගැලීම් සටහනක දත්ත ගලායන ආකාරයෙන් ඒවා යම් යම් ක්‍රියාකාරකම් වලට භාජනය වන ආකාරයක් පෙන්නුම් කරයි.

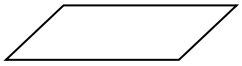
1. ආරම්භ හෝ අවසානය - Start / Stop (end)



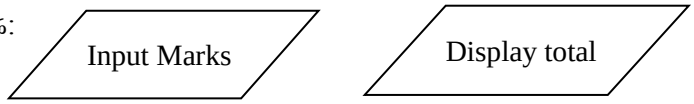
උදාහරණ:

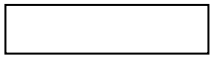


2. ආදානය හෝ ප්‍රතිදානය - Input / Output

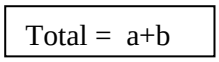


උදාහරණ:

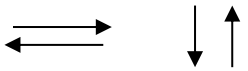


3. සැකසීම -Process 

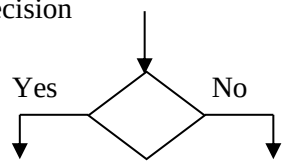
උදාහරණ:



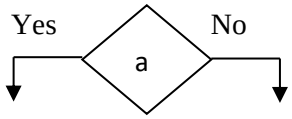
4. දත්ත ගැලීම - Data Flows / Flow line



5. තීරණ - Decision



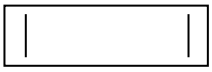
උදාහරණ:



6. සම්බන්ධකය - Connector



7. අනු සැකසුම - Sub Process



ව්‍යාජ කේත/ කල්පිත කේත (Pseudocode)

පරිගණක වැඩසටහනක් සකස් කිරීමේදී ක්‍රමලේඛයක් වඩාත් පහසුවෙන් ඇල්ගොරිතමයක් තේරුම් ගැනීම සඳහා ව්‍යාජ කේත භාවිතා කරයි. මෙහිදී ඉංග්‍රීසි භාෂාවේ වචන භාවිතාකර උපදෙස් මාලාව සකස් කෙරේ. මෙම වචන කිසිදු ක්‍රමලේඛ භාෂාවකට භාවිතාකරන වචන නොවූවත් ඒවා බොහෝ දුරට ඒ හා සමාන බවක් පෙන්නුම් කරයි.

ආදානය

Get, read, input

Eg:- Get mark

Read mark

ප්‍රතිදානය

Output, Display, Print

Eg:- Display total

Print total

සැකසීම

Set, process, calculate

Eg:- Set total = 0

හස්ත රේඛණය **(Hand trace)**

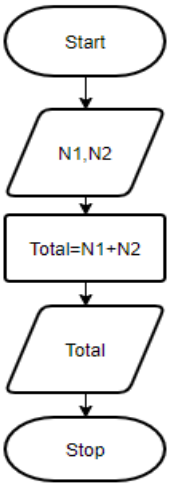
හස්ත ක්‍රම භාවිතයෙන් කේතනය කරන ලද ක්‍රමලේඛනය අදාල දත්ත ආදානය කිරීමෙන් අදාල දත්ත භාවිතා කර එය නිවැරදිව ක්‍රියාත්මක කරන ආකාරය බැලීමට නිර්මානය කරන ලද වගුව හස්ත රේඛනය ලෙස හඳුන්වයි.

Step	Input			Process	Output
	A	B	C	Total= A+B+C	Result
1	10	20	30	10+20+30=60	60
2	70	60	70	70+60+70=200	200
3	60	40	30	60+40+30=130	130
4	10	10	20	10+10+20=40	40

ගැලීම් සටහන් සහ ව්‍යාජ කේත සඳහා උදාහරණ

1. සංඛ්‍යා දෙකක් ආදානය කර එම සංඛ්‍යා දෙකෙහි එකතුව සෙවීම සඳහා ගැලීම් සටහනක් ඇඳ එයට අදාල ව්‍යාජකේතයක් සහ පයිතන් ක්‍රමලේඛනයක් ලියන්න.

ගැලීම් සටහන



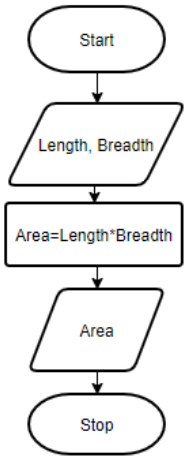
ව්‍යාජකේතය

```
Begin
    Input N1,N2
    Process Total=N1+N2
    Print Total
End
```

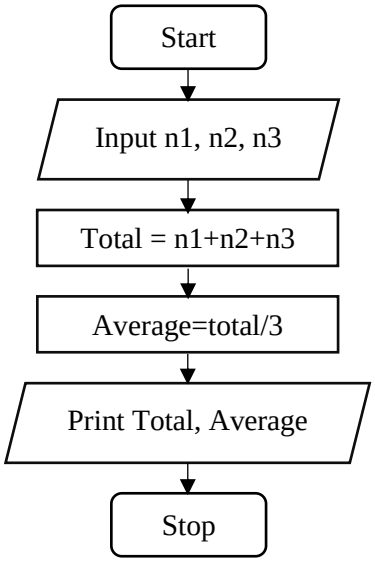
පයිතන් ක්‍රමලේඛනය

```
n1=int(input('Enter number 1:'))
n2=int(input('Enter number 2:'))
total =n1+n2
print('Total is',total)
```

2. සෘජුකෝණාස්‍රයක වර්ගඵලය සෙවීම සඳහා ගැලීම් සටහනක් ඇඳ එයට අදාල ව්‍යාජකේතයක් සහ පයිතන් ක්‍රමලේඛනයක් ලියන්න.

ගැලීම් සටහන	ව්‍යාජකේතය	පයිතන් ක්‍රමලේඛනය
	Begin	l=int(input(Enter Length:'))
	Input Length, Breadth	b=int(input('Enter Breadth:'))
	Process Area=Length*Breadth	area =l*b
	Print Total	print('Area is', area)
	End	

3. සංඛ්‍යා තුනක් ආදානය කර ඒවායේ එකතුව සහ සාමාන්‍ය සෙවීමට ආදාල ගැලීම් සටහනක් ඇද එයට අදාල ව්‍යාජකේතයක් සහ පයිතන් ක්‍රමලේඛනයක් ලියන්න.



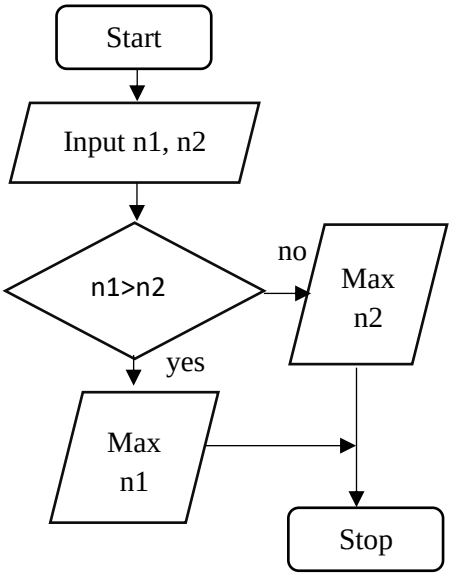
Pseudocode

```
Begin
        Input no1, no2, no3
        Total=no1+no2+no3
        Average=Total/3
        Print Total, Average
End
```

Python Code

```
n1=int(input('Enter a number 1: '))
n2=int(input('Enter a number  2: '))
n3=int(input('Enter a number 3: '))
total=n1+n2+n3
average=total/3
print('Total is : ',total)
print('Average is : ',average)
```

4. සංඛ්‍යා දෙකක් ආදානය කර විශාලතම සංඛ්‍යාව සෙවීමට ආදාල ගැලීම් සටහනක් ඇද එයට අදාල ව්‍යාජකේතයක් සහ පයිතන් ක්‍රමලේඛනයක් ලියන්න.



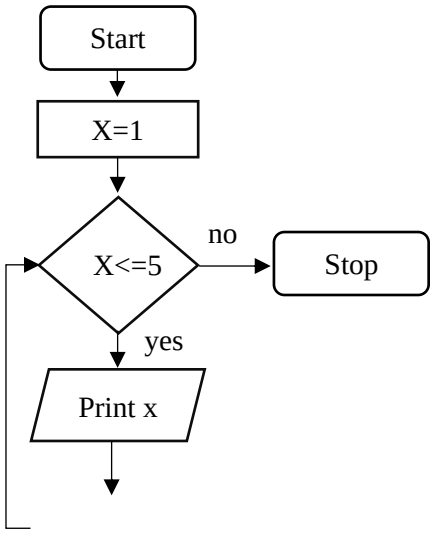
Pseudocode

```
Begin
        Input no1, no2, no3
        If no1>no2 then
                Max=na1
        Else
                Max=no2
        End if
        Print max
End
```

Python Code

```
n1=int(input('Enter a number 1: '))
n2=int(input('Enter a number  2: '))
if n1>n2:
   max=n1
else:
   max=n2
print('maximum number is :',max)
```

5. 1 සිට 5 දක්වා සංඛ්‍යාව මුද්‍රනය කිරීමට ආදාල ගැලීම් සටහනක් ඇද එයට අදාල ව්‍යාජකේතයක් සහ පයිතන් ක්‍රමලේඛනයක් ලියන්න.



Pseudocode

```
Begin
        x=1
        while x<=5 do
                print x
                x=x+1
        end while
End
```

Python Code

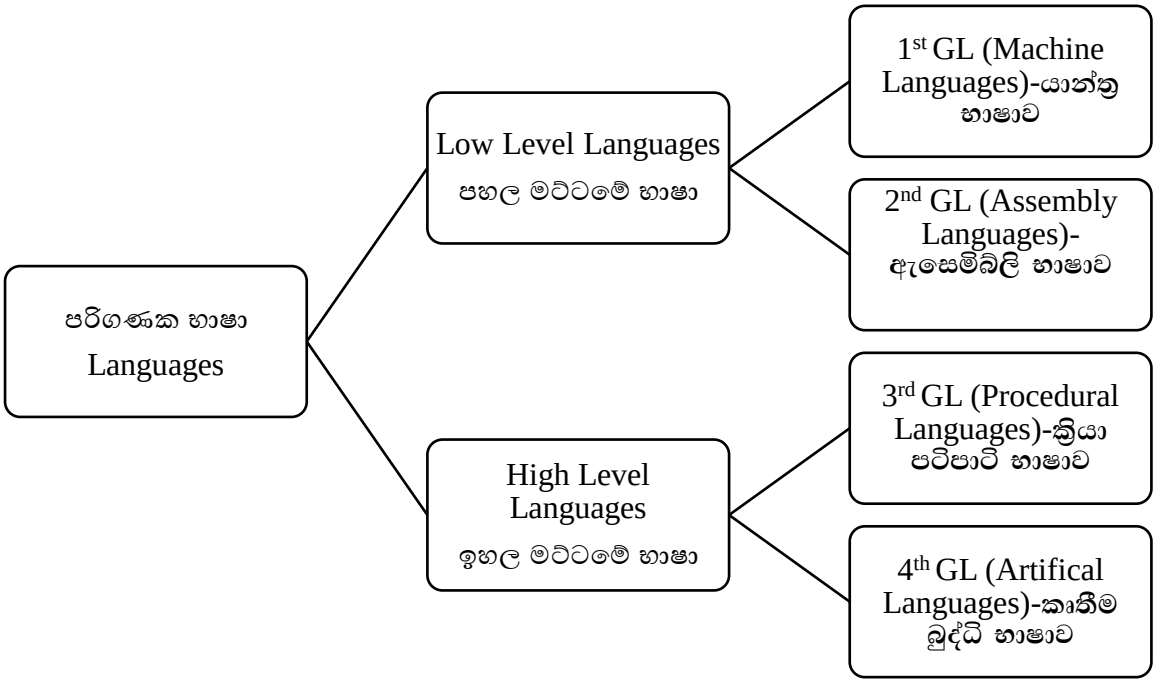
```
x=1
while x<=5:
   print(x)
   x=x+1
```

X=X+1

පරිගණක ක්‍රමලේඛන භාෂා

ක්‍රමලේඛනයක් යනු පරිගණකයට යම් කාර්යයක් කිරීම සඳහා ක්‍රමලේඛන භාෂාවක් මඟින් ලබාදෙන විධාන මාලාවක එකතුවකි. එනම් ක්‍රමලේඛන භාෂාවක් යනු පරිගණකයට එවන් විධාන ලබාදීම සඳහා සැනසූ කෘතීම භාෂාවකි.

පරිගණක භාෂාවන්ගේ පරිණාමය



පළමු පරම්පරාවේ පරිගණක භාෂාව
(**Low Level Languages**)

- පළමු පරම්පරාවේ පරිගණක භාෂා මඟින් ලියන ලද වැඩසටහන් 0 හා 1 යන සංකේත පමණක් භාවිතා කර ලියන ලදී.
- එබැවින් විධාන ඇතුලත් කිරීම ඉතා සංකීර්ණ කාර්යයක් වේ.
- යන්ත්‍ර මත යැපෙන භාෂාවකි.
- ක්‍රමලේඛකයා විසින් පරිගණක දෘඩාංග පිළිබඳ දැනගත යුතු වේ.
- පරිවර්තන මෘදුකාංගයක් අවශ්‍ය නොවේ.
- වැඩසටහන් ක්‍රියාත්මක වීම වේගවත්ය.

දෙවන පරම්පරාවේ පරිගණක භාෂාව

- මෙම පරිගණක භාෂා විධාන යන්ත්‍ර භාෂාවට සාපේක්ෂව සරලය.
- 0 හා 1 මත පදනම් වූ විධාන වෙනුවට සංකේත නාම (mnemonics) භාවිතාවන නිසා මෙය සංකේත භාෂාව ලෙසඳ හැඳින්වේ.
- Assembler නම් වූ සුවිශේෂී භාෂා පරිවර්තකයක් සංකේතාත්මක භාෂා පරිවර්තනයට යොදාගනී.
- යන්ත්‍රය මත යැපෙන භාෂාවකි.
- මෙම ක්‍රමලේඛන පළමු පරම්පරාවට වඩා ක්‍රියාත්මක වේගය අඩුය.

තෙවන පරම්පරාවේ පරිගණක භාෂාව
(**High Level Languages**)

- මෙම පරිගණක භාෂාව ඉහල මට්ටමේ පරිගණක භාෂා ලෙස හඳුන්වයි.
- ඉංග්‍රීසි භාෂාවට සමාන පරිගණක භාෂා මෙම ගණයට ගැනේ.
- ක්‍රමලේඛන සම්පාදනය පහසු කරයි.
- යන්ත්‍රය මත යැපීමක් සිදුනොවේ.
- ක්‍රමලේඛකයෙකු පුහුණු කිරීම පහසු වේ.
- පරිවර්තන මෘදුකාංග අවශ්‍ය වේ.
  උදා:- 1. සම්පාදක (Compiler)
  2. අර්ථවිනාශක (Interpreter)
- FORTAN (Formula **Tran**slation)
- COBOL (Common **B**usiness **O**riental **L**anguages)
- Pascal
- C
- C^{++}
- Java
- Basic (**B**eginners **A**ll Purpose **S**ymbol **I**nstruction **C**ode)
- Python
- Delphi
- Visual Basic

කෘත්‍රිම බුද්ධි සඳහා පරිගණක භාෂා

- තීරණ ගැනීම සඳහා පරිගණක මාධ්‍යය මඟින් නිර්මාණය කිරීමට භාවිතා කරයි.
- පරිගණක වලට මිනිසුන් භාවිතා කරන භාෂා තේරුම් ගැනීමට පහසුවෙන් මෘදුකාංග නිර්මාණයට පහසු වේ.
- නිපුණතා පද්ධති නිර්මාණය සඳහා භාවිතා වේ.
- රොබෝ යන්ත්‍රවලට සංවේදී උත්තේජකවලට ප්‍රතිචාර දැක්වීම සඳහා වැඩසටහන් නිර්මාණය කිරීමට භාවිතා කරයි.

පරිගණක භාෂා පරිවර්තක මෘදුකාංග

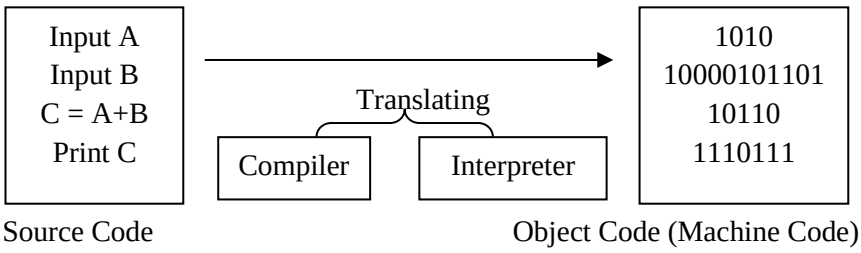
ප්‍රභව කේතය හා විභව කේතය

ප්‍රභව කේතය **(Source Program)**

පරිගණක භාෂාවකින් ලියන ලද කේතයක් පරිවර්තනය කිරීමට පෙර මුල් පිටපත මෙනමින් හැඳින්වේ.

විභව කේතය **(Object Program)**

පරිගණක භාෂාවකින් ලියන ලද කේතයක් යන්ත්‍ර භාෂාව බවට පරිවර්තනය කලපසු ලැබෙන කේතය විභව කේතය ලෙස හඳුන්වයි.



පරිගණක භාෂාවකින් ලියන ලද කේතයක් යන්ත්‍ර භාෂාව බවට හැරවීමේදී එහි ක්‍රියාකාරීත්වය අනුව ප්‍රධාන වර්ග 3ට බෙදිය හැක. එනම්,

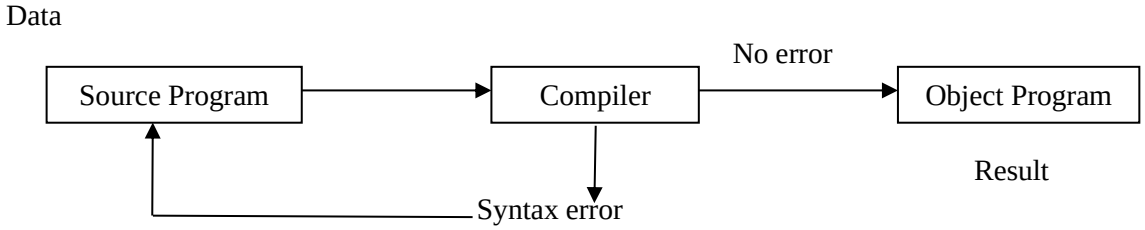
1. තුන්වන පරම්පරාවේ පරිවර්තක මෘදුකාංග ලෙස සම්පාදක හා අර්ථ විනාශක හැඳින්විය හැකිය.
2. දෙවන පරම්පරාවේ පරිගණක මෘදුකාංග ලෙස ඇසෙම්බ්ලර් හැඳින්විය හැක.

සම්පාදක **(Compiler)**

මෙම වර්ගයේ මාදුකාංග මඟින් ක්‍රමලේඛයේ ඇති සියළු විධාන එකවර යන්ත්‍ර භාෂාව බවට හැරවීම Compiler ලෙස හඳුන්වයි. මෙසේ පරිගණකගත වූ පසු එම යාන්ත්‍රික භාෂාවෙන් වූ ක්‍රමලේඛය විධාන දත්ත මත ක්‍රියාත්මක කළහැක. ක්‍රමලේඛය ක්‍රියාත්මක කිරීමේ වේගය අර්ථ විනිවිදයට වඩා වැඩිය.

උදා:-

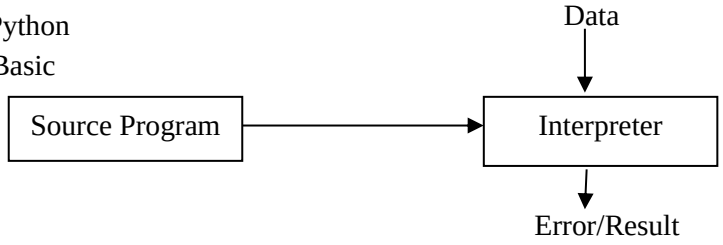
1. Pascal
2. Fortran
3. C^{++}
4. C
5. C #
6. Visual Basic



අර්ථ විනිවිදයක **(Interpreter)**

මෙම පරිවර්ථක මෘදුකාංගය මඟින් ක්‍රමලේඛයේ අඩංගු වගන්ති පියවරෙන් පියවර යන්ත්‍ර භාෂාව බවට පරිවර්තනය කර විදාන එසැනින් ක්‍රියාත්මක වීමද සිදු වේ. මෙම ක්‍රමයේදී ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීමට ගන්නා කාලය සම්පාදක මඟින් පරිවර්තනය කල ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීමට ගන්නා කාලය වැඩිය.

උදා:- 1. Python
2. Basic

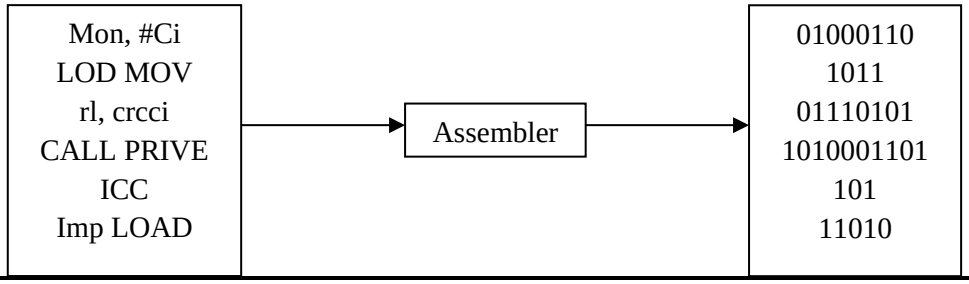


සම්පාදක හා අර්ථ විනිවිදයක අතර වෙනස්කම්

සම්පාදක (Compiler)	අර්ථ විනිවිදයක (Interpreter)
ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීමට පෙර සම්පූර්ණ ක්‍රමලේඛයක් එකවර යන්ත්‍ර භාෂාව බවට හැරවිය යුතුය.	ක්‍රමලේඛයක් පියවරෙන් පියවර යන්ත්‍ර භාෂාවක් බවට හරවන අතර එසැනින්ම ක්‍රියාත්මක වීමද සිදුවේ.
ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීමේ වේගය වැඩිය.	ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීමේ වේගය අඩුය.

එසෙම්බ්ලි **(Assembler)**

ප්‍රාථමික මට්ටමේ පරිගණක භාෂාවකින් ලියන ලද ක්‍රමලේඛයක් (mnemonics) යන්ත්‍ර භාෂාව බවට පරිවර්තනය කිරීම එසෙම්බ්ලි ලෙස හඳුන්වයි.



Assembly Language Program Object Program

ක්‍රමලේඛන දෝෂ - **Programming Errors (Bugs)**

1. **කාරක දෝෂ** (Syntax Error) (Compile-time error)
    1. වැරදි මූලපද එක් කිරීම (Wrong Keyword)
    2. වැරදි විරාම ලකුණු භාවිතා කිරීම (Wrong Punctuation Marks)
2. **ක්‍රමලේඛයක් ක්‍රියාත්මක වීමේදී ඇතිවන දෝෂ** (Run-time error)
    1. සංඛ්‍යාවක් බිංදුවෙන් බෙදීම (Division by Zero)
    2. ඍණ සංඛ්‍යා වල වර්ග මූලය සෙවීම (Find Square root in negative number)
3. **තාර්කික දෝෂ** (Logical error)
    1. Unexpected Output (**අපේක්ෂිත ප්‍රතිඵල නොපෙන්වීමේදී ඇතිවන දෝෂ**)

පයිතන් ක්‍රමලේඛන භාෂාව (Python Programming Language)

මෙය ඉහල මට්ටමේ ක්‍රමලේඛ භාෂාවක් වන අතර මෙම භාෂාව 90 දශකයේ මුල් භාගයේ නෙදර්ලන්තයේ CWICN (Research Institute of Mathematics and Computer Science) ආයතනයේ **Guido Van Rossum** විසින් නිර්මාණය කරන ලදී. මෙම භාෂාව අර්ථ වින්‍යාසක මත පදනම් වූ වස්තුගත (Object Oriented) භාෂාවක් වන අතර පටිපාටිගත භාෂාවක් ලෙසද යොදාගත හැක. විවෘත මෘදුකාංග මූලාශ්‍ර ක්‍රමලේඛන භාෂාවක් වන අතර http://www.python.org වෙබ් අඩවියෙන් නොමිලේ බාගතකර ගත හැක. පයිතන් ක්‍රමලේඛන භාෂාව පහත මෙහෙයුම් පද්ධති මත ක්‍රියාත්මක කලහැක.

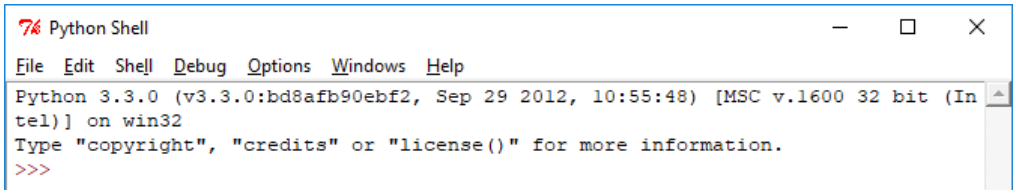
1. Linux / Unix
2. Mac OS X
3. OS / 2
4. Amiga
5. Palm Handhelds
6. Nokia mobile phones

Python Shell විවෘත කිරීම

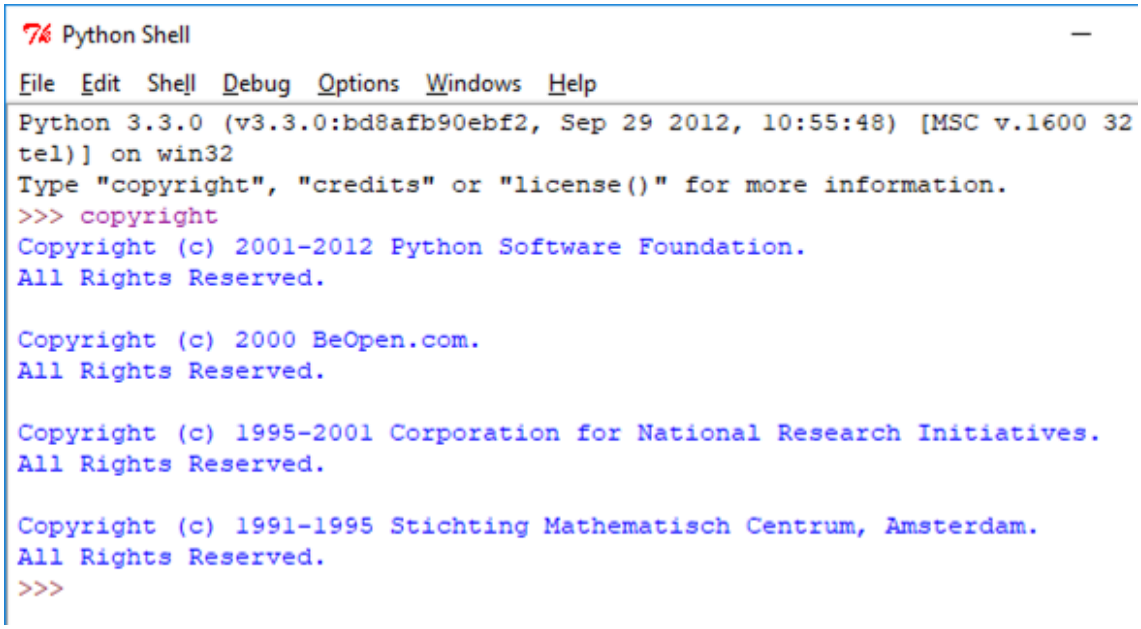
Start→ All Program→ python 3.3→ IDLE (python GUI)

Python හි අතුරු මුහුණත් ආකාර 2කි.

1. **රූපක අතුරු මුහුණත**
2. විධාන රේඛා (Command line) **අතුරු මුහුණත**

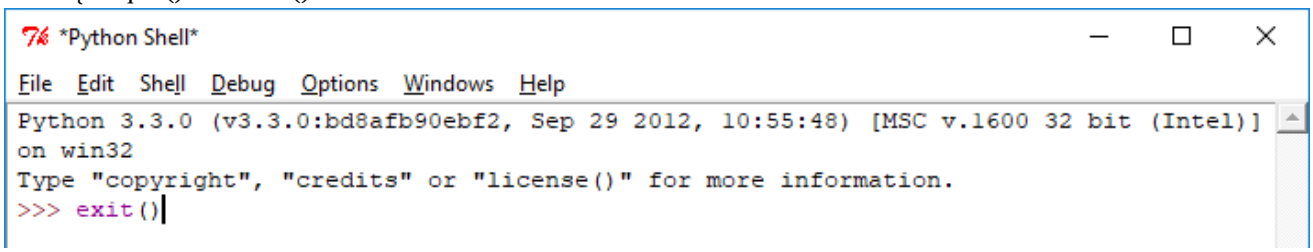


Copyright විධානය

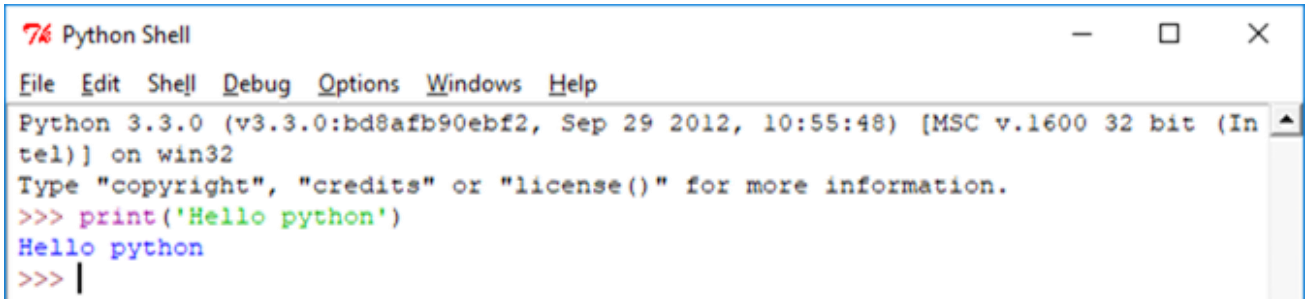


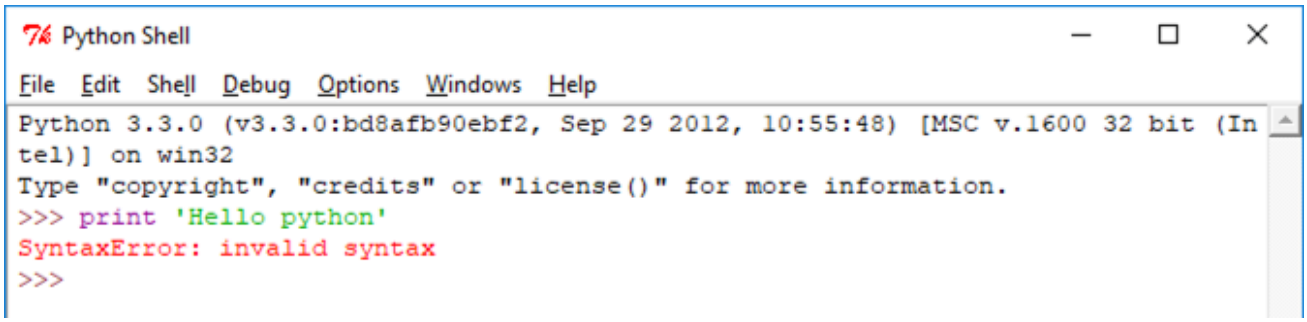
පයිතන් shell එකෙන් ඉවත් වීම

මේ සඳහා quit() හෝ exit() විධානය භාවිතා කරයි.



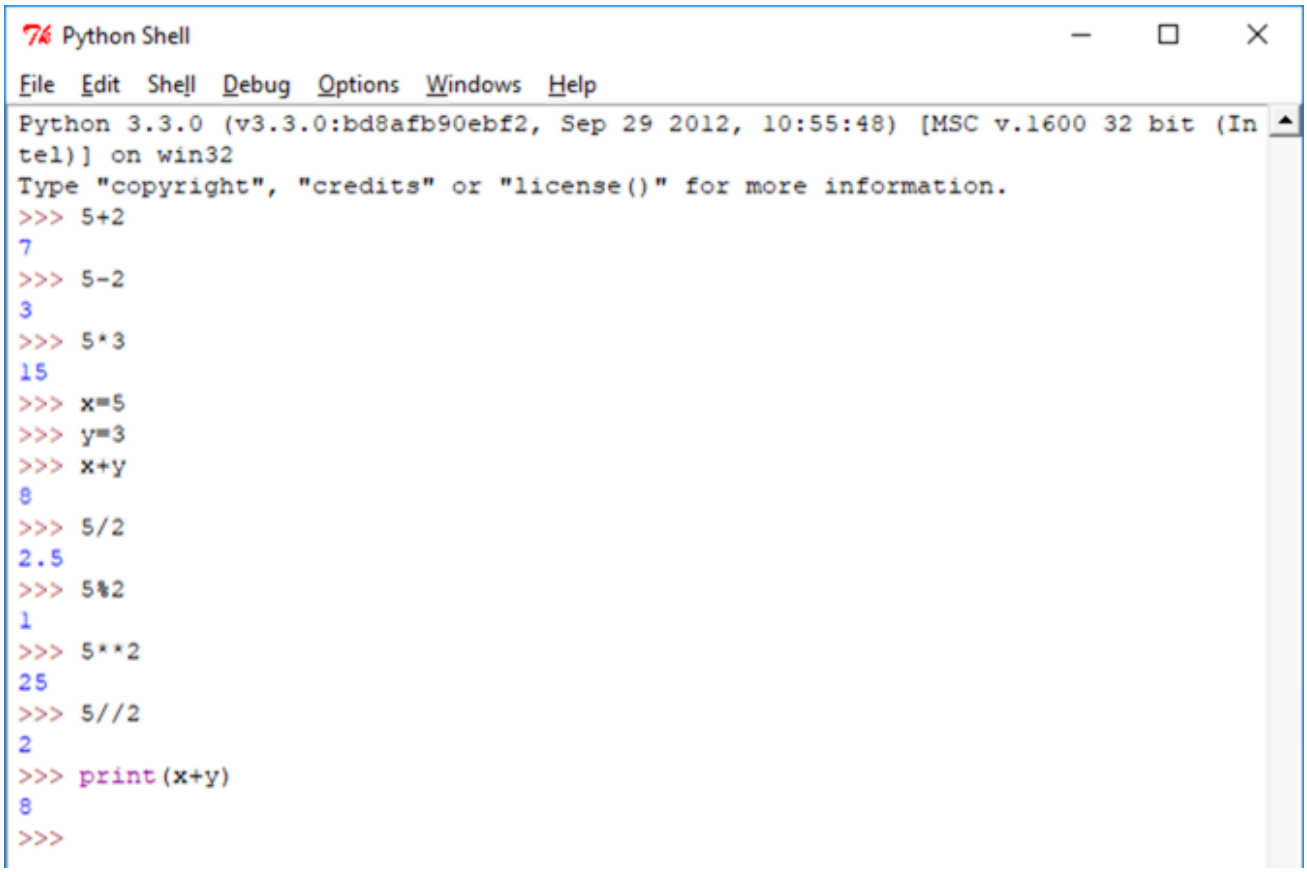
Hello Python ලෙස මුද්‍රණය කිරීම





Python නව සංස්කරන වලදී print විධානය පසුව වරහනක් ( ) තුල මුද්‍රණය කළ යුතු වගන්තිය ලිවිය යුතුය. නැතහොත් ඉහත ආකාරයේ දෝෂ පණිවිඩයක් නිකුත් කරයි. මුද්‍රණය කල යුත්තේ වගන්තියක් (String Values) නම් එය උද්ධෘත තුල දැක්විය යුතුය. පයිතන් භාෂාවේදී තනි උද්ධෘතය (‘Hello python’) , ද්විත්ව උද්ධෘතය (“Hello python”) සහ ත්‍රිත්ව උද්ධෘතය (‘‘‘Hello python’’’) වලංගු වේ.

Python Shell තුල ගණිත කර්ම සිදුකල හැක.



ක්‍රමලේඛයක් ලිවීම සඳහා නව ගොනුවක් විවෘත කිරීම

File→ New Window (Ctrl+N)

Print (“Hello World”)

ක්‍රමලේඛයක් ගබඩා කිරීම

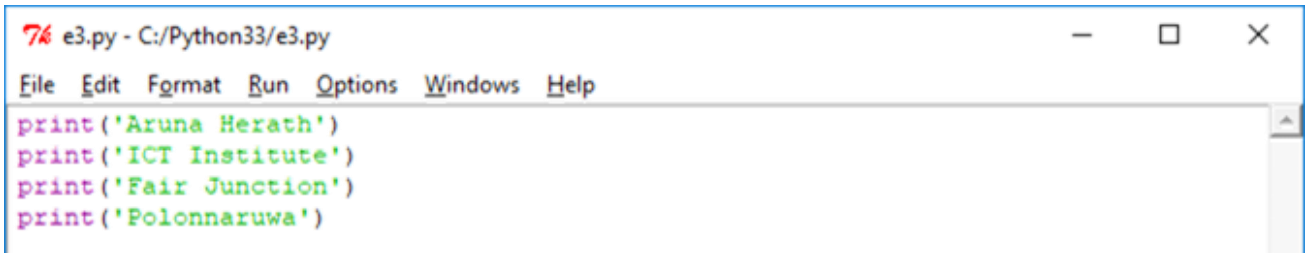
File → Save→ (extension →.py) .py හෝ .pyw ගොනු දිගු භාවිතා කළ හැක.

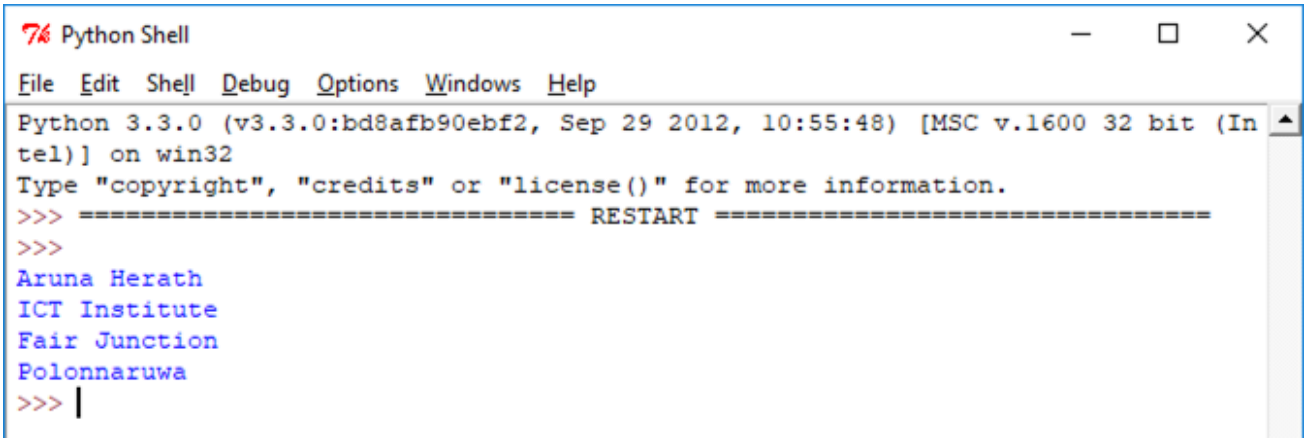
ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීම

Run→ Run Module (F5)

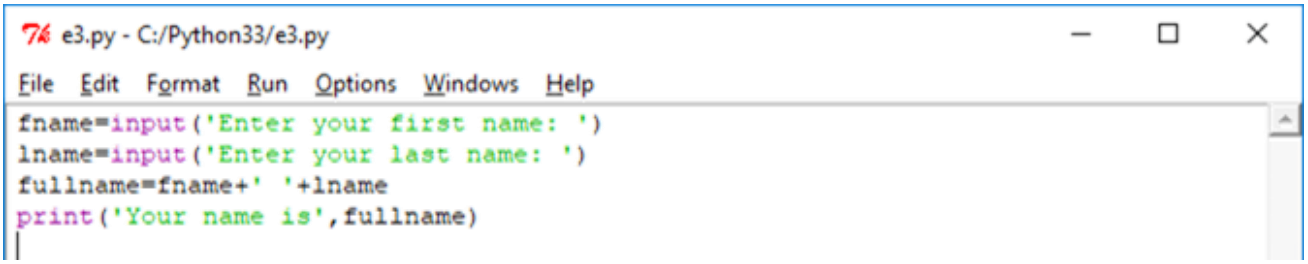
>>>Hello World

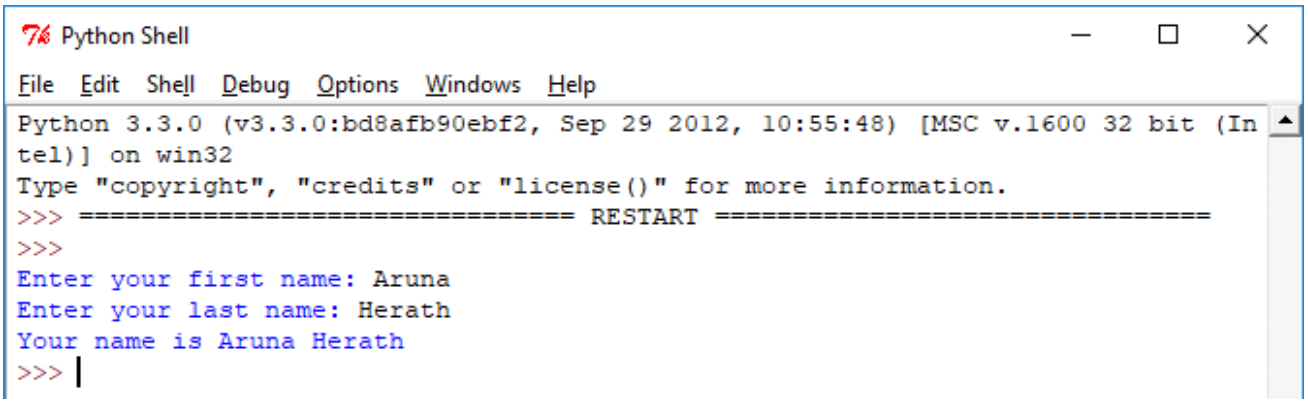
ඔබගේ නම සහ ලිපිනය මුද්‍රණය කිරීම සඳහා පයිතන් ක්‍රමලේඛනයක් ලිවීම.





දත්තයක් යතුරු පුවරුව මඟින් ක්‍රමලේඛයට ආදානය කිරීම සඳහා **input** විධානය භාවිතා කරයි.



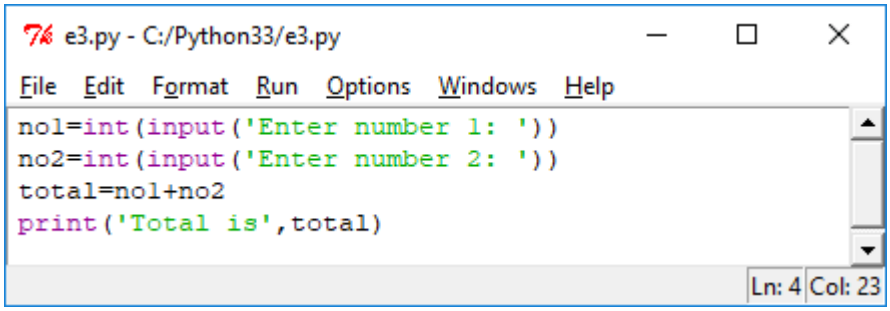


නම සහ වයස ඇතුලත් කිරීම සඳහා පයිතන් ක්‍රමලේඛනයක් ලිවීම

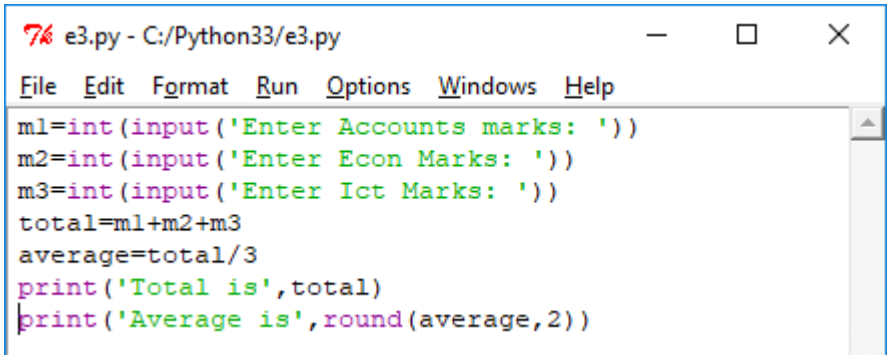
```
name=input('Enter your first name: ')
age=input('Enter your age: ')
print('Your name is',name, 'your age is',age)
```

```
>>>
Enter your first name: Aruna
Enter your age: 34
Your name is Aruna your age is 34
>>>
```

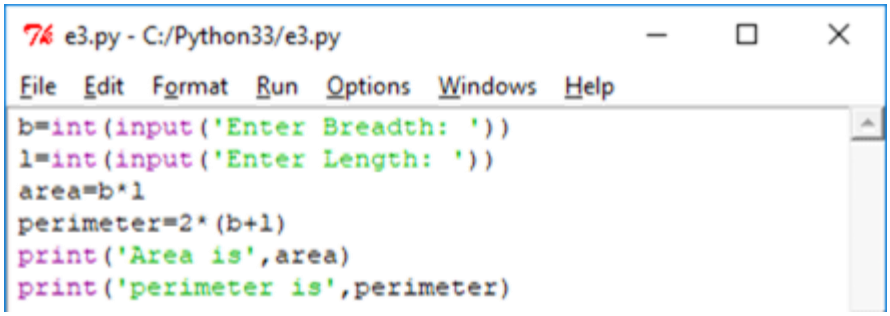
සංඛ්‍යා 2ක් ආදානය කර ඒවායේ එකතුව සෙවීම



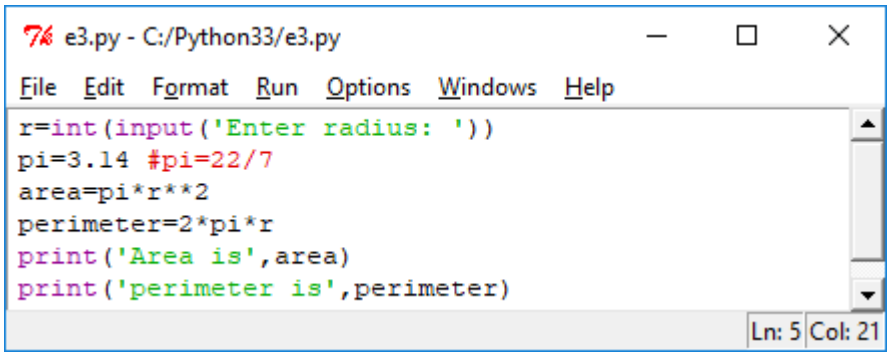
විෂයන් තුනක ලකුණු ඇතුලත් කර ඒවායේ එකතුව සහ සාමාන්‍ය සෙවීම



සෘජුකෝනාස්‍රයක වර්ගඵලය සහ පරිමිතිය සෙවීම



වෘත්තයක වර්ගඵලය සහ පරිමිතිය සෙවීම



කාරක රීති සහ ශබ්දාර්ථ (Syntax and Semantics)

පරිගණක භාෂා නිර්වචනය කිරීම සඳහා ප්‍රධාන සංඝටක 2ක් අවශ්‍ය වේ.

1. කාරක රීති (Syntax)
2. ශබ්දාර්ථ (Semantics)

කාරක රීති **(Syntax)**

ක්‍රමලේඛ භාෂාවක් කාරක රීති මඟින් ක්‍රමලේඛ භාෂාවේ ඇති ව්‍යාකරණ වලට අනුව සංකේත සම්බන්ධ කලයුතු ආකාරය නිර්වචනය කරයි. මෙම කාරක රීතිවලින් නිර්වචනය වන්නේ සංකේතවල ව්‍යුහයන්ගේ නිරවද්‍යතාව පිළිබඳ පමණක් වන අතර ඒවායේ අර්ථය පිළිබඳ සැලකිල්ලක් නොදක්වයි.

උදා:- Dog is a bird

ශබ්දාර්ථ **(Semantics)**

මෙමඟින් භාෂාවක ඇති ව්‍යාකරණ ව්‍යුහයන්ට අනන්‍ය අර්ථ පවරනු ලැබේ. එමඟින් භාෂාවකින් සම්පාදනය කල වැඩසටහන් ක්‍රියාත්මක කිරීමේදී අනුගමනය කලයුතු චර්යාව ශබ්දාර්ථ මඟින් නිර්වචනය කෙරේ.

පයිතන් හඳුන්වන **(Python Identifiers)**

හඳුන්වනයක් යනු විචල්‍යයක්, නියතයක්, ක්ලාස්, මොඩියුල හෝ ශ්‍රිතයක් (Variable, Constant, Function, Class, Module) හැඳින්වීම සඳහා යොදාගන්නා නාමයකි. Identifier නිර්මාණය කිරීමේදී අනුගමනය කලයුතු නීති පහතින් දැක්වේ.

- හඳුන්වනයක් ඉංගීසි භාෂාවෙහි අක්ෂරයකින් හෝ යටි ඉර underscore (_) මඟින් දැක්විය යුතුය.
- පළමු අක්ෂරයට පසු එන අක්ෂරය සඳහා ඉහත ආකාරයේ අක්ෂර හා 0-9 වූ ඉලක්කම්ද භාවිතා කල හැකිය.
- තිබිය හැකි අනුලක්ෂණ සංඛ්‍යාවේ සීමාවක් නොමැත.
- Python හි භාවිතා වන මූල පද භාවිතා කල නොහැක.
- ඉංග්‍රීසි භාෂාවේ Simple අකුරු හා Capital අකුරු එකිනෙකට වෙනස් අක්ෂර ලෙස සලකනු ලැබේ.(Mark ≠ mark)
- @, $, % යන අනුලක්ෂණ භාවිතා කල නොහැක.

Python මූල පද/අවුරුනු පද **(Reserved Words)**

Python භාෂාවේද අනෙක් බොහෝ පරිගණක භාෂාවල මෙන් මූලපද ලෙස අර්ථ දැක්වූ වචන ලැයිස්තුවක් වේ. ක්‍රමලේඛන ගොඩනැගීමේදී එම වචනවල භාවිතය ඒවායේ අනුරූප අර්ථයන්ට සිදුකල යුතුය. python භාෂාවේ ඇති මූල්පද ලැයිස්තුවක් පහත දැක්වේ.

and	not
as	or
assert	pass
break	print
class	raise
continue	return
def	try
del	while
elif	with
else	yield
except	true
exec	false
finally	None
for	
from	
global	
if	
import	
in	
is	
lambda	

පරිසීමක (Delimiters)

ප්‍රකාශන කොටස් වෙන්කර දැක්වීම සඳහා පරිසීමක භාවිතා කරයි. පයිතන් භාෂාවේ භාවිතා වන පරිසීමක කිහිපයක් පහත දැක්වේ.

( )	‘	%=
[ ]	=	<=
{ }	!	^ =
@	+=	>>=
,	-=	<<=
:	/=	** =
.	//=	!=

උදා:- a=5
b=10
a=a+b → a+ = b

print (“Answer is” , a)
Statement Variable
Delimiters

බහු පේළි ප්‍රකාශන (**Multi – Line Statements**)

Explicit Line Join (\ )

ක්‍රමලේඛනයක කිසියම් පේලියක් ‘ \ ’ යන අනු ලක්ෂණයෙන් අවසාන වී ඇති අතර එම පේලිය ඊට පසුව ඇති පේලිය සමඟ සම්බන්ධ කරනු ලබයි.

e.g :- print(‘ict \
institute)

Impact Line Join – ( ) [ ] { }

- [ ], ( ) හා { } යන වරහන් තුල වූ ප්‍රකාශ පේලි කීපයක් කේත කලහැක.

e.g :- x=[5,2,
3,4]

පයිතන් හි භාවිතා කලහැකි උද්දෘත (Quotes)

Python භාෂාවේ භාවිතා කරන අක්ෂර පාඨ හෝ ඡේද දැක්වීම සඳහා ‘, “, “ “ “ භාවිතා කරයි.

උදා:-

1. a = ‘ It is a cat’
2. a = “ It is a Cat ”
3. a = ‘‘‘It’s a Cat’’’ → a= “““It’s a Cat””” (තනි උද්ධෘත තුනක් හෝ ද්විත්ව උද්දෘත තුනක් භාවිතා කළ හැක.)

පේළි සහ අනුඡේද (Lines and Indentation)

පයිතන් භාෂාවේ භාවිතා වන ප්‍රකාශ අනුක්‍රමයන් කට්ටයක් ලෙස හැඳින්වේ. ලේඛනයක ඇති මෙවැනි කොටසක් ලිවීමේදි අනුඡේදනය භාවිතා කිරීම ඉතා වැදගත් වේ. මෙහිදී ක්‍රමලේඛකයක් ඉදිරියේ ඇති අවකාශය මඟින් අනුඡේදයක් දක්වයි.

e.g:- If True:
print (“True”)
Else:
print (“False”)

පයිතන් හි භාවිතාවන විවරණ (Comments)

ක්‍රමලේඛයක් පිළිබඳ විස්තර ක්‍රමලේඛයක් තුලට ලේඛන ගත කිරීම සඳහා විවරණ භාවිතා කරයි. විවරණ ක්‍රමලේඛනයක ඇති අනෙකුත් ප්‍රකාශ මෙන් ක්‍රියාත්මක නොවේ. ක්‍රමලේඛයක් තුල ඇති විවරණ අන්තර්ගත කිරීමේ මූලික පරමාර්ථය වන්නේ මෙම ක්‍රමලේඛය පරීශීලනය කරන්නෙකුට එය පහසුවෙන් තේරුම් ගැනීමට අවශ්‍ය විස්තර සපයා දීමටය.

උදා:- # First Comment
print (“Hello, python”) # Second Comment

පයිතන් හි විචල්‍ය සහ දත්ත වර්ග
Python Variables and Data Types

ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීමේදී දත්තයක් තාවකාලිකව ප්‍රධාන මතකයේ තබාගැනීමට අවශ්‍ය වේ. මෙසේ දත්තයක් මතකයේ තාවකාලිකව තබාගැනීම සඳහා විචල්‍ය භාවිතා වේ. පරිගණකයේ ප්‍රධාන මතකයේ නිශ්චිත කොටසකට ප්‍රවේශවීම සඳහා උපයෝගී කරගත හැකි සංකේත නාමයක් ලෙස විචල්‍ය දැක්විය හැකිය. මෙලෙස විචල්‍යක් මතකයේ කිසියම් කොටසකට තාවකාලිකව අනුබද්ධ විට විචල්‍ය උපයෝගී කරගෙන එම අදාල මතක කොටසේ දත්ත තාවකාලිකව ගබඩා කිරීම හා නැවත ලබා ගැනීම සිදුකල හැකිය.

විචල්‍යකට අගයක් ආදේශ කිරීම

විචල්‍යකට අගයක් ආදේශ කිරීමේදී සමාන ලක්ෂණ භාවිතා කරයි. එය Assign Operators ලෙස හඳුන්වයි.

Variable Name = Value

උදා:-

1. mark =98
2. age = 18
3. pi = 3.14
4. name = Python
5. x = 5, y = 'ICT', z = 72 — x = 1, y = 1, z = 1 — x=y=z=1 — a =5, b=6 — a,b,=5,6

8. x= 1.5
   y = 'String'
   z = 1
   x,y,z = 1.5, 'String', 1

පයිතන් විචල්‍ය වර්ග **(Python Variable Types)**

කලින් නිර්වචනය කල ලක්ෂණ හා මෙහෙයවන (Operators) සහිත දත්ත සමූහයක් දත්ත ප්‍රරූපයක් ලෙස හැඳින්විය හැකිය. දත්ත භාවිතයට පෙර ඒවායේ ප්‍රරූප දැනගත යුතුය.

පයිතන් වල භාවිතා වන දත්ත වර්ග (**Python standard data types**)

→ Numbers
- Integral
  - Integer -int
  - Boolean -bool
- Real -float
- Complex -complex

→ Sequences
- Immutable Sequences
  - String -str
  - Tuple -tuple
  - Byte -byte
- Mutable Sequences
  - Lists - list
  - Byte Arrays -bytearray

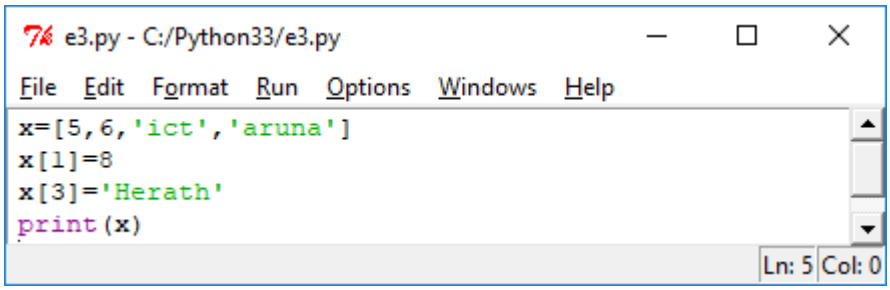
→ Set types
- Sets -set
- Frozen Set -forzenset

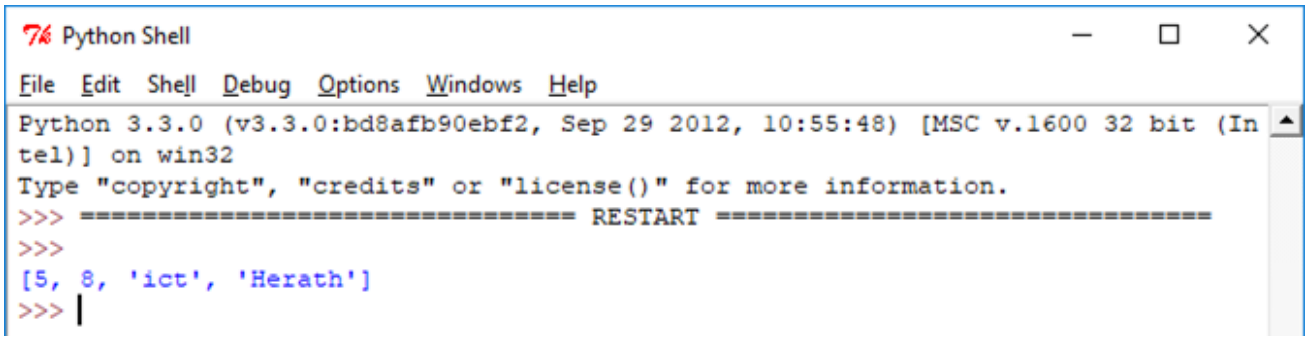
→ Mappings
- Dictionaries -dict

Mutable

Python හි වස්තු ස්වරූපයකට අයත් වස්තූන්ගේ අගයන් එම වස්තු නිර්මාණය කිරීමෙන් පසු ඕනෑම අවස්ථාවක වෙනස් කලහැකි නම් Mutable සණයට අයත් වේ.
උදා:-

1. Dictionaries
2. Set
3. Lists

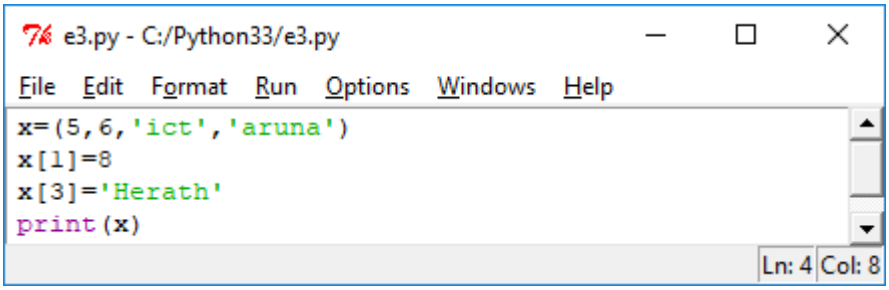


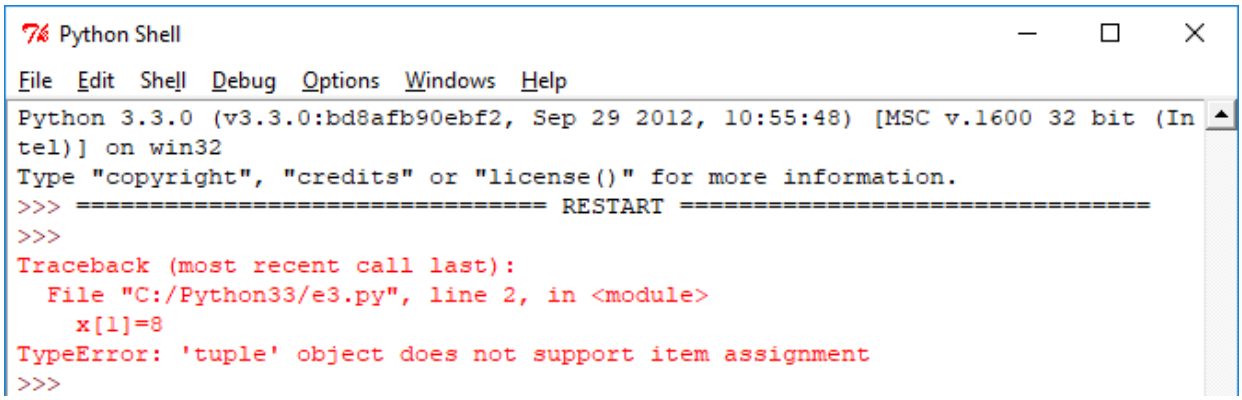


Immutable

වස්තුවක් නිර්මාණය කිරීමෙන් පසු ඒවායේ අගයන් වෙනස් කල නොහැකි නම් එම දත්ත Immutable සණයට අයත් වේ.

- Number
- String
- Tuple
- Frozen test





Iterable

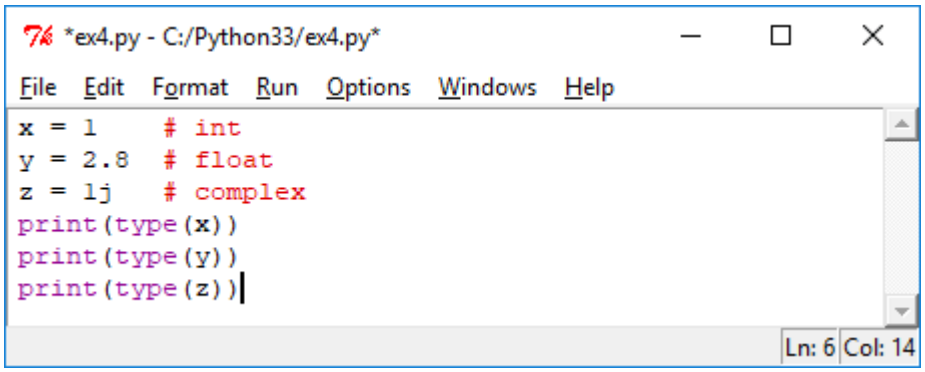
එක් වරකට එක් අවයවය බැගින් මුදාහැරිය හැකි දත්ත එකතුවක් iterable ලෙස හැඳින්වේ.

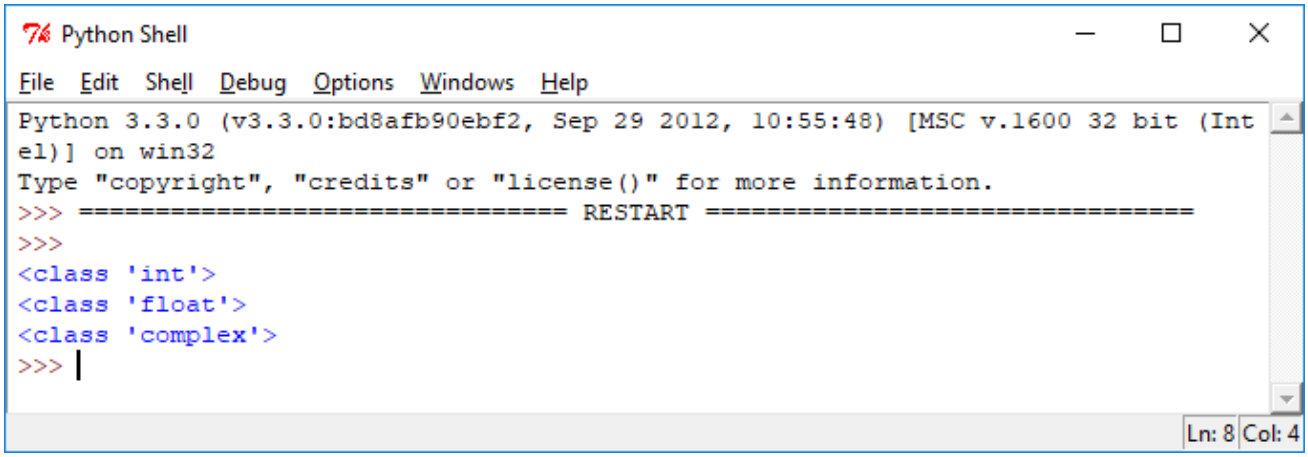
- List
- String
- Tuple

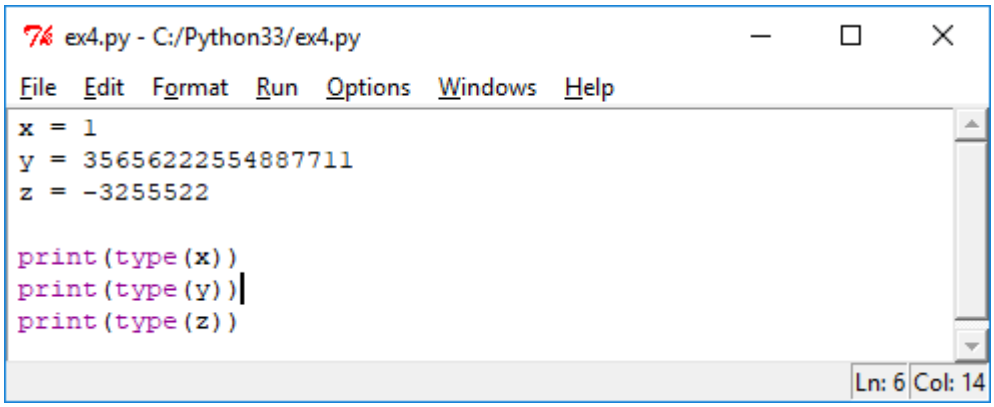
සංඛ්‍යා සහිත දත්ත (Python Number)

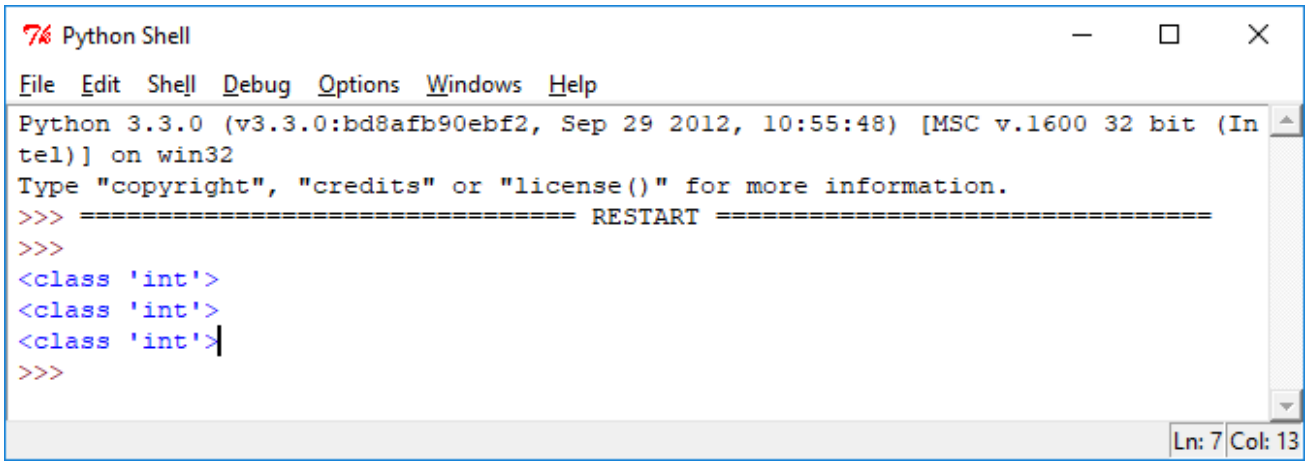
සංඛ්‍යාත්මක අගයන් ගබඩා කිරීම සඳහා භාවිතා කරයි. මෙය මඟින් තබාගත හැකි කුඩාම සහ විශාලතම සංඛ්‍යාව පරිගණක සංඛ්‍යා නිරූපනය සඳහා ඇති සීමාවන් මත රදා පවතී.

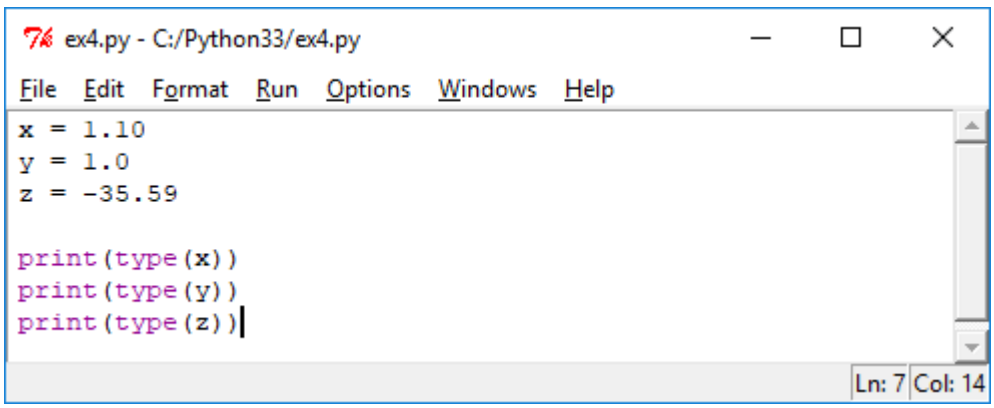
- Int – ධන හෝ සෘණ පූර්ණ සංඛ්‍යා සඳහා භාවිතා කරයි. උදා:- 1,2, -5
- Logn - විශාල පූර්ණ සංඛ්‍යා නිරූපනය සඳහා භාවිතා කරයි. සංඛ්‍යාවේ අවසානයට L හෝ l භාවිතා කරයි. උදා:- 51924361L
- Float - දශම සංඛ්‍යා (තාත්වික සංඛ්‍යා) දැක්වීම සඳහා භාවිතා කරයි. උදා:- 2.25, 5.0, -2.3, 2.5e2 = 2.5 x 10^2 = 250
- Complex - සංකීර්ණ සංඛ්‍යා දැක්වීම සඳහා භාවිතා කරයි. සංඛ්‍යාවේ අවසානයට J හෝ j යොදයි. උදා:- 3.14j

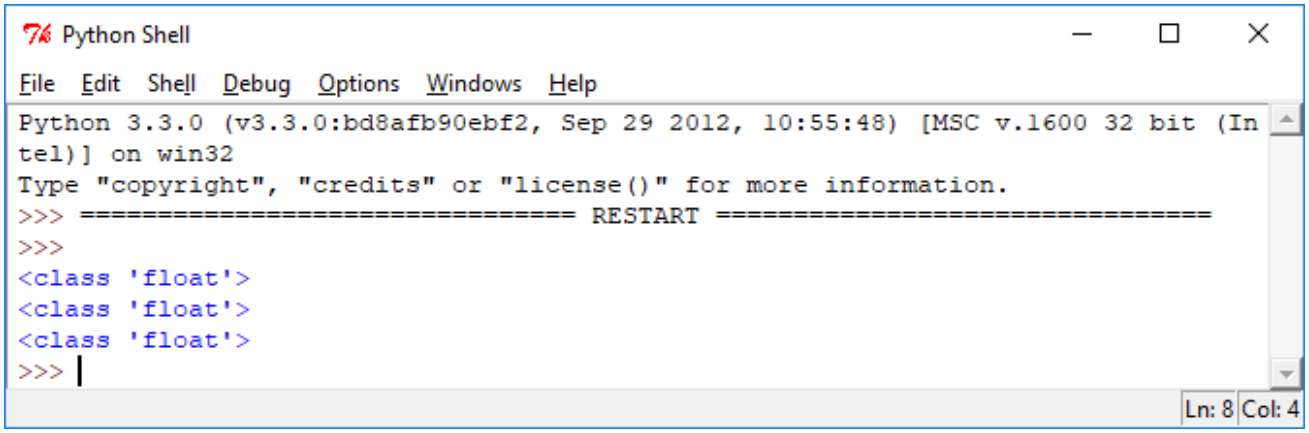


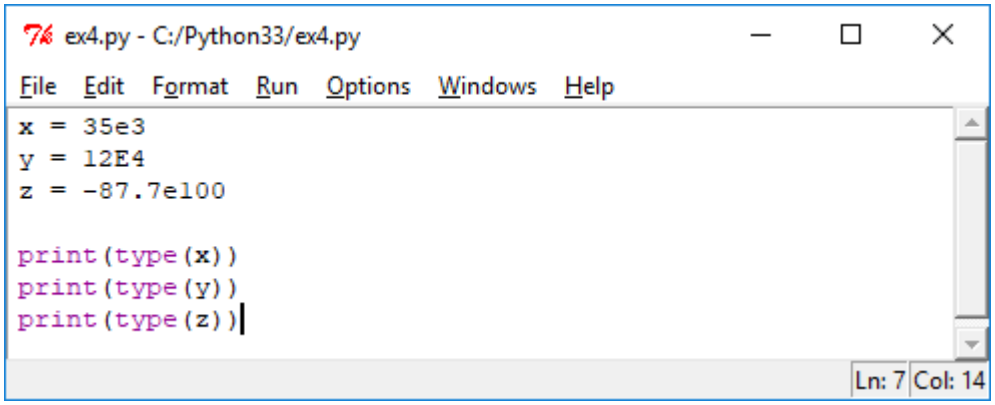

Python Shell
File Edit Shell Debug Options Windows Help
Python 3.3.0 (v3.3.0:bd8afb90ebf2, Sep 29 2012, 10:55:48) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
<class 'int'>
<class 'float'>
<class 'complex'>
>>>
Ln: 8 Col: 4

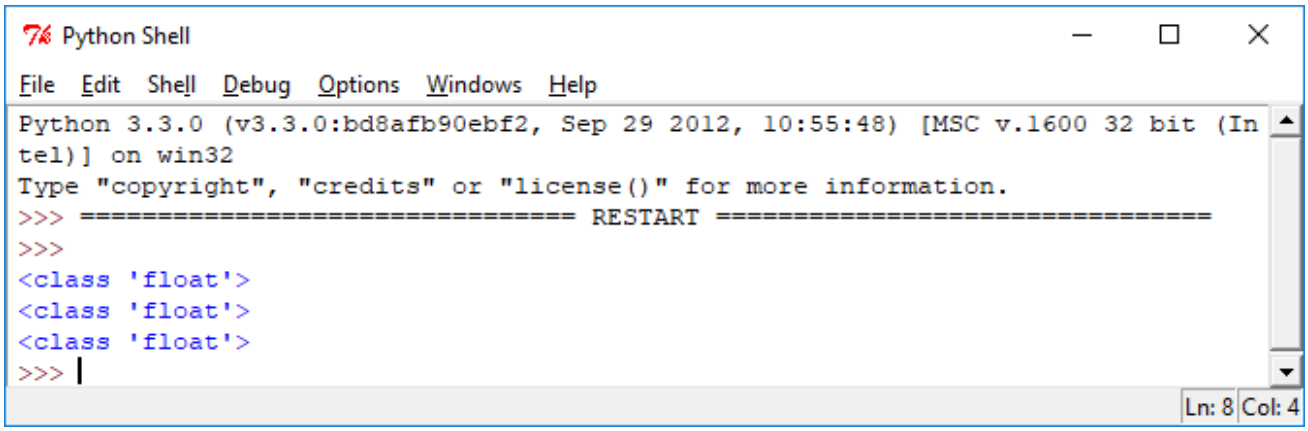

ex4.py - C:/Python33/ex4.py
File Edit Format Run Options Windows Help
x = 1
y = 35656222554887711
z = -3255522
print(type(x))
print(type(y))
print(type(z))
Ln: 6 Col: 14

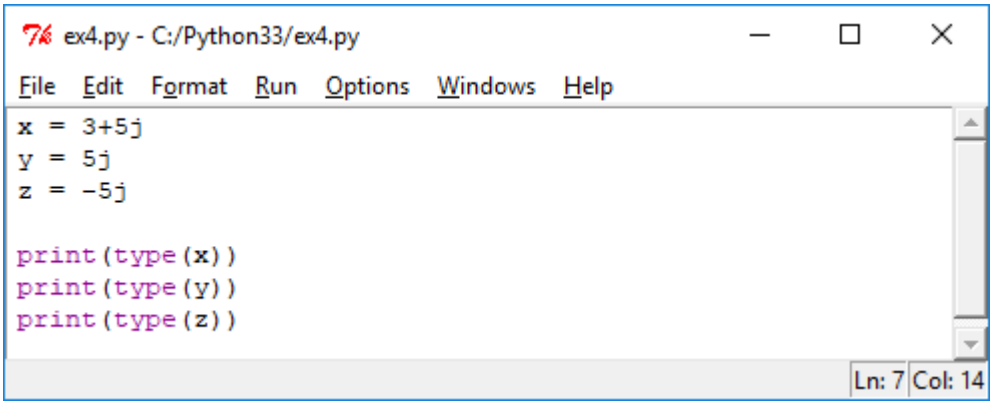

Python Shell
File Edit Shell Debug Options Windows Help
Python 3.3.0 (v3.3.0:bd8afb90ebf2, Sep 29 2012, 10:55:48) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
<class 'int'>
<class 'int'>
<class 'int'>
>>>
Ln: 7 Col: 13

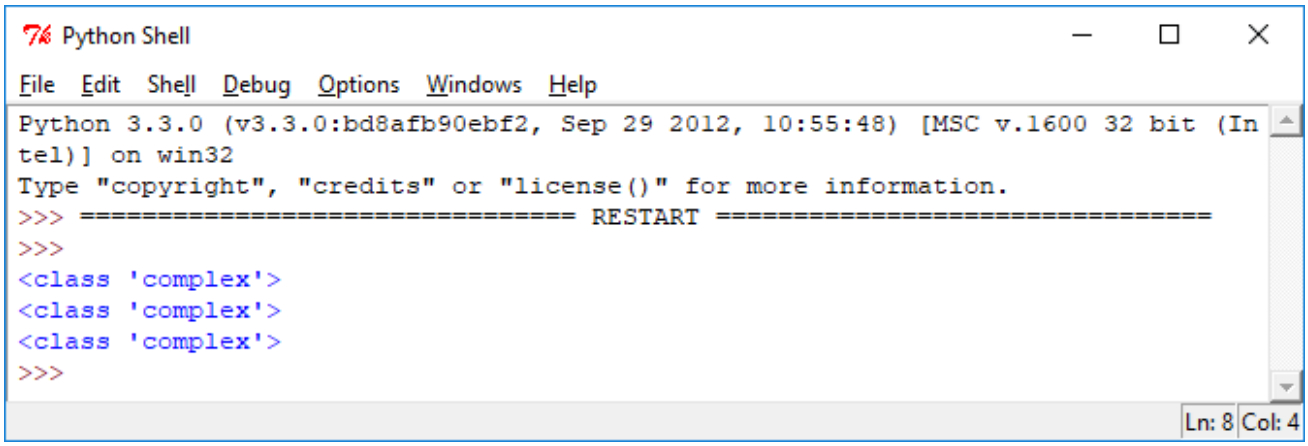

ex4.py - C:/Python33/ex4.py
File Edit Format Run Options Windows Help
x = 1.10
y = 1.0
z = -35.59
print(type(x))
print(type(y))
print(type(z))
Ln: 7 Col: 14


Python Shell
File Edit Shell Debug Options Windows Help
Python 3.3.0 (v3.3.0:bd8afb90ebf2, Sep 29 2012, 10:55:48) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
<class 'float'>
<class 'float'>
<class 'float'>
>>>
Ln: 8 Col: 4


ex4.py - C:/Python33/ex4.py
File Edit Format Run Options Windows Help
x = 35e3
y = 12E4
z = -87.7e100
print(type(x))
print(type(y))
print(type(z))
Ln: 7 Col: 14


Python Shell
File Edit Shell Debug Options Windows Help
Python 3.3.0 (v3.3.0:bd8afb90ebf2, Sep 29 2012, 10:55:48) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ========== RESTART ==========
>>>
<class 'float'>
<class 'float'>
<class 'float'>
>>>
Ln: 8 Col: 4


ex4.py - C:/Python33/ex4.py
File Edit Format Run Options Windows Help
x = 3+5j
y = 5j
z = -5j
print(type(x))
print(type(y))
print(type(z))
Ln: 7 Col: 14



Python Number Log10()

```
import math
print(math.log10(100))
```

Output
2.0

Python Number min()

```
x=10,2,25,4
print(min(x))
```

Output
2

Python Number max()

```
x=10,2,25,4
print(max(x))
```

Output
25

Python Number pow()

```
import math
print(math.pow(5,2))
```

Output
25.0

Python Number round()

```
x=25.25678
print(round(x,2))
```

Output
25.27

Python Number sqrt()

```
import math
print(math.sqrt(25))
```

Output
5.0

අක්ෂර සහිත දත්ත (Python String)
උද්ධෘත තුල දක්වනු ලබන අනුලක්ෂණයක් හෝ අනුලක්ෂණ කාණ්ඩයක් මෙයට අයත් වේ.

උදා:- ‘5’, ‘aruna’, ‘I’, ‘ict institute’

```
>>> '5'
'5'
>>> 'aruna'
'aruna'
>>> 'ict institute'
'ict institute'
>>> x='ict'
>>> print(x)
ict
>>> print(x, 'institute')
ict institute
>>> print(x+'institute')
ictinstitute
>>> print(x+' '+'institute')
ict institute
>>> print(x*2)
ictict
>>> print(len(x))
3
>>> print(type(x))
<class 'str'>
>>> print(x[0])
i
```

```
>>> x='ict institute'
>>> y='polonnaruwa'
>>> print(x,y)
ict institute polonnaruwa
>>> print(x[-1])
e
>>> print([:-1])
SyntaxError: invalid syntax
>>> print(x[:-1])
ict institut
>>> print(x[::])
ict institute
>>> print(x[::2])
itisiue
>>> print(x[::3])
i ste
>>> print(x[-1:])
e
>>> "ict"+'institute'
'ictinstitute'
>>> 'ict institute'[2:]
't institute'
>>> print([2:])
SyntaxError: invalid syntax
>>> print(x[2:])
t institute
>>>
```

```
>>> x='Python is an interpreted'
>>> print("%s is a programming language"%x[:6])
Python is a programming language
>>> x[:6]
'Python'
>>> max(x)
'y'
>>> min(x)
' '
>>> min('Ict')
'I'
>>> x.count('i')
2
>>> x.count('e')
3
>>> x.find('h')
3
>>> x[::-1].startswith('d')
True
>>>
```

```
>>> x='ict institute'
>>> print(x.title())
Ict Institute
>>> print(x.upper())
ICT INSTITUTE
>>> print(x.lower())
ict institute
>>> print(x.capitalize())
Ict institute
>>> print(x.replace('ict','abc'))
abc institute
>>> print(x.split(' '))
['ict', 'institute']
```

```
>>> print('%s is %d years old.'%('nimal',16))
nimal is 16 years old.
>>> print('ict'+\
      'institute')
ictinstitute
>>> print('ict\ninstitute')
ict
institute
>>> print('ict\tinstitute')
ict     institute
```

end( ) විධානය

```
a='ict'

b='institute'

c='Polonnaruwa'

print(a ,end=',')
print(b ,end=',')
print(c ,end=',')
```

ප්‍රතිදානය

ict, institute, polonnaruwa,

```
>>> x='ict'
>>> for i in x:
        print(i)

i
c
t
>>> print(x.split(','))
['ict', 'bit', 'fit', 'cit']
>>> y=x.split(',')
>>> for i in y:
        print(i)


ict
bit
fit
cit
>>>
```

දත්ත ව්‍යුහ (Data Structure)

ක්‍රමලේඛනයක් තුල දත්ත කාර්යක්ෂමව යොදා ගැනීම සඳහාත් දත්ත පිළිවෙලකට සංවිධානය කිරීම සඳහාත් දත්ත ව්‍යූහ භාවිතා කරයි. පයිතන් ක්‍රලේඛන භාෂාවේදී භාවිතා කරනු ලබන දත්ත ව්‍යූහ ආකාර කිහිපයකි.

1. List
2. Tuple
3. Dictionary
4. Set

List

පයිතන් ක්‍රමලේඛන භාෂාවේ භාවිතා වන ප්‍රධාන දත්ත ව්‍යුහය මෙයයි. List දැක්වීම සඳහා කොටු වරහන [] භාවිතා කරයි. list එකක් තුල ඇති සෑම අංගයක් සඳහාම සුචි අගයක් (index) ඇත. පළමු සුචිය 0න් ආරම්භ වේ. පයිතන් List එකක් තුල එකිනෙකට වෙනස් දත්ත ප්‍රරූපවලට අදාල දත්ත ඇතුලත් කලහැකිය. List එකක් තුල ඇති අයිතම කොමාවක් මඟින් වෙන්කරනු ලබයි.

```
>>> list1=['ict','aruna','polonnaruwa']
>>> list2=['a','b','c','d','e','f']
>>> list3=[5,6,10,6,2.25]
>>> print(list1)
['ict', 'aruna', 'polonnaruwa']
>>> print(list2)
['a', 'b', 'c', 'd', 'e', 'f']
>>> print(list3)
[5, 6, 10, 6, 2.25]
>>> print(list1,list2,list3)
['ict', 'aruna', 'polonnaruwa'] ['a', 'b', 'c', 'd', 'e', 'f'] [5,
6, 10, 6, 2.25]
```

```
>>> print(list1+list2)
['ict', 'aruna', 'polonnaruwa', 'a', 'b', 'c', 'd', 'e', 'f']
>>> print(list1*2)
['ict', 'aruna', 'polonnaruwa', 'ict', 'aruna', 'polonnaruwa']
>>> print(list1+['a','r','u','n','a'])
['ict', 'aruna', 'polonnaruwa', 'a', 'r', 'u', 'n', 'a']
>>> for i in list1:
        print(i)


ict
aruna
polonnaruwa
>>> for i in list1:
        print(i,end="")


ictarunapolonnaruwa
```

```
>>> x=['aruna','ict',5,6]
>>> print(x[0])
aruna
>>> print(x[1])
ict
>>> print(x[-1])
6
>>> print(x[:2])
['aruna', 'ict']
>>> print(x[::])
['aruna', 'ict', 5, 6]
>>> print(x[1:3])
['ict', 5]
```

List සඳහා භාවිතා කල හැකි ශ්‍රිත

```
>>> #maximum item of a list
>>> x=[5,6,10,2,3]
>>> y=['a','b','c','abc','A']
>>> print(max(x))
10
>>> print(max(y))
c
>>> #minimum item of a list
>>> print(min(x))
2
>>> print(min(y))
A
```

append()

සකස් කර ඇති List එකක අවසානයට නව අයිතමයක් එක් කිරීම සඳහා append භාවිතා කරයි.

```
>>> animals=['cat','dog']
>>> animals.append('cow')
>>> print(animals)
['cat', 'dog', 'cow']
>>> animals.append('fish')
>>> print(animals)
['cat', 'dog', 'cow', 'fish']
```

ගැටලු :

1. සිසුන් පස් දෙනෙකුගේ නම් සහ ලකුණු ඇතුලත් කිරීම සඳහා ලැයිස්තු දෙකක් සකස් කර අවසානයේ සිසුන්ගේ නම් සහ එක් එක් සිසුවා ලබාගත් ලකුණු දැක්වීමටත් එම ලකුණු වල එකතුව සහ සාමාන්‍යය දැක්වීමටත් පයිතන් ක්‍රමලේඛනයක් ලියන්න.

```
name=[]
marks=[]
total=0
average=0
c=1
while c<=5:
    n=str(input('Enter name: '))
    name.append(n)
    m=int(input('Enter marks: '))
    marks.append(m)
    c=c+1
x=0
print('name \t\t marks')
print('------------------------')
while x<5:
    print(name[x],'\t\t',marks[x])
    total=total+marks[x]
    x=x+1
average=total/5
print('------------------------')
print('Total marks is : ',total)
print('Average marks is : ',average)
```

```
Enter name: Amal
Enter marks: 90
Enter name: Kamal
Enter marks: 95
Enter name: Saman
Enter marks: 85
Enter name: Lalith
Enter marks: 94
Enter name: Bimal
Enter marks: 48
name                marks
------------------------
Amal                90
Kamal               95
Saman               85
Lalith              94
Bimal               48
------------------------
Total marks is :  412
Average marks is :  82.4
```

2. පරිගණක වෙළඳ සැලකින් භාණ්ඩ 5ක් ලබාගෙන ඒවායින් ලබාගත් ඒකක ගනන සහ ඒකකයක මිළද ඇතුළත්කර භාණ්ඩ සියල්ල සඳහා ගෙවිය යුතු මුදල ගණනය කිරීම සඳහා පයිතන් ක්‍රමලේඛනයක් ලියන්න. භාණ්ඩවල නම් සහ ඒකකයක මිළ ලැයිස්තු දෙකකට ඇතුළත් විය යුතුය. ප්‍රතිදානය බිල්පතක් ආකාරයෙන් විය යුතුය.

```
#products
p=[]
up=[]
u=[]
c=1
total=0
while c<=5:
    product=str(input('Enter product name: '))
    p.append(product)
    price=float(input('Enter price of product: '))
    up.append(price)
    unit=int(input('Enter number of units: '))
    u.append(unit)
    c+=1
#find the amount
print()
print("-------------------------------------------------------")
print("ICT Computers (Pvt) Ltd")
print("Hospital Junction - Polonnaruwa")
import datetime
now=datetime.datetime.now()
print(now.strftime('%Y-%m-%d %H:%M:%S'))
print("-------------------------------------------------------")
print("Item Name\tQuentity\tUnit Price\tAmount")
print("-------------------------------------------------------")
for i in range (5):
    print(p[i],'\t\t', u[i], '\t\t',up[i],'\t\t',u[i]*up[i])
    total=total+u[i]*up[i]
print("-------------------------------------------------------")
print('Total price: ',total)
```

```
Enter product name: CD
Enter price of product: 15
Enter number of units: 100
Enter product name: DVD
Enter price of product: 25
Enter number of units: 100
Enter product name: Ram
Enter price of product: 750
Enter number of units: 2
Enter product name: HDD
Enter price of product: 900
Enter number of units: 3
Enter product name: VGA
Enter price of product: 500
Enter number of units: 4

------------------------------------------------------------
ICT Computers (Pvt) Ltd
Hospital Junction - Polonnaruwa
2021-05-22 10:15:10
------------------------------------------------------------
Item Name       Quentity        Unit Price      Amount
------------------------------------------------------------
CD              100             15.0            1500.0
DVD             100             25.0            2500.0
Ram             2               750.0           1500.0
HDD             3               900.0           2700.0
VGA             4               500.0           2000.0
------------------------------------------------------------
Total price:  10200.0
```

count()

List එකක ඇති අවයවයක් කොපමණ වාර ගණනක් යේදී තිබේ දැයි සෙවීමට භාවිතා කරයි.

```
>>> list1=['a','b','a','c','a']
>>> list1.count('a')
3
```

extend()
වෙනත් List එකක ඇති අයිතම දැනට පවතින List එකකට එක් කිරීම සඳහා භාවිතා කරයි.

```
>>> list1=['a','b','a','c','a']
>>> list2=[1,2,3]
>>> list2.extend(list1)
>>> print(list2)
[1, 2, 3, 'a', 'b', 'a', 'c', 'a']
```

insert()
List එකක සුචි අංකය අනුව අයිතම එක් කිරීම සඳහා භාවිතා කරයි.

```
>>> list2=[1,2,3]
>>> list2.insert(0,5)
>>> print(list2)
[5, 1, 2, 3]
```

pop()
List එකක ඇති අවසාන අයිතමය හෝ ලබාදුන් සුචි අගයට අදාල අයිතමය ඉවත් කිරීම සඳහා භාවිතා කරයි.

```
>>> list1=['a','b','a','c','a']
>>> list2=[1,2,3]
>>> list1.pop()
```

```
'a'
>>> list2.pop(1)
2
>>> print(list1)
['a', 'b', 'a', 'c']
>>> print(list2)
[1, 3]
```

remove()
තෝරාගත් අයිතමයක් List එකකින් ඉවත් කිරීම සඳහා භාවිතා කරයි.

```
>>> animals=['cat','dog','rat','bat']
>>> animals.remove('dog')
>>> print(animals)
['cat', 'rat', 'bat']
```

reverse()
List එකක අයිතම දකුණේ සිට වමට දැක්වීම සඳහා භාවිතා කරයි.

```
>>> list1=[10,2,5,15,12,1,7]
>>> list1.reverse()
>>> print(list1)
[7, 1, 12, 15, 5, 2, 10]
```

sort()
List එකක ඇති අයිතම ආරෝහන හෝ අවරෝහන පිළිවෙලකට සකස් කිරීම සඳහා භාවිතා කරයි.

ආරෝහන පිළිවෙලට සකස් කිරීම.

```
>>> list1=[10,2,5,15,12,1,7]
>>> list1.sort()
>>> print(list1)
[1, 2, 5, 7, 10, 12, 15]
```

අවරෝහන පිළිවෙලට සකස් කිරීම.

```
>>> list1=[10,2,5,15,12,1,7]
>>> list1.sort(reverse=True)
>>> print(list1)
[15, 12, 10, 7, 5, 2, 1]
```

Tuple

මෙම වර්ගයේ දත්තයක් කොමා වලින් වෙන්කරනු ලබන පටිපාටිගත අගයන්ගෙන් සමන්විතවේ. Tuple දත්ත ව්‍යුහයක් සෑදීම සඳහා () භාවිතා කරයි. මෙය immutable හා iterable දත්ත වර්ගයට අයත් වේ. Tuple එකක් නිර්මාණය කල පසු එහි අඩංගු දත්තයන් කිසිවිටක වෙනස් කල නොහැක. tuple එකක දත්ත අයිතම list එකකදී මෙන් පරිශීලනය කලහැක.

```
>>> school=('rcc','tmv','gmv','pmv')
>>> animal=('cat','dog','bat','fish')
>>> num=(5,3,2,1,4)
>>> print(num)
(5, 3, 2, 1, 4)
```

```
>>> print(school)
('rcc', 'tmv', 'gmv', 'pmv')
>>> print(school[0])
rcc
>>> print(school[:2])
('rcc', 'tmv')
>>> print(school[2:])
('gmv', 'pmv')
>>> print(school[1:3])
('tmv', 'gmv')
>>> print(school[-1])
pmv
>>> print(len(school))
4
>>> print(animal+num)
('cat', 'dog', 'bat', 'fish', 5, 3, 2, 1, 4)
>>> print(5 in num)
True
>>> print(school.index('gmv'))
2
>>> num=(5,3,2,1,4,3,1,3,3)
>>> print(num.count(3))
4
```

Dictionary

Dictionary වර්ගයේ දත්තයක් කොමා වලින් වෙන් කරනු ලබන අතර මේ සඳහා {} වරහන් භාවිතා කරයි. පටිපාටිගත නොවන යතුරඃඅගය(Key:Value) යුගල කුලකයකින් යුක්ත වේ. මෙය mutable දත්ත වර්ගයට අයත් වේ. මෙහිදී යතුර ලෙස අගයක් හෝ අක්ෂර භාවිතා කල හැක. Dictionary තුල ඇති දත්තයන් වෙනස් කල හැක.

```
>>> dict1={1:'aruna',2:'kamal',3:'saman'}
>>> dict2={'a':'bimal','b':'tamal','c':'nimal'}
>>> dict3={1:'sun',2:'mon',3:'tue'}
>>> print(dict1)
{1: 'aruna', 2: 'kamal', 3: 'saman'}
>>> print(dict2)
{'a': 'bimal', 'b': 'tamal', 'c': 'nimal'}
>>> print(dict3)
{1: 'sun', 2: 'mon', 3: 'tue'}
```

එකම key එක දෙවරක් භාවිතා කළ විට පෙර අගය මත පසුව එක් කරනු ලබන අගය මත ලියවීම සිදුවේ.

```
>>> animals={1:'cat',2:'dog',3:'bat',1:'fish'}
>>> print(animals)
{1: 'fish', 2: 'dog', 3: 'bat'}

>>> animals={1:'cat',2:'dog',3:'bat',1:'fish'}
>>> print(animals[2])
Dog
>>> dict2={'a':'bimal','b':'tamal','c':'nimal'}
>>> print(dict2['a'])
```

Bimal

Dictionary වල වටිනාකම් යාවත්කාලීන කිරීම

```
>>> dict2={'a':'bimal','b':'tamal','c':'nimal'}
>>> dict2['a']='sunil'
>>> dict2['c']='tharidu'
>>> print(dict2)
{'a': 'sunil', 'b': 'tamal', 'c': 'tharidu'}

>>> dict2={'a':'bimal','b':'tamal','c':'nimal'}
>>> print(dict2.items())
dict_items([('a', 'bimal'), ('b', 'tamal'), ('c', 'nimal')])

>>> print(dict2.keys())
dict_keys(['a', 'b', 'c'])
>>> print(dict2.values())
dict_values(['bimal', 'tamal', 'nimal'])

>>> dict3=dict1.copy()
>>> print(dict3)
{1: 'aruna', 2: 'kamal', 3: 'saman'}
```

Dictionary වල සුචි නොමැති අතර යතුරු පමනක් ඇත.

Dictionary තුල ඇති එක් එක් යතුරු ඉවත් කිරීම.

```
>>> dict2={'a':'bimal','b':'tamal','c':'nimal'}
>>> del dict2['a']
>>> print(dict2)
{'b': 'tamal', 'c': 'nimal'}
```

Dictionary තුල ඇති සියලු දත්ත ඉවත් කිරීම.

```
>>> dict2={'a':'bimal','b':'tamal','c':'nimal'}
>>> dict2.clear()
>>> print(dict2)
{}
```

Dictionary සම්පූර්ණයෙන්ම ඉවත් කිරීම.

```
>>> dict2={'a':'bimal','b':'tamal','c':'nimal'}
>>> del dict2
>>> print(dict2)
```

Set

පයිතන් භාෂාවේ භාවිතා වන තවත් දත්ත වර්ගයකි. මෙය පටිපාටිගත නොවන දත්ත ප්‍රරූපයක් වන අතර එකම දත්තය නැවැත නැවැත භාවිතා කල නොහැක. මෙහි ඇති ප්‍රධාන වාසිය නම් අවශ්‍ය දත්තයක් පහසුවෙන් සොයාගැනීමට හැකිවීම හා එකම දත්තය නැවැත නැවැත තිබීම වලකා තිබීමයි. set දැක්වීම සඳහා {} භාවිතා කරයි. set දත්ත වර්ගය ප්‍රධාන ආකාර දෙකකි.

1. set (mutable)
2. Frozenset (immutable)

```
>>> x=[5,6,7,5,9,8]
>>> y=set(x)
>>> print(y)
{5, 6, 7, 8, 9}
```

X නැමැති list එකෙහි 5 ඉලක්කම් දෙකක් ඇත. නමුත් එය set එකක් බවට පත් කල පසු අනුපිටපත් ඉවත්වී ඇති බව පෙනේ. තවද Set {} වරහන මගින් නිරූපනය වේ.

```
>>> fruits=['mango','banana','apple','mango']
>>> x=set(fruits)
>>> x
{'apple', 'banana', 'mango'}

>>> animals={'cat','dog','bat','rat','fish','cat'}
>>> print(animals)
{'fish', 'cat', 'bat', 'rat', 'dog'}
>>> 'cat' in animals
True
>>> 'lion' in animals
False
>>> 'lion' not in animals
True
>>> 'lion' is animals
False
>>> 'lion' is not animals
True
```

සාමාන්‍ය ගණිතයේදී භාවිතා කරනු ලබන කුලක (set) වල භාවිතා කරන ගණිතකර්ම මෙහිදීද සිදුකරගත හැක.

1. Union
2. Intersection
3. Difference
4. Symmetric difference

```
>>> x=('aruna')
>>> y=('waruna')
>>> x=set('aruna')
>>> y=set('waruna')
>>> x
{'a', 'r', 'n', 'u'}
>>> y
{'r', 'w', 'a', 'n', 'u'}

>>> y-x # letters in y but not in x
{'w'}

>>> x|y #letters in either x or y
```

{'a', 'r', 'w', 'n', 'u'}

>>> x&y # letters in both x and y
{'a', 'r', 'n', 'u'}

>>> x^y # letters in x or y but not both
{'w'}

පයිතන් භාෂාවේ භාවිතා වන ගණිත කර්ම **(Python Basic Operators)**

1. අංක ගණිතමය කාරක (Arithmetic Operators)
2. සංසන්දනාත්මක කාරක (Comparison Operators)
3. තාර්කික කාරක (Logical Operators)
4. ආදේශන කාරක (Assignment Operators)
5. බිටු අනුසාරිත කාරක (Bitwise Operators)

අංක ගණිතමය කාරක (Arithmetic Operators)

කාරකය	විස්තරය	නිදසුන (x=5, y=2 නම්)
+	එකතු කිරීම	x+y=7
-	අඩු කිරීම	x-y=3
*	ගුණ කිරීම	x*y=10
/	බෙදීම	x/y=2.5
%	ශේෂය (mod)	x%y=1
**	බලය	x**y=25
//	පූර්ණ සංඛ්‍යාවකට බෙදීම (Floor Division)	x//y=2

```
a = 21
b = 10
c = 0
c = a + b
print ("Value of c is ", c)
c = a - b
print ("Value of c is ", c )
c = a * b
print ("Value of c is ", c )
c = a / b
print ("Value of c is ", c )
c = a % b
print ("Value of c is ", c)
a = 2
b = 3
c = a**b
print ("Value of c is ", c)
a = 10
b = 5
c = a//b
print ("Value of c is ", c)
```

```
Value of c is  31
Value of c is  11
Value of c is  210
Value of c is  2.1
Value of c is  1
Value of c is  8
Value of c is  2
```

සංසන්දනාත්මක කාරක (Comparison Operators)

අගයන් දෙකක් හෝ කිහිපයක් සංසන්ධනය කර ප්‍රතිදානය True හෝ False ලෙස ලබාදෙයි.

කාරකය	විස්තරය	නිදසුන (x=5, y=2 නම්)
==	සමානයි	x=y →False
!=	අසමානයි	x!=y →True
>	විශාලයි	x>y →True
<	කුඩයි	x<y → False
>=	විශාලයි හෝ සමානයි	x>=y → True
<=	බලය	x <=y →False

```
a = 21
b = 10
c = 0
if (a == b):
  print ("a is equal to b")
else:
  print ("a is not equal to b")
if (a! = b):
  print ("a is not equal to b")
else:
  print ("a is equal to b")
if (a < b):
  print ("a is less than b")
else:
  print ("a is not less than b")
if (a > b):
  print ("a is greater than b")
else:
  print ("a is not greater than b")
a = 5;
b = 20;
if (a <= b):
  print ("a is either less than or equal to b")
else:
  print ("a is neither less than nor equal to b")
if (b >= a):
  print ("b is either greater than or equal to b")
else:
  print ("b is neither greater than nor equal to b")
```

```
a is not equal to b
a is not equal to b
a is not less than b
a is greater than b
a is either less than or equal to  b
b is either greater than  or equal to b
```

තාර්කික කාරක (Logical Operators)

and, or සහ not යන තාර්කික මෙහෙයවන භාවිතා වන කාරක මෙයට අයත්වේ.

කාරකය	විස්තරය	නිදසුන (x=5, y=2 නම්)
and	තාර්කික and	x=y and x>y →false x!=y and x>y →true
or	තාර්කික or	x!=y or x<y →true x>y or x!=y →true
not	තාර්කික not	not(x>y) →false not(x!=y) →true

```
a = 10
b = 20
c = 0
if (a and b):
   print ("a and b are true")
else:
   print ("Either a is not true or b is not true")
if (a or b):
   print ("Either a is true or b is true or both are true")
else:
   print ("Neither a is true nor b is true")
a = 0
if (a and b):
   print ("a and b are true")
else:
   print ("Either a is not true or b is not true")
if (a or b):
   print ("Either a is true or b is true or both are true")
else:
   print ("Neither a is true nor b is true")
if not(a and b):
   print ("Either a is not true or b is not true")
else:
   print ("a and b are true")
```

```
a and b are true
Either a is true or b is true or both are true
Either a is not true or b is not true
Either a is true or b is true or both are true
Either a is not true or b is not true
```

ආදේශන කාරක (Assignment Operators)

කිසියම් විචල්‍යයක් සඳහා අගයක් ආදේශ කිරීමේදී යොදාගත හැකි මෙහෙයවනයකි. මෙහිදී ආදේශවන විචල්‍යය operand ලෙස හඳුන්වයි.

කාරකය	විස්තරය	නිදසුන (x=5, y=2 නම්)
=	දකුණු පස අගය වම්පස අගයට ආදේශ කිරීම	y=x →y=5
+=	දකුණු පස ඇති අගය වම්පස අගයට එකතුකර ප්‍රතිඵලය නැවැත වම්පසට ආදේශ කිරීම	x+=y →x=7
-=	වම්පස අගයෙන් දකුණුපස අගය අඩුකර ප්‍රතිඵලය නැවැත වම්පසට ආදේශ කිරීම	x-=y →x=3
*=	දකුණු පස අගය වම්පස අගයෙන් ගුණකර ප්‍රතිඵලය වම්පසට ආදේශ කිරීම	x*=y → x=10

/=	වම්පස අගය දකුණු පස අගයෙන් බෙදා ප්‍රතිඵලය වම්පසට ආදේශ කිරීම	x/=y →x=2.5
%=	වම්පස අගය දකුණු පස අගයෙන් බෙදා ශේෂය වම්පසට ආදේශ කිරීම	x%=y →x=1
**=	වම්පස අගයේ බලය දකුණුපස අගය ලෙස ගෙන පිළිතුර නැවැත වම්පසට ආදේශ කිරීම.	x**y →x=25
//=	වම්පස අගය දකුණු පස අගයෙන් බෙදා පිළිතුර පූර්ණ සංඛ්‍යාවක් ලෙස වම්පසට ආදේශ කිරීම	x//y →x=2

```
a = 8
b = 3
c = 0
c = a + b
print ("Value of c is ", c)
c += a
print ("Value of c is ", c )
c *= a
print ("Value of c is ", c )
c /= a
print ("Value of c is ", c )
c  = 2
c %= a
print ("Value of c is ", c)
c **= a
print ("Value of c is ", c)
c //= a
print ("Value of c is ", c)
```

```
Value of c is  11
Value of c is  19
Value of c is  152
Value of c is  19.0
LValue of c is  2
Value of c is  256
Value of c is  32
```

බිටු අනුසාරිත කාරක (Bitwise Operators)

ද්විමය සංඛ්‍යාමත පදනම් ගණනය කිරීම් සඳහා භාවිතා කරයි. යම් අගයන් දෙකක් විචල්‍යයන් භාවිතයෙන් ඇතුලත් කල පසු එම සංඛ්‍යාවන් වලට අදාළ ද්විමය අගයන් ලබාගෙන ගණනය කිරීම් සිදුකර ප්‍රතිඵලය දශමය ආකාරයෙන් ලබාගත හැක.

කාරකය	විස්තරය	නිදසුන (x=15, y=11 නම්)
&	Binary AND	x&y → 11
|	Binary OR	x|y → 15
^	Binary XOR	x^y → 4
~	Ones Complement (පළමු අනුපූරකය)	~x →-16, ~y →-12
<<	Left shift (වමට සීරුමාරු කිරීම)	x>>2 → 3, y>>2 →2
>>	Right shift (දකුණට සීරුමාරු කිරීම)	x<<2 → 60, y>>2 →44

Binary AND

x=15 හා y=11 ලෙසගමු.

x	0	0	0	0	1	1	1	1
y	0	0	0	0	1	0	1	1
x&y	0	0	0	0	1	0	1	1
							දශමය අගය	11

Binary OR

x=15 හා y=11 ලෙසගමු.

x	0	0	0	0	1	1	1	1

y	0	0	0	0	1	0	1	1
x|y	0	0	0	0	1	1	1	1
						දශමය අගය	15	

Binary XOR

x=15 හා y=11 ලෙසගමු.

x	0	0	0	0	1	1	1	1
y	0	0	0	0	1	0	1	1
x^y	0	0	0	0	0	1	0	0
						දශමය අගය	4	

Binary ones complement

පරිගණකවලදී ධන සංඛ්‍යා සෘජුවම භාවිතා කල හැකි අතර සෘණ සංඛ්‍යා නිරූපනය කරනු ලබන්නේ දෙවන අනුපූරක ආකාරයෙනි.

දෙවන අනුපූරක සංඛ්‍යාව = සංඛ්‍යාව +1

සංඛ්‍යාව = 15 → 00001111

දෙවන අනුපූරක සංඛ්‍යාව = 00001111 +1 = 00010000

දශමය සංඛ්‍යාව = -16 (00001111 පළමු අනුපූරකයට හැරවූ විට 11110000 ලෙස ලැබේ. මෙහි Sign බිටුව 1 නිසා සෘණ සංඛ්‍යාවක් ලෙස ලියනු ලබයි.)

වමට සීරුමාරු කිරීම (Left shift)

x=15, y=11 නම්



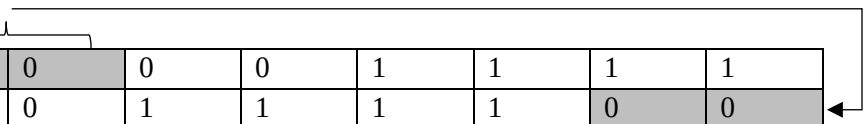
x	0	0	0	0	1	1	1	1
x<<2	0	0	0	0	0	0	1	1
						දශමය අගය	3	



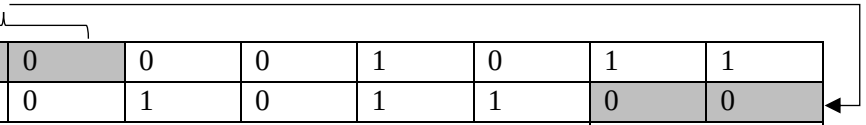
y	0	0	0	0	1	0	1	1
y>>2	0	0	0	0	0	0	1	0
						දශමය අගය	2	

දකුණට සීරුමාරු කිරීම (Right shift)

x=15, y=11 නම්



x	0	0	0	0	1	1	1	1
x<<2	0	0	1	1	1	1	0	0
						දශමය අගය	60	



y	0	0	0	0	1	0	1	1
y<<2	0	0	1	0	1	1	0	0
						දශමය අගය	44	

```
a = 60          # 60 = 0011 1100
b = 13          # 13 = 0000 1101
c = 0
c = a & b;      # 12 = 0000 1100
print ("Value of c is ", c)

c = a | b;      # 61 = 0011 1101
print ("Value of c is ", c)

c = a ^ b;      # 49 = 0011 0001
print ("Value of c is ", c)

c = ~a;         # -61 = 1100 0011
print ("Value of c is ", c)

c = a << 2;     # 240 = 1111 0000
print ("Value of c is ", c)

c = a >> 2;     # 15 = 0000 1111
print ("Value of c is ", c)
```

```
Value of c is  12
Value of c is  61
Value of c is  49
Value of c is  -61
Value of c is  240
Value of c is  15
```

Membership Operators

මෙහිදී in සහ not in කාරක භාවිතා කරයි. යම්කිසි විචල්‍යයක් තුල අන්තර්ගත දත්තයක් නම් සත්‍යය ලෙසත් අන්තර්ගත නොවන දත්තයක් නම් අසත්‍යය ලෙසත් දැක්වීම සිදුවේ. සත්‍යය බව දැක්වීමට in කාරකයද අසත්‍යය බව දැක්වීමට not in කාරකයද භාවිතා කරයි.

```
>>> a=[1,2,3,5]
>>> print (3 in a)
True
>>> print(4 in a)
False
```

Identity Operators

මෙහිදී is සහ is not කාරක භාවිතා කරයි. යම්කිසි විචල්‍යය දෙකක් සසඳා එක් විචල්‍යයක් තුල අන්තර්ගත දත්තයක් අනෙත් විචල්‍යය තුල පවතීනම් සත්‍යය (True) ලෙසත් අන්තර්ගත නොවන දත්තයක් නම් අසත්‍යය (False) ලෙසත් දැක්වීම සිදුවේ. සත්‍යය බව දැක්වීමට is කාරකයද අසත්‍යය බව දැක්වීමට is not කාරකයද භාවිතා කරයි.

```
>>> a=5
>>> b=5
>>> if a is b:
        print('a equal to b')
else:
        print('a not equalt to b')
```

Output

```
a equal to b
```

```
>>> a=5
>>> b=5
```

```
>>> if a is not b:
        print(' a not equal to b’)
else:
        print("a equal to b ')
```

Output

a equal to b

කාරක ප්‍රමුඛතාවය

පයිතන් ක්‍රමලේඛන භාෂාවේදී පහත සඳහන් කාරක ප්‍රමුඛතා අනුපිළිවෙල භාවිතා කරයි.

ප්‍රමුඛතා මට්ටම	කාරකය
1	**
2	- + ~ (unary plus and minus), complement
3	* / % //
4	+ -
5	>> <<
6	&
7	^ |
8	<= < > >=
9	== !=
10	= += -= *= /= %= //= |= &= >>= <<= **=
11	is, is not
12	in, not in
13	not, and, or

ප්‍රමුඛතාවය වැඩි

ප්‍රමුඛතාවය අඩු

```
>>> 5+2*3
11
>>> 5-2*6
-7
>>> 6+2**3
14
>>> (5+2)//2+2**4
19
>>> 5/2+3
5.5
>>> 4/2+5//2
4.0
```

```
>>> 2+3**2+2**2
15
>>> 8//2**3*2-6//4+1
2
```

Escape characters

අනුලක්ෂණය	විස්තරය
\n	Newline
\\	Backslash (\)
\’	Single quote (‘)
\’’	Double quote (“)
\a	ASCII bell
\b	ASCII Backspace
\f	ASCII Form feed
\t	Tab

```
>>> print('Ict\ninstitute')
Ict
institute
>>> print('Ict\tinstitute')
Ict	institute
>>> print('Ict\\institute')
Ict\institute
>>> print('Ict\'institute')
Ict'institute
>>> print('Ict\"institute')
Ict"institute
```

පාලන ව්‍යූහ (Control structure)

වරණය (Selection)

If statement

කොන්දේසියක් තෘප්ත වන්නේ නම් පමණක් ප්‍රකාශනය ක්‍රියාත්මක වේ.

උදාහරණ: විෂයකට ලබාගත් ලකුණු ඇතුලත් කර එය 50 හෝ 50ට වඩා වැඩි නම් සමත් ලෙස දැක්වීම.

```
marks=int(input('Enter marks: '))
if marks>=50:
    print('Pass')
```

```
Enter marks: 60
Pass
```

If …else statement

මෙහිදී කොන්දේසිය තෘප්ත වන්නේ නම් පළමු ප්‍රකාශයද නැතිනම් දෙවන ප්‍රකාශයද ක්‍රියාත්මක වේ.

උදාහරණ: විෂයකට ලබාගත් ලකුණු ඇතුලත් කර එය 50 හෝ 50ට වඩා වැඩි නම් සමත් ලෙසත් නැතිනම් අසමත් ලෙස දැක්වීම.

```
marks=int(input('Enter marks: '))
if marks>=50:
   print('Pass')
else:
   print('fail')
```

```
Enter marks: 45
fail
```

elif statement

උදාහරණ: විෂයකට ලබාගත් ලකුණු ඇතුලත් කර එය 75 හෝ 75ට වඩා වැඩි නම් 'Excellent' ලෙසත් 50 හෝ 50ට වඩා වැඩි නම් 'Good' ලෙසත් නැතිනම් 'Weak' ලෙස දැක්වීම.

```
marks=int(input('Enter marks: '))
if marks>=75:
   print('Excellent')
elif marks>=50:
   print('Good')
else:
   print('Weak')
```

```
Enter marks: 76
Excellent
```

Nested if Statement

කොන්දේසියකට පසුව කොන්දේසියක් වශයෙන් කොන්දේසි සමූහයක් යෙදෙන අවස්ථාවක භාවිතා කරයි.

උදාහරණ : සංඛ්‍යා තුනකින් විශාලතම සංඛ්‍යාව සෙවීම.

```
n1=int(input('Enter number 1 :'))
n2=int(input('Enter number 2 :'))
n3=int(input('Enter number 3 :'))
if n1>n2:
   if n1>n3:
      print('maximum number :',n1)
   else:
      print('maximum number :',n3)
elif n2>n3:
   print('maximum number :',n2)
else:
   print('maximum number :',n3)
```

```
Enter number 1 :4
Enter number 2 :5
Enter number 3 :7
maximum number : 7
```

පුනර්කරණය (Repetition /Looping)

පුනර්කරණය වන වාර ගණන නිශ්චිතව දන්නා විට ක්‍රමලේඛනයක් ක්‍රියාත්මක කිරීම සඳහා භාවිතා කරයි.

While loop

යම් කොන්දේසියක් සත්‍යව පවතින තුරු වගන්තියක් හෝ වගන්ති කිහිපයක් නැවැත නැවැත ක්‍රියාත්මක වීම සිදුවේ.

උදාහරණ : 1 සිට 5 දක්වා සංඛ්‍යා මුද්‍රණය කිරීම

```
x=1
while x<=5:
   print(x, end=" ")
   x=x+1
```

```
x=1
while x<=5:
   print(x)
   x=x+1
```

```
1
2
3
4
5
```

```
1 2 3 4 5
```

For loop

වගන්තියක් හෝ කිහිපයක් නිශ්චිත වාර ගණනක් පුනර්කරණය කිරීමට භාවිතා වේ. මෙය යොදාගත හැක්කේ අවස්ථා දෙකකදී පමනි.

1. යම්කිසි සංඛ්‍යා පරාසයක් තුල පුනර්කරණය සඳහා
2. දත්ත ලැයිස්තුවක් භාවිතා කිරීම සඳහා

```
>>> for x in range(5):
        print(x)
0
1
2
3
4
>>> for x in range(1,5):
        print(x)
1
2
3
4


>>> for x in range(1,5,2):
        print(x)
1
3

>>> city='polonnaruwa'
>>> for i in city:
        print(i)
p
o
l
o
n
n
a
r
u
w
a


>>> x=['a','aruna','ict',5,2]
>>> for i in x:
```

```
        print(i)
a
aruna
ict
5
2
```

Nested loop

පුනර්ණයක් තුල තවත් පුනර්කරණයක් යෙදෙන විට භාවිතා කරයි.

උදාහරණ : 1 සිට 20 දක්වා ඇති ප්‍රථමක සංඛ්‍යා ලබාගැනීම

```
i = 2
while(i < 20):
  j = 2
  while(j <= (i/j)):
    if not(i%j): break
    j = j + 1
  if (j > i/j) :
     print (i, " is prime")
  i = i + 1
```

```
2  is prime
3  is prime
5  is prime
7  is prime
11  is prime
13  is prime
17  is prime
19  is prime
```

උදාහරණ : 2 සහ 3 ගුණන වගු ලබාගැනීම.

```
for i in range (2,4):
   print (i, 'multiplication table')
   print ('-------------------------')
   for x in range(1,13):
      print(i,'*',x,'=',i*x)
```

```
2 multiplication table
-----------------------
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10
2 * 6 = 12
2 * 7 = 14
2 * 8 = 16
2 * 9 = 18
2 * 10 = 20
2 * 11 = 22
2 * 12 = 24
3 multiplication table
-----------------------
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
3 * 10 = 30
3 * 11 = 33
3 * 12 = 36
```

පුනර්කරණයක් තුල වරණයක් යෙදීම

උදාහරණ : 1 සිට 15 දක්වා ඇති ඔත්තේ සහ ඉරට්ටේ සංඛ්‍යා වෙන වෙනම ප්‍රතිදානය කිරීම.

```
for i in range(16):
    if i%2==0:
        print(i,'is an even number')
    else:
        print(i,'is an odd number')
```

```
0 is an even number
1 is an odd number
2 is an even number
3 is an odd number
4 is an even number
5 is an odd number
6 is an even number
7 is an odd number
8 is an even number
9 is an odd number
10 is an even number
11 is an odd number
12 is an even number
13 is an odd number
14 is an even number
15 is an odd number
```

පුනර්කරණ පාලන වගන්ති (Loop control statements)

Break Statement

පුනර්කරණයක් ක්‍රියාත්මක වෙමින් පවතින අවස්ථාවක අතරමගින් පුනර්කරණය නතර කර ගැනීම සඳහා යොදාගත හැක.

උදාහරණ : 1 සිට 5 දක්වා සංඛ්‍යා මුද්‍රණය වන ක්‍රමලේඛයක 3වන සංඛ්‍යාවේදී ක්‍රමලේඛය නතර කිරීම.

```
x=1
while x<=5:
    if x==3:
        break
    print(x)
    x=x+1
```

```
1
2
```

උදාහරණ : polonnaruwa යන වචනයේ n අකුරේදී ක්‍රමලේඛනය නතර කිරීම.

```
for i in 'polonnaruwa':
    if i=='n':
        break
    print(i)
```

```
p
o
l
o
```

Continue Statement

පුනර්කරණයක සමහර ප්‍රකාශන අතහැර ඊලග ප්‍රකාශනයට යාම සිදුවේ.

උදාහරණ : 1 සිට 5 දක්වා සංඛ්‍යා මුද්‍රණය වන ක්‍රමලේඛයක 3වන සංඛ්‍යාව අතහැර ක්‍රමලේඛය ක්‍රියාත්මක කිරීම.

```
for i in range(1,6):
    if i==3:
        continue
    print(i)
```

```
1
2
4
5
```

උදාහරණ : polonnaruwa යන වචනයේ n අකුරු අතහැර ඊළග අකුරු මුද්‍රනය කිරීම.

```
for i in 'polonnaruwa':
    if i=='n':
        continue
    print(i)
```

```
p
o
l
o
a
r
u
w
a
```

Pass Statement

පුනර්කරණයක් ක්‍රියාත්මකවීමේදී අතරමැද යම්කිසි වාක්‍යයක් එක් කළ යුතු නම් මෙය භාවිතා කරයි.

උදාහරණ : 1 සිට 5 දක්වා සංඛ්‍යා මුද්‍රණය වන ක්‍රමලේඛයක 3වන සංඛ්‍යාවට පෙර ict institute ලෙස එක්කර ක්‍රමලේඛය ක්‍රියාත්මක කිරීම.

```
for i in range(1,6):
    if i==3:
        pass
        print('ict institute')
    print(i)
```

```
1
2
ict institute
3
4
5
```

පයිතන් ශ්‍රිත **(Python function)**

ශ්‍රිතයක් යනු අවශ්‍ය කාර්යයක් හෝ කාර්යයන් කිහිපයක් කරගැනීම සඳහා භාවිතා කරනු ලබන ප්‍රකාශ අනුක්‍රමයකි. ශ්‍රිතයක අදාල කාර්යය හෝ කාර්යයන් ශ්‍රිතයේ නිර්වචනය මගින් දැක්වේ. ශ්‍රිත මගින් ක්‍රමලේඛයක ඇති කොටසක් තනි ඒකකයක් ලෙස උපුටාගෙන එම කොටස එම ක්‍රලේඛයේම වෙනත් ක්‍රමලේඛයක නැවැත නැවැත භාවිතයට ගතහැකි පරිදි පහසුකම් ලබාදේ. එමගින් ක්‍රමලේඛයක් තුල එකම කේතය නැවැත නැවැත ලිවීමට අවශ්‍යය නොවේ. ශ්‍රිත ප්‍රධාන අකාර දෙකකි.ඛ.

1. කලින් සකස් කරන ලද ශ්‍රිත (Standard Function)
2. පරිශීලකයාට අවශ්‍යය පරිදි සකසන ලද ශ්‍රිත (User Define Function)

විචල්‍යයන් හා පරාමිතීන් අතර වෙනස (Difference between variable and parameters)
විචල්‍යයක් යනු මතකයේ දත්තයක් ගබඩා කර තබාගන්නා ස්ථානයක් වන අතර එම දත්ත ක්‍රියාපටිපාටියකට භාරදීම සඳහා පරාමිතීන් උදව් වේ.

පරාමිති වර්ග
ආදාන පරාමිති (Input Parameters)
ආදාන හෝ ප්‍රතිදාන පරාමිතීන් (Input/output parameters)

ශ්‍රිතයක් නිර්වචනය කිරීම (Define a function)

පයිතන් භාෂාවේදී ශ්‍රිතයක් නිර්වචනය කිරීම සඳහා ‘def’ යන මූලපදය භාවිතා කරයි. විධිමත් පරාමිති ලැයිස්තුවක් වරහන් තුල ඇතුලත් කරනු ලබන අතර (:) සංකේතය මගින් අවසන් කරයි. ශ්‍රිතයේ නම හඳුන්වනයක් බැවින් හඳුන්වන නාමා ලිවීමේදී භාවිතා කරනු ලබන නීති මේ සඳහා වලංගු වේ. පරාමිති ලැයිස්තුව පරාමිති එකක් හෝ කිහිපයක් තිබිය හැක. පරාමිති නොමැති ශ්‍රිත ද සකස් කල හැක. ශ්‍රිතයක් ක්‍රියාත්මක වන්නේ කැඳවූ විටය. ක්‍රියාවට නංවන ශ්‍රිතය සහ එහි පරාමිතීන් මෙහිදී ලබාදිය යුතුය. ශ්‍රිතයකට තනිව ක්‍රියාත්මක විය වෙනාහැකි අතර එය ක්‍රියාත්මක වන්නේ ප්‍රධාන ක්‍රමලේඛනයක් තුලය. පහත අකාරයට ශ්‍රිතය අර්ථදැක්විය හැක.

```
def function_name(arg1,arg2):
        statment1
        statment2
```

පරාමිතීන් නොමැතිව ශ්‍රිත ගොඩනැගීම

```
def display():
        print()
```

```
#function
def display():
    print('hi')
#program
display()#calling the function
print('Student')
display()#calling the function
print('Aruna')
```

```
hi
Student
hi
Aruna
```

return statement

ශ්‍රිතයක් කැඳවන සෑම අවස්ථාවකම අගයක් මුදාහරී. එය return යන පයිතන් මූලපදය මඟින් ප්‍රධාන ක්‍රමලේඛයට අගයක් මුදාහැරීම සිදුකල හැකිය. මුදාහරින ලද වගන්තියේ ශ්‍රිතය තුල අගයක් නොමැති විට None යන අගය ලබාදේ.

ගැටලුව 1
සංඛ්‍යා දෙකක් ආදානය කර ඒවායේ එකතුව ලබාගැනීමට ක්‍රමලේඛනයක් ශ්‍රිත භාවිතයෙන් ලිවීම.

```
#function
def sum(x,y):
   print('Total is :',x+y)
#program
x=int(input('Enter a number 1: '))
y=int(input('Enter a number 2: '))
sum(x,y)
```

```
Enter a number 1: 5
Enter a number 2: 6
Total is : 11
```

```
#function
def sum(x,y):
   return x+y
#program
x=int(input('Enter a number 1: '))
y=int(input('Enter a number 2: '))
print('Total is :',sum(x,y))
```

ගැටලුව 2
පළමු නම, දෙවන නම සහ අවසාන නම ඇතුලත් කර මුලකුරු සමඟ අවසාන නම දැක්වීම සඳහා පයිතන් ක්‍රමලේඛනයක් ලියන්න.

```
#function
def name(fn,sn,ln):
   print(fn[0],'.',sn[0],'.',ln)
#program
fn=input('Enter first name: ')
sn=input('Enter second name: ')
ln=input('Enter last name: ')
name(fn,sn,ln)
```

```
Enter first name: Aruna
Enter second name: Kumara
Enter last name: Herath
A . K . Herath
```

ගැටලුව 3
පළමු නම, දෙවන නම, අවසාන නම, වාසගම, ස්ත්‍රී/ පුරුෂ බව සහ විවාහක අවිවාහක බව ඇතුලත් කර Mr, Mrs. හෝ Miss. ඇතුලත් මුලකුරු සමඟ වාසගම දැක්වීම සඳහා පයිතන් ක්‍රමලේඛනයක් ලියන්න.

```
#function
def name(fn,sn,ln,su,g,c):
   if g=='male':
      s='Mr'
   elif g=='female':
      if c=='maried':
         s='Mrs'
      else:
         s='Miss'
```

```
Enter first name: Chanaka
Enter second name: aruna
Enter last name: kumara
Enter surname: herath
Enter 'male' or 'female' :male
Enter 'married' or 'unmarried' :married
Mr . C . a . k . herath
```

```
    print(s,'.',fn[0],'.',sn[0],'.',ln[0],'.',su)
#program
fn=input('Enter first name: ')
sn=input('Enter second name: ')
ln=input('Enter last name: ')
su=input('Enter surname: ')
g=input("Enter 'male' or 'female' :")
c=input("Enter 'married' or 'unmarried' :")
name(fn,sn,ln,su,g,c)
```

උදාහරණ 1

```
def changeme( mylist ):
   "This changes a passed list into this function"
   mylist = [1,2,3,4]; # This would assign new reference in mylist
   print ("Values inside the function: ", mylist)
   return

# Now you can call changeme function
mylist = [10,20,30];
changeme( mylist );
print ("Values outside the function: ", mylist)
```

Output

Values inside the function: [1, 2, 3, 4]
Values outside the function: [10, 20, 30]

උදාහරණ 2

```
# Function definition is here
def printinfo( name, age ):
   "This prints a passed info into this function"
   print ("Name: ", name)
   print ("Age :", age)
   return;
# Now you can call printinfo function
printinfo( age=38, name="aruna" );
```

Output

Name: aruna
Age : 38

උදාහරණ 3

```
# Function definition is here
def printinfo( name, age = 17 ):
   "This prints a passed info into this function"
   print ("Name: ", name)
   print ("Age :", age)
```

```
   return;

# Now you can call printinfo function
printinfo( age=18, name="kasun" );
printinfo( name="supun" );
```

Output

```
Name:  kasun
Age : 18
Name:  supun
Age : 17
```

උදාහරණ 4

```
# Function definition is here
sum = lambda arg1, arg2: arg1 + arg2;

# Now you can call sum as a function
print ("Value of total : ", sum( 10, 20 ))
print ("Value of total : ", sum( 20, 20 ))
```

Output

```
Value of total :  30
Value of total :  40
```

ශ්‍රිත විස්තාරයන් (function arguments)
පයිතන් පරිගණක භාෂාවේදී ශ්‍රිත විස්තාරයන් කිහිපයක් භාවිතා කරයි. ඒවා පයිතන් ශ්‍රිත වල ඇති විශේෂාංග ලෙසද හැඳින්විය හැක.

1. Default arguments
2. Keyword arguments
3. Required arguments
4. Variable arguments

Default arguments
ශ්‍රිත නිරවචනයේදී ස්වයං පැවැරූ අගයන් භාවිතා කළ හැකිය. යම් අගයක් ශ්‍රිතය කැඳවීමේදී නොමැති නම් ශ්‍රිතය තුල අන්තරගත ස්වයං පැවැරූ අගය කැඳවීම සිදුකරයි.

```
#default arguments
#function
def myself(name='kamal',age=38,address='kaduruwela'):
   print('Name is : ',name)
   print('Age is :',age)
   print('Address is :',address)
   return
#program
myself(name='aruna',address='polonnaruwa')
```

Output

```
Name is :  aruna
Age is : 38
Address is : polonnaruwa
Name is :  herath
Age is : 35
Address is : kaduruwela
```

```
myself(name='herath',age=35)
```

Keyword arguments

මූල පදයක් ශ්‍රිත කැඳවීම සඳහා භාවිතා කරයි නම් එය පරාමිති නාමයන් හඳුනා ගනී.

```
#keyword arguments
#function
def myself(str):
   print(str)
   return
#program
myself(str='Aruna herath')
myself(str='Nethumsa herath')
```

Output

```
Aruna herath
Nethumsa herath
```

Required arguments

ශ්‍රිතයක් තුල විස්තාරයන් භාවිතා කිරීමේදී එය පිළිවෙලකට යැවිය යුතුය. එමෙන්ම ශ්‍රිතය කැඳවීමේ අඩුම තරමින් එක් විස්තාරයක් හෝ පැවැරිය යුතුය. එසේ නොමැති නම් දෝෂ පණිවිඩයක් ලබාදෙයි.

```
#keyword arguments
#function
def myself(str):
   print(str='aruna')
   return
#program
myself()
```

Output

```
Error
```

Variable length arguments

යම් ශ්‍රිතයක් නිර්වචනයට වඩා වැඩි විස්තාරයන් සඳහා ශ්‍රිත සැකසීම සිදුකරයි නම් ඒවා මෙලෙස හඳුන්වයි. විශේෂිත (*) සලකුණක් අදාල විචල්‍යයට පෙර යොදාගැනේ. එමගින් මූලපද නොවන විචල්‍යයන් සඳහා අගයයන් රඳවා තබාගනී.

```
#Variable length arguments
#function
def my_function(*kids):
  print("The youngest child is " + kids[1])

#program
my_function("Suranga", "Sandun", "Thusitha")
```

Output

```
The youngest child is Sandun
```

```
#Variable length arguments
#function
def my_sum(*args):
   return sum(args)
#program
print(my_sum(1, 3, 5, 7,9))
print(my_sum(2,4,6,8,10))
```

Output

```
25
30
```

<u>විචල්‍යයන්ගේ සීමාවන් ඇඳීම</u>

පයිතන් භාෂාව විචල්‍යය සඳහා අගයන් ලබාදෙන ස්ථානය එම විචල්‍යය් සීමාව (scope) තීරණය කරනු ලබයි. ශ්‍රිතයක් තුල විචල්‍යයන් ප්‍රධාන ආකාර දෙකකට වර්ග කල හැකිය.

1. ස්ථානීය විචල්‍යය (Local Variable)
2. ගෝලීය විචල්‍යයන් (Global Variable)

ස්ථානීය විචල්‍යය (Local Variable)

ශ්‍රිතය තුල විචල්‍යයක් ආරම්භ වේනම් එය ස්ථානීය විචල්‍යයක් ලෙස හැඳින්වේ. මෙම විචල්‍යය භාවිතා කල හැක්කේ ශ්‍රිතය තුලදී පමණි.

ගෝලීය විචල්‍යයන් (Global Variable)

ගෝලීය විචල්‍යයක් ශ්‍රිතයක් තුලදී හා ඉන් පිටතදී භාවිතා කල හැකිය. එය ක්‍රමලේකනයේ ඕනෑම ස්ථානයක භාවිතා කල හැකිය. ශ්‍රිතයක් තුලදී **ගෝලීය විචල්‍යයක් සකස් කර ගැනීම සඳහා ශ්‍රිතය තුල** global **ලෙස යොදා විචල්‍යය සකස් කරගත යුතුය.**

```
x=5 #global variable
#function
def func1():
   x=3 #local variable
   print(x)
#program
func1()
```

Output
3

```
x=5 #global variable
#function
def func1():
   x=3 #local variable
   print(x)
#program
func1()
print(x)
```

Output
3
5

```
i=5 #global variable
#function
def func1():
   x=3 #local variable
   print(x)
#program
func1()
print(x)
```

Output
3
error

```
i=5 #global variable
#function
def func1():
   global x
   x=3 #global variable
   print(x)
#program
func1()
print(x)
```

Output
3
3

පයිතන් භාෂාවේදී ගොනු හැසිරවීම (File handling in python)

ක්‍රමලේඛණයක් තුලින් ලබාදෙන දත්ත ප්‍රතිදාන ලෙස දෙයාකාරයකට සිදුකල හැකිය.

1. මුද්‍රණ යන්ත්‍රයක් භාවිතයෙන් මුද්‍රණය කරගැනීම.
2. ගොනු හෝ දත්ත පාදක භාවිතා කර ආවයන උපාංගයක් තුල ගබඩා කිරීම.

ගොනු සඳහා සිදුකල හැකි මෙහෙයුම්

1. ගොනුවක ඇති දත්ත කියවීම (Read)
2. ගොනුවක දත්ත තැන්පත් කිරීම (Write)
3. ගොනුවක අවසානයට දත්ත එකතු කිරීම (Append)
4. ගොනුවක ඇති දත්ත යාවත්කාලීන කිරීම (Update)

ගොනු භාවිතයේදී සිදුකල යුතු ප්‍රධාන කාර්යයන් තුනක් ඇත.

1. ගොනුවට සම්බන්ධතාවයක් විවෘත කිරීම
2. ගොනුවේ ඇති දත්ත කියවීම හෝ ගොනුවට දත්ත එක් කිරීම
3. ගොනුව වැසීම

ගොනුවට සම්බන්ධතාවයක් විවෘත කිරීම

open(“file_name”, ‘mode’)

ප්‍රවේශ ආකාර (Access mode)	විස්තරය (Description)
‘r’	ගොනුවේ ඇති දත්ත කියවීම පමණයි
‘w’	ගොනුවට දත්ත ලිවීම පමණයි.
‘a’	ගොනුවේ අගට දත්ත එක් කිරීම.
‘r+’	ගොනුවේ ඇති දත්ත කියවීම සහ ලිවීම.
‘w+’	ගොනුවට දත්ත ලිවීම සහ කියවීම.

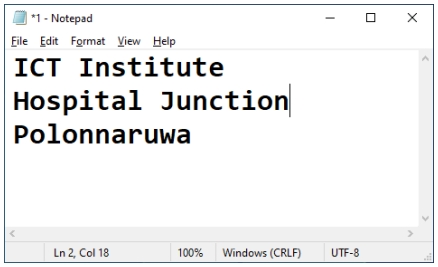
```
x=open('1.txt','r')
x.close()
```

ඉහත පයිතන් ගොනුව සුරකින ලද ස්ථානයේ 1.txt ලෙස ගොනුවක් සකස් වී ඇති බව දැකිය හැක.

ගොනුව ඇති සියළුම දත්ත කියවීම.

මේ සඳහා පහත ආකාරයේ ගොනුවක් සකස් කර එය පයිතන් ගොනුව සුරකින ලද ස්ථානයේම සුරකින්න.

1.txt ගොනුව




```
x=open('1.txt','r')
data=x.read()
print(data)
x.close()
```

Output

ICT Institute
Hospital Junction
Polonnaruwa

ගොනුව ඇති අකුරු කිහිපයක් කියවීම.

```
x=open('1.txt','r')
data=x.read(3)
print(data)
x.close()
```

Output

ICT

අකුරු තුනක් පමණක් මුද්‍රණය වී ඇත.

ගොනුව ඇති එක් පේළියක් පමණක් කියවීම.

Output

ICT Institute

පළමු පේළිය පමණක් මුද්‍රණය වී ඇත.

```
x=open('1.txt','r')
data=x.readline()
print(data)
x.close()
```

පේලියෙන් පේලිය ගොනුවක ඇති සියළුම පේලි කියවීම.

1 ක්‍රමය

```
x=open('1.txt','r')
data=x.readline()
while data:
    print(data)
    data=x.readline()
x.close()
```

Output

ICT Institute

Hospital Junction

Polonnaruwa

2 ක්‍රමය

```
x=open('1.txt','r')
for i in x:
    print(i)
x.close()
```

Output

ICT Institute

Hospital Junction

Polonnaruwa

```
x=open('1.txt','r')
data=x.readlines()
print(data)
x.close()
```

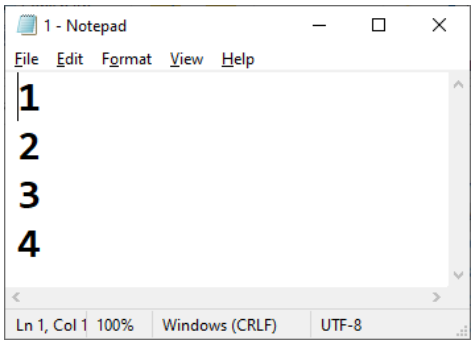
Output

['ICT Institute\n', 'Hospital Junction\n', 'Polonnaruwa']

ලිස්ට් එකක් තුල සියළුම පේළි දැක්වේ.

ගැටලුව 1

පහත ගොනුවේ ඇති සංඛ්‍යා කියවා ඒවායේ එකතුව සහ සාමාන්‍යය ලබා ගැනීමට පයිතන් ක්‍රමලේඛණය




```
tot=0
x=open('1.txt','r')
data=x.readlines()
for i in data:
    no=int(i)
    tot=tot+no
avg=tot/3
print('Total is', tot)
print('Average is', round(avg,2))
```

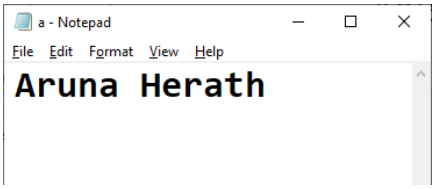
Output

Total is 10

Average is 3.33

ගොනුවකට දත්ත ලිවීම

```
x=open('a.txt','w')
x.write('Aruna Herath')
x.close()
```

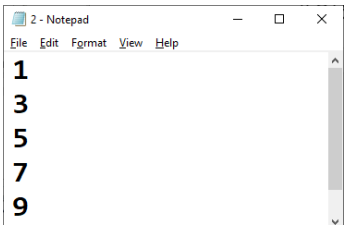


මෙම කේතය ක්‍රියාත්මක කල විට පයිතන් ගොනුව සුරකින ලද ස්ථානයේ a.txt නැමැති ගොනුවක් නිර් මානය වේ.

ගැටලුව 1

1 සිට 10 දක්වා ඇති ඔත්තේ සංඛ්‍යා ගොනුවකට ලිවීමට අදාල python ක්‍රමලේඛනය

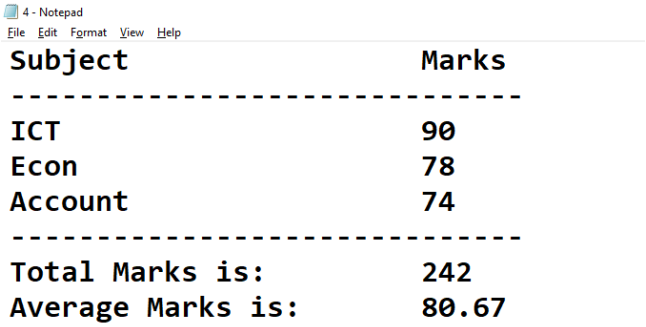
```
x=open('2.txt','w')
for i in range(1,10,2):
    b=str(i)
    x.write(b+'\n')
x.close()
```



ගැටලුව 2

විෂයන් තුනක ලකුණු ඇතුලත් කර ඒවායේ එකතුව සහ සාමාන්‍යය සෙවීම සොයා ගොනුවක ලිවීමට පයිතන් ක්‍රමලේඛනය.

```
total=0
x=open('4.txt','w')
x.write('Subject\t\t\t'+'Marks\n')
x.write('------------------------------\n')
for i in range(3):
    name=str(input('Enter Subject:'))
    marks=str(input('Enter Marks:'))
    x.write(name+'\t\t\t'+marks+'\n')
    total=total+int(marks)
average=round(total/3,2)
t=str(total)
a=str(average)
x.write('------------------------------\n')
x.write('Total Marks is: \t'+t+'\n')
x.write('Average Marks is: \t'+a+'\n')
x.close()
```



ගැටලුව 3

```
for count in range(2):
    subject=['ICT','BS','AC']
    total=0
    x=open('5.txt','a')
    name=str(input('Enter your name:'))
    x.write('----------------------------------------\n')
    x.write('Your name :'+name+'\n')
    x.write('----------------------------------------\n')
    x.write('Subject\t\t\t'+'Marks\t\tGrade\n')
    x.write('----------------------------------------\n')
    for i in range(3):
        s=subject[i]
        marks=str(input('Enter '+s+' Marks:'))
        m=int(marks)
        if m>=75:
            g='A'
        elif m>=65:
            g='B'
        elif m>=55:
            g='C'
        elif m>=35:
            g='S'
        else:
            g='W'
        x.write(s+'\t\t\t'+marks+'\t\t'+g+'\n')
        total=total+int(marks)
    average=round(total/3,2)
    t=str(total)
    a=str(average)
    x.write('----------------------------------------\n')
    x.write('Total Marks is: \t'+t+'\n')
    x.write('Average Marks is: \t'+a+'\n')
    x.write('----------------------------------------\n')
    x.close()
```

5 - Notepad
File Edit Format View Help

```
----------------------------------------
Your name :kamal
----------------------------------------
----------------------------------------
Your name :451
----------------------------------------
Subject                 Marks           Grade
----------------------------------------
ICT                     50              S
BS                      60              C
AC                      70              B
----------------------------------------
Total Marks is:         180
Average Marks is:       60.0
----------------------------------------
```

ගොනුවක නම වෙනස් කිරීම

```
import os
os.rename('1.txt','abc.txt')
```

ගොනුවක් ඉවත් කිරීම

```
import os
os.remove('a.txt')
```

ෆෝල්ඩරයක් සකස් කිරීම

```
import os
os.mkdir('aruna')
```

os විධානය මඟින් windows මෙහෙයුම් පද්ධතියේ විධාන ක්‍රියාත්මක කල හැක.

Notepad මෘදුකාංගය විවෘත කිරීම

```
import os
cmd='notepad'
os.system(cmd)
```

mspaint මෘදුකාංගය විවෘත කිරීම

```
import os
cmd='mspaint'
os.system(cmd)
```

ෆෝල්ඩරයක් ඉවත් කිරීම

```
import os
os.rmdir('aruna')
```

Python Sort

මෙහිදී කිසියම් අයිතම සමූහයක් ආරෝහන හෝ අවරෝහණ පිළිවෙලකට සක්ස් කිරීම සිදුකරනු ලබයි. මේ සඳහා භාවිතා වන ප්‍රධාන ක්‍රම දෙකකි.

1. **වරණ තේරීම** (Selection sort)
2. බුබුළු තේරීම (Bubble sort)

වරණ තේරීම (Selection sort)

මෙහිදී ලැයිස්තුවක ඇති අයිතම පිළිවෙලකට සැකසෙන තුරු තේරීම සිදුකරයි. ලැයිස්තුවේ ඇති අයිතම අතරින් විශාලතම අගය හා කුඩාම අගය නිවැරදිව ස්ථානගත කිරීම මගින් මෙම තේරීම සිදුකරයි.

```
#function
def selection_sort(L):
    for i in range(len(L)-1):
        min_index = i
        for j in range(i+1, len(L)-1):
            if L[j] < L[min_index]:
                min_index = j
        L[i], L[min_index] = L[min_index], L[i]
#program
L = [3, 1, 41, 59, 26, 53, 59]
print(L)
selection_sort(L)
print(L)
```

```
3, 1, 41, 59, 26, 53, 59
1, 3, 41, 59, 26, 53, 59
1, 3, 26, 59, 41, 53, 59
1, 3, 26, 41, 59, 53, 59
1, 3, 26, 41, 53, 59, 59
```

Output

```
[3, 1, 41, 59, 26, 53, 59]

[1, 3, 26, 41, 53, 59, 59]
```

බුබුළු තේරීම (Bubble sort)

යම්කිසි ලැයිස්තුවක් පිළිවෙලකට සකසන තුරු ලැයිස්තුවක ඇති යාබද මූලාංග සංසන්ධනය කර ඒවායෙන් කුඩා අගය මුලටත් විශාල අගය ඊලඟටත් පිහිටන පරිදි පිලිවෙලකට තේරීම බුබුළු තේරීම මඟින් සිදුකරයි.

```
#function
def bubble_sort(our_list):
    for i in range(len(our_list)):
        for j in range(len(our_list) - 1):
            if our_list[j] > our_list[j+1]:
                our_list[j], our_list[j+1] = our_list[j+1], our_list[j]
#program
our_list = [19, 13, 6, 2, 18, 8]
bubble_sort(our_list)
print(our_list)
```

```
19, 13, 6, 2, 18, 19
13, 19, 6, 2, 18, 19
13, 6, 19, 2, 18, 19
13, 6, 2, 19, 18, 19
13, 6, 2, 18, 19, 19
6, 13, 2, 18, 19, 19
6, 2, 13, 18, 19, 19
2, 6, 13, 18, 19, 19
```

Output

```
[2, 6, 8, 13, 18, 19]
```

Python ක්‍රමලේඛනයක් My SQL දත්ත සමුදායක් සමඟ සම්බන්ධ කිරීම.

සම්බන්ධතාවය පවතීදැයි පරීක්ෂා කිරීම.

- Python connector නැමැති මෘදුකාංගය ස්ථාපනය කරගන්න.
  https://dev.mysql.com/downloads/connector/python/2.0.html
- පහත විධානය භාවිතා කර මෘදුකාංගය ස්ථාපනය කර ඇත්දැයි බලන්න.
- import mysql.connector
- නැවැරදිව ස්ථාපනය කර ඇත්නම් දෝෂ පණිවිඩයක් නෙපෙන්වයි.

දත්ත සමුදාය සමඟ සම්බන්ධතාවයක් ඇති කර ගැනීමට (Create a connection) my sql ස්ථාපනයේදී ලබාදුන් පරිශීලක නාමය සහ මුරපදය සහිතව පහත පයිතන් කේතය ලියා ක්‍රියාත්මක කරන්න.

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword"
)
print(mydb)
```

දත්ත සමුදායක් සකස් කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword"
)
mycursor = mydb.cursor()
mycursor.execute("CREATE DATABASE mydatabase")
```

සකස් කළ දත්ත සමුදායන් නිරීක්ෂණය

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="myusername",
  passwd="mypassword")
mycursor = mydb.cursor()
mycursor.execute("SHOW DATABASES")
for x in mycursor:
  print(x)
```

වගුවක් නිර්මාණය කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
```

```
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
mycursor.execute("CREATE TABLE customers (name VARCHAR(255), address
VARCHAR(255))")
```

සකස් කළ වගු නිරීක්ෂණය කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()

mycursor.execute("SHOW TABLES")

for x in mycursor:
  print(x)
```

වගුවකට ප්‍රාථමික යතුරක් එක් කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
mycursor.execute("CREATE TABLE customers (id INT AUTO_INCREMENT PRIMARY KEY,
name VARCHAR(255), address VARCHAR(255))")
```

සකස් කල වගුවකට ප්‍රාථමික යතුරක් එක් කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
mycursor.execute("ALTER TABLE customers ADD COLUMN id INT AUTO_INCREMENT
PRIMARY KEY")
```

වගුවකට දත්ත එක් කිරීම

```
import mysql.connector

mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)

mycursor = mydb.cursor()
sql = "INSERT INTO customers (name, address) VALUES (%s, %s)"
val = ("John", "Highway 21")
mycursor.execute(sql, val)
mydb.commit()
print(mycursor.rowcount, "record inserted.")
```

වගුවකට එකවර රෙකෝඩ කිහිපයක් එක් කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
sql = "INSERT INTO customers (name, address) VALUES (%s, %s)"
val = [
  ('Peter', 'Lowstreet 4'),
  ('Amy', 'Apple st 652'),
  ('Hannah', 'Mountain 21'),
  ('Michael', 'Valley 345'),
  ('Sandy', 'Ocean blvd 2'),
  ('Betty', 'Green Grass 1'),
  ('Richard', 'Sky st 331'),
  ('Susan', 'One way 98'),
  ('Vicky', 'Yellow Garden 2'),
  ('Ben', 'Park Lane 38'),
  ('William', 'Central st 954'),
  ('Chuck', 'Main Road 989'),
  ('Viola', 'Sideway 1633')
]
mycursor.executemany(sql, val)
mydb.commit()
print(mycursor.rowcount, "was inserted.")
```

වගුවක ඇති සියළු දත්ත නිරීක්ෂණය කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
mycursor.execute("SELECT * FROM customers")
myresult = mycursor.fetchall()
for x in myresult:
  print(x)
```

තීරු කිහිපයක ඇති දත්ත ලබාගැනීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
mycursor.execute("SELECT name, address FROM customers")
myresult = mycursor.fetchall()
for x in myresult:
  print(x)
```

My SQL Where විධානය භාවිතා කිරීම

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
sql = "SELECT * FROM customers WHERE address ='Park Lane 38'"
mycursor.execute(sql)
myresult = mycursor.fetchall()
for x in myresult:
  print(x)
```

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
```

```
)
mycursor = mydb.cursor()
sql = "SELECT * FROM customers WHERE address LIKE '%way%'"
mycursor.execute(sql)
myresult = mycursor.fetchall()
for x in myresult:
  print(x)
```

```
import mysql.connector
mydb = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mydatabase"
)
mycursor = mydb.cursor()
sql = "SELECT * FROM customers WHERE address = %s"
adr = ("Yellow Garden 2", )

mycursor.execute(sql, adr)
myresult = mycursor.fetchall()
for x in myresult:
  print(x)
```

නිපුණතාවය 10- බහුමාධ්‍ය තාක්ෂණය භාවිතයෙන් වෙබ් අඩවි නිර්මාණය

ලෝක විසිරි වියමනෙහි අවශ්‍යතාවය ගවේෂණය කරයි

සන්නිවේදනය යනු පුද්ගලයින් අතර තොරතුරු හුවමාරු කර ගැනීමයි. අද ඇති වී තිබෙන තාක්ෂණික දියුණුව නිසා ම අකුරු,වචන,ශබ්ද, චලන රූප සහ වීඩියෝපට ඡායාරූප භාවිත කරමින් තොරතුරු හුවමාරු කරගත හැකි ය. පරිගණකයේ සහ සන්නිවේදනයේ සම්බන්ධතාව මගින් මාධ්‍යයන් හැසිරවීමේ හැකියාව පරිගණකය සතුවීමත් සමග ම තොරතුරු තාක්ෂණ සංකල්පය සහ අන්තර්ජාල තාක්ෂණයේ දියුණුව සමගාමීව සිදුවීමත් එමගින් විශ්ව ගම්මානය යන සංකල්පය බිහිවීමත් නිසා අද අප ජීවත් වන මේ සමාජය දුරස්ථ සමාජයක් නොවන බව පැහැදිලි ය. එසේ ම එදා දත්ත ගණනය කිරීම්වලට පමණක් සකස් කළ පරිගණකය අද බහුමාධ්‍ය තාක්ෂණය උපයෝගී කරගනිමින් තොරතුරු හුවමාරුවේ නියැලී සිටින ප්‍රබලම නියමුවා ලෙස හැඳින්විය හැකි ය.

බහුමාධ්‍ය යනු කුමක් ද? බහුමාධ්‍ය යන්න ඔබට අලුත් වචනයක් විය හැකි ය. මෙහි අදහස සරලව පැවසුවහොත් විවිධ මාධ්‍යයන්ගේ එකතුවක් ලෙස එය පෙන්වාදිය හැකිය. පුවත්පත් සඟරා චිත්‍රපට ගුවන් විදුලිය රූපවාහිනිය ආදී සන්නිවේදන මාධ්‍යයන්ගේ එකතුවක් ලෙස බහුමාධ්‍ය නිර්වචනය කළ හැකි අතර මෙම නිර්වචනය කිරීම තුළින් හැඳින්වූ බහුමාධ්‍යල පරිගණක තාක්ෂණය හරහා අකුරු, වචන, දත්ත, ඡායාරූප හෝ චිත්‍ර ,ශබ්ද, චලන රූප සහ වීඩියෝපට භාවිත කරමින් තොරතුරු හුවමාරු කරගත හැකි ආකාරයට වර්ධනය වීම නිසාම අද බහුමාධ්‍ය තාක්ෂණය නමින් අලුත් තාක්ෂණයක් අප අතරට පැමිණ තිබේ.

මෙවැනි බහුමාධ්‍ය තාක්ෂණික තොරතුරු අන්තර්ජාලයෙහි වෙබ් තාක්ෂණය භාවිත කරගනිමින් විවිධ ක්ෂේත්‍රයන් හි බහුල ව භාවිත වන අතර ඉන් ප්‍රධාන වශයෙන් අධ්‍යාපන ක්ෂේත්‍රයන්හිත්, ඉංජිනේරු ක්ෂේත්‍රයන්හිත් ,වෛද්‍ය ක්ෂේත්‍රයන්හිත්, විනෝදාස්වාදය ලබාදීමේදීත් වෙළෙඳ ප්‍රචාරණයේදීත් චිත්‍ර කරණයේදීත් වෙළෙඳ ලෝකයේ විවිධ අවස්ථාවලදීත් යොදා ගන්නා අතර විද්‍යාත්මක පර්යේෂණ කටයුතුවලදීත් යොදා ගැනේ. අධ්‍යාපන ක්ෂේත්‍රය මෙහි දී විශේෂයෙන් සඳහන් කළ යුතු වන අතර ඊ-අධ්‍යාපන කාර්යාවලියේ දී හෝ පරිගණක ආශි්‍රත අධ්‍යාපනයේ දී හෝ වෙබ් ආශ්‍රිත අධ්‍යාපනයේ දී හෝ පරිගණක විශ්වකෝෂ නිර්මාණයේ දී හෝ බහුමාධ්‍ය තාක්ෂණය බෙහෙවින් ම භාවිත වේ.

මෙවැනි අවස්ථාවල දී විස්තරාත්මක තොරතුරු වලට අමතරව එම විස්තර කරනු ලැබූ දෙය ඡායාරූප ද්විමාන හෝ ත්‍රිමාණ රූප සටහන් මගින් ඉදිරිපත් කරනු ලබන අතර තවදුරටත් පැහැදිලි කිරීම් අවශ්‍ය වූ අවස්ථාවල දී චලන චිත්‍ර හෝ වීඩියෝ පටයක ආකාරයෙන් ද ඉදිරිපත් කරනු ලැබේ. සමහර අවස්ථාවල දී භාෂාවක වචන උච්චාරණය කරන ආකාරය හෝ පක්ෂියකු ගේ මිහිරි නාදය ශබ්දයක ආකාරයෙන් ඉදිරිපත් කිරීමට ද සිදු වේ.

ඔබ විසින් වෙනත් රටක සිටින ඔබගේ මිතුරෙකුට පණිවුඩයක් ලබාදීමට බලාපොරොත්තුවෙන් සිටිනවා යැයි සිතන්න. එම පණිවුඩය ඔබගේ කටහඬින් ම ශබ්දයක ආකාරයෙන් බහුමාධ්‍ය තාක්ෂණය යොදා ගනිමින් විද්‍යුත් තැපෑල මගින් යැවිය හැකිනම් හෝ වෙබ් පිටුවක ආකාරයෙන් ලැබීමට සැලැස්වීමෙන් ඔබේ මිතුරා සතුටට පත් වෙනවා නොඅනුමානය. ඔබ හා ඔහු අතර පවත්නා දුරස්ථභාවය නැති කර ගැනීමට හෝ අඩු කර ගැනීමට මෙය මනා පිටිවහලකි. ඔබ දෙදෙනා අතර සබඳතා පවත්වා ගැනීමට මේ සඳහා ඔබ නොදැනුවත්වම බහුමාධ්‍ය තාක්ෂණය භාවිත කර ඇත. එයින් ඔබ ලබන ප්‍රයෝජනය සහ තෘප්තිය ඉතාමත් ඉහල මට්ටමක පවතී.

මේ සුළු උදාහරණයෙන් වුවද ඔබට හැඟීයන්නේ කුමක් දැ? සමාජයේ දුරස්ථභාවය අඩුකරමින් පවුල් සබඳතාව වැඩි කිරීමට පරිගණක තාක්ෂණයල සන්නිවේදන තාක්ෂණය සහ බහුමාධ්‍ය තාක්ෂණය සම්මිශ්‍රණය වීමෙන් බිහි වූ තොරතුරු තාක්ෂණය නම් වූ නව තාක්ෂණය ඉවහල් වී ඇති බවයි.

ටිම් බර්නස් ලී විසින් 1989 දී ලොවට පළමුවරට හඳුන්වා දෙනු ලැබූ විශ්ව විසිරි වියමන තුළ විශේෂිත වූ බහුමාධ්‍ය තොරතුරු හුවමාරු වන සහ බෙදාහරින පද්ධතියක් ආකාරයට අද වනවිට වර්ධනය වී ඇත. විශ්ව විසිරි වියමන මගින්

1. කුමන ආකාරයකට බහුමාධ්‍ය සහ අනෙකුත් තොරතුරු සංවිධානය කළ යුතු ද?
2. සංවිධානය කරනු ලබන තොරතුරු කුමන ආකාරයකට පරිගණක පද්ධතිය තුළ තැන්පත් කළ යුතු ද?
3. එක් පරිගණක පද්ධතියක තිබෙන තොරතුරක් තවත් ස්ථානයක තිබෙන පරිගණක පද්ධතියක තොරතුරක් හා සම්බන්ධ කරන්නේ කෙසේ ද?
4. එම තොරතුරු නැවත ලබා ගන්නා ආකාරය
5. භාවිත කළයුතු ආකාරය

පිළිබඳ ව එකඟතාවයකට පත් වූ නියමයන් පද්ධතියකින් යුත් නිර්මාණයකි. www server නමින් හඳුන්වනු ලබන පරිගණකය තුළ ඉහත සඳහන් කරන ආකාරයට සකස් කරනු ලබන තොරතුරු අඩංගු වෙබ් පිටු රාශියක් තැන්පත් කර ඇති අතර එම පරිගණක යන්ත්‍රවල ඇති තොරතුරු ඕනෑම කෙනෙකුට පහසුවෙන් ලබා ගැනීමේ හැකියාව ද ඇත. www server නමින් හඳුන්වනු ලබන පරිගණකය තුළ එක් එක් ආයතනයන් හෝ එක් එක් පුද්ගලයින් සතු වෙබ් අඩවි රාශියක් පවත්වා ගෙන යා හැකි අතර මෙම සෑම වෙබ් අඩවියක් සඳහා ම ප්‍රධාන වෙබ් පිටුවක් ඇති අතර අනෙකුත් අදාළ වෙබ් පිටු 'Hyper-links'. මාර්ගයෙන් ප්‍රධාන පිටුවට සම්බන්ධ කර ඇත. ප්‍රධාන පිටුවට සම්බන්ධ කර ඇති මෙවැනි වෙබ් පිටු 'Linked Pages' ලෙස හඳුන්වනු ලැබේ.

වෙබ් පිටු ලෙස 'Hyper-links' මගින් හෝ 'Hyper-media'. මගින් හෝ තොරතුරු පද්ධතීන් එකිනෙකට සම්බන්ධ කරමින් නිර්මාණය වී ඇති මෙම . www server පද්ධතියේ මූලික ස්වරූපය වන්නේ ඕනෑම අයෙකුට ඕනෑම ආකාරයකට හෝ අනුපිළිවෙළකට ඕනෑම වෙලාවකට තොරතුරු ලබාගත හැකි ආකාරයට සකස් වී තිබීමයි. (බොහෝ අවස්ථාවලදී තොරතුරු ලබාදීමේ දී එය කිසියම් අනුපිළිවෙලකට ලබාදීමට සිදු වී ඇති අවස්ථා ද ඇත. උදාහරණ ලෙස ගුවන් විදුලිය මගින් හෝ රූපවාහිනිය මගින් හෝ අපට ලබාදෙන තොරතුරු මේ වර්ගයට වැටේ.*

එහෙත් . 'Hyper-media'. ක්‍රමයට අනුව එක් ලිපියක වැදගත් යමක් පිළිබඳ දීර්ඝ විවරණයක් අවශ්‍ය වූ විට ඒ සඳහා කිසියම් සම්බන්ධතාවක් ඇති කරමින් එම දීර්ඝ විස්තරය එම පරිගණක පද්ධතිය තුළ ඇති වෙනත් වෙබ් ලිපියකට හෝ වෙනත් වෙබ් අඩවියක ඇති

වෙබ් පිටුවකට හෝ තවත් . www server පද්ධතියක ඇති වෙනත් වෙබ් අඩවියකට හෝ වෙබ් පිටුවකට සම්බන්ධ කළ හැකිය. මේ අනුව මෙම සම්බන්ධතාවය ඇති කළ හැකි අවස්ථා 4ක් පිළිබඳව සඳහන් කළ හැකි ය. එනම්

1. එම වෙබ් පිටුවෙහි ඇති වෙනත් කොටසකට සම්බන්ධතාවය ඇති කිරීම.
2. වෙබ් අඩවිය තුල තැන්පත් කරඇති වෙනත් ලිපියකට සම්බන්ධතාවය ඇති කිරීම.
3. වෙනත් වෙබ් අඩවියකට සම්බන්ධතාවය ඇති කිරීම.
4. වෙනත් වෙබ් අඩවියක ඇති වෙනත් වෙබ් පිටුවකට සම්බන්ධතාවය ඇති කිරීම කළ හැකි ය.

WWW' හි ඇති තවත් වැදගත් ලක්ෂණයක් වන්නේ විවිධ බහුමාධ්‍ය තොරතුරු සඳහා සම්බන්ධතා ඇති කිරීමයි. උදාහරණ ලෙසල මෙම සම්බන්ධතාව ඡායාරූපකයකට වීඩියෝපට ඛණ්ඩයකටල ශබ්ද ඛණ්ඩයකට හෝ චලන චිත්‍රයකට ඇති කළ හැකි ය.

අන්තර්ජාලය තුළ ඇති ලිපිගොනු සකස් කිරීමේ දී අකුරු පමණක් භාවිත කරමින් සකස් කර ඇතිනම් අන්තර්ජාලය හරහා එවැනි වෙබ් ලිපි ලබා ගැනීම එතරම් අපහසු නොවේ. පරිගණක ලිපිගොනු තුළ ඇති තොරතුරු අන්තර්ජාලය තුළින් භාවිත කිරීමට හැකිවන අයුරින් නිර්මාණය කිරීම සඳහා සකස් කළ යුතු ආකාරය දක්වන උපදේශ කේත අන්තර්ජාලය තුළ ලිපිගොනු හසුරුවන ආකාරය පිළිබඳව කලින් සකස් කරන ලද නීති මාලාවකට යටත්ව සකස් කර . 'HyperText Markup Language' (HTML) මගින් ඉදිරිපත් කර ඇති හෙයින් ඔබට . 'HTML' භාවිත කරමින් වෙබ් ලිපි සකස් කළ හැකිය. අවශ්‍ය වූ

විටෙක රූපල ශබ්දල වීඩියෝ පට හෝ චලන රූප ඇතුළත් කිරීම සඳහා අවශ්‍ය වන සියලු ම උපදේශ කේත ද . 'HTML'භාෂාව මගින් ඉදිරිපත් කර ඇත.

අන්තර්ජාලය තුළ තොරතුරු ලබා ගැනීමට භාවිත කරන වෙබ් බ්‍රවුසරය මගින් මෙම උපදේශ කේත නිසි ආකාරයට හසුරුවමින් එම ලිපියේ ඇති තොරතුරු පරිගණක තිරය මතට ලබා දේ. මෙය පියවර හතරකින් සිදු වේ.

'WWW' තුළ ඇති අසීමිත වූ තොරතුරු ලබා ගැනීම සඳහා භාවිත කරනු ලබන වෙබ් බ්‍රවුසරය නමින් හඳුන්වනු ලබන පරිගණක යෙදුම ඕනෑම පරිගණක යන්ත්‍රයක භාවිත කළ හැකි ආකාරයට සකස් වී ඇත. නිදහසේ වෙබ් අඩවිවලින් තොරතුරු සොයා යමින් එම තොරතුරු තම පරිගණක යන්ත්‍රයකට ලබාදෙන මෙවැනි වෙබ් යෙදුම් මුදලක් නොගෙවා වුව ද ලබා ගැනීමට හැකිවේ. 'Firefox', 'Internet Explorer' , 'Safari' හෝ 'Google Crome'. වැනි වෙබ් බ්‍රවුසරයන් මගින් වෙබ් අඩවි වල තොරතුරු ලබාදීමට ඉදිරිපත් වී සිටී.

එපමණක් නොව මෙම වෙබ් බ්‍රවුසරයන් මගින් පහසුවෙන් තොරතුරු සොයා යෑම සහ ඉතාමත් පහසුවෙන් භාවිත කිරීමට හැකිවන ආකාරයට නිර්මාණය කර ඇති අතර කිසිම අපහසුවකින් තොරව මූසිකයෙහි බොත්තමක් තද කරමින් පමණක්අවශ්‍ය තොරතුරු ලබාගැනීමේ හැකියාව ද ඇත.

විශ්ව විසිරි වියමන යනු අන්තර්ජාලය හරහා පිවිසිය හැකිවූ ද එකිනෙකට සම්බන්ධ වූ දල ලෝකය පුරා ඇති පරිගණක වල ගබඩා කර ඇති විද්‍යුත් ලේඛනවල විශාල එකතුවකි. මෙය අන්තර්ජාලයේ පරිගණක අතර තොරතුරු බෙදාහරින ආකාරයක් ලෙස හැඳින්විය හැකිය. උඋඋ හි නිර්මාතෘ ලෙස ස්විස්ටර්ලන්තයේ ජිනීවා නුවර **සර් ටිම් බර්නර්ස් ලී** සලකනු ලැබේ. භාවිත කරන්නාට වෙබ් අතරික්සුවක් හරහා වෙබ් අඩවි වලට ප්‍රවේශ විය හැක.

වෙබ් පිටුවකින් හෝ පිටු කීපයකින් වෙබ් අඩවියක් සමන්විත වේ. ලිඛිත සටහන්ල පින්තූරල වීඩියෝ සහ අනෙකුත් බහුමාධ්‍ය වලින් සමන්විත වෙබ් පිටු දැකගත හැකි අතර අධි සම්බන්ධක භාවිතයෙන් ඒවා අතර සැරි සැරීමට හැකිය.

අන්තර්ගතය හා ව්‍යුහය සලකා බැලීමෙන් වෙබ් අඩවි වර්ග පහත ආකාරයට බෙදා දැක්විය හැකි වුවද ඒවාහි ඉතා පැහැදිලි වර්ගීකරණයක් දක්නට නො ලැබේ.

වෙබ් අඩවි ප්‍රරූප

වෙබ් අඩවියක් නිර්මාණයේදී අවධානය යොමු කළ යුතු අංශ බොහෝමයක් ඇත උදා උදාහරණයක් ලෙස වෙළඳසැල් සඳහා වෙබ් අඩවියක් නිර්මාණය කිරීමට පැවරී ඇතැයි සිතන්න ඔබ විසින් එම වෙබ් අඩවිය සාදනු ලබන්නේ කුමක් සඳහාද කා සඳහාද කෙසේද නඩත්තු කරන්නේ කෙසේද යන කරුණු පිළිබඳ ගැඹුරින් හා පුළුල්ව විමසා බැලිය යුතුය.

වෙබ් අඩවි ඒවායේ ස්වභාවය සහ අරමුණු අනුව වර්ගීකරණයකට ලක් කිරීමට පුලුවන එම වර්ග වර්ගීකරණය අතරින් ප්‍රධාන වර්ග වර්ගීකරණය කිහිපයක් පහත දක්වා ඇත.

01. තොරතුරු හා ප්‍රවෘත්ති

මෙම වෙබ් අඩවි ඉතා ඉක්මනින් බහුමාධ්‍ය ආකාරයට ප්‍රවෘත්ති ලබාදෙන අතර සෞඛ්‍ය අධ්‍යාපනික ආදී විවිධ තොරතුරු සන්නිවේදනය වෙනුවෙන් සකස් වී ඇති බැවින් ජනතාව අතර ඉතා ප්‍රචලිත ය

නිදසුන්

- www.itn.lk
- www.rupavahini.lk
- www.bbc.co.lk
- www.nie.lk
- www.webopedia.org

02. පෞද්ගලික, අධ්‍යාපනික ,ව්‍යාපාරික සහ පර්යේෂණ

වෙබ් අඩවි නාම හා අන්තර්ජාල පහසුකම් සපයන ආයතන විසින් ලබාදෙන නාමයන් යටතේ පෞද්ගලික ඡායාරූපප මාර්ගගත දිනපොත් වැනි තොරතුරු පවත්වාගෙන යාම සඳහා නිර්මාණය කරන ලද වෙබ් අඩවි පෞද්ගලික වෙබ් අඩවි ලෙස හඳුන්වා දීමට පුළුවන් තවද විශේෂිත විෂයයක් හෝ සංවිධානයක් ගැන තොරතුරු සොයා ගත හැකි වෙබ් අඩවි මේ වර්ගයට අයත් වේ. මෙය බහුලව දක්නට ලැබෙන අතර මෙහි සෘජුව භාණ්ඩ විකිණීමක් නො කරන නමුදු බැනර් සහ දැන්වීම් දැකිය හැකිය. මෙම වෙබ් අඩවි වල විද්‍යුත් තැපැල් ලිපිනයක් හා දුරකථන අංකයක් බොහෝවිට දක්නට ලැබේ. රාජ්‍ය ආයතන වල වෙබ් අඩවි, අධ්‍යාපන ආයතනවල වෙබ් අඩවි මේ වර්ගයට අයත් වේ

- www.abcblogspot.com
- www.amazon.com
- www.linkedin.com
- www.arthurcclarke.net
- www.yahoo.com

03. ජාල ප්‍රවේශ ද්වාර

ජාල ප්‍රවේශ ද්වාර මගින් අදාළ තොරතුරු කෙටි කාලයකින් ලබාදේ. සමහරවිට වෙළඳ පොළ වාර්තා, ප්‍රවෘත්ති, කාලගුණ වාර්තා, දුරකථන නාමාවලි වැනි දෑ මෙන්ම E-mail ගිණුම්, විනෝදාස්වාදය, පිළිසඳර ක්‍රීඩා, සෙවුම් යන්ත්‍ර යනාදී සේවා ද මේවායේ අඩංගුය.

- www.superonline.com
- www.netflixcom
- www.youtube.com
- www.msn.com
- www.google.com

පරිශීලක අවශ්‍යතා විශ්ලේෂණය කිරීම

වෙබ් අඩවියක අභිමතාර්ථ නිර්වචනය

කලින් සඳහන් කළ පරිදි . 'Rich-text' සහිත සහසම්බන්ධතා වලින් යුක්ත වූ . Hypermedia වර්ගයේ වෙබ් ලිපි හැසිරවීමේ හැකියාව වෙබ් බ්‍රවුසරයකට ඇති අතර, වෙබ් බ්‍රවුසරයන්ට හසුවන වෙබ් අඩවි සහ එම වෙබ් අඩවි නිර්මාණය කිරීම සඳහා භාවිත කර ඇති වෙබ් ලිපි තුළ ඇති විශේෂිත වූ ක්‍රම වලින් සකස් කරන ලද, ඡේද අක්ෂර සහ චිත්‍ර හැසිරීමට ද හැකියාව ඇත. ඊට අමතරව සජීවී ගුවන් විදුලි නාලිකා හෝ සජීවී රූපවාහිනී නාලිකා ඇසීමට හෝ නැරඹීමට හැකිවන පරිදි සකස් කරන ලද වෙබ් අඩවි ද ඇත. එවැනි වෙබ් අඩවි තුළට පිවිසීමේ හැකියාව ඇතිවා පමණක් නොව එම සජීවී තොරතුරු ලබාදීමේ හැකියාව ද මෙම වෙබ් බ්‍රවුසරයන්ට ඇත. තවත් අවස්ථා වල දී සජීවී කතා කිරීම් අන්තර්ජාල ,දුරකථන ,සාකච්ඡා මණ්ඩල ,ප්‍රවෘත්ති මණ්ඩල සහ ටෙලි සම්මේලන හැසිරවීමේ හැකියාව ද ඇත. මීට අමතරව නොයෙකුත් අවස්ථා වලදී නොයෙකුත් පරිගණක යෙදුම් මගින් නිර්මාණය කරන ලද අන්තර්ගත යන්ගෙන්යුත් වෙබ් අඩවි තුල ඇති තොරතුරු කියවීමේ හැකියාව ද (උපකාරක යෙදුම් මගින් [Plug-in]) මෙම වෙබ් බ්‍රවුසරයන්ට ඇත.

උපකාරක යෙදුම් භාවිත කරනු ලබන වෙබ් බ්‍රවුසරයන් හි බලය ඉතා ඉහළ තත්ත්වයක පවතී. උදාහරණයක් ලෙස වෙබ් අඩවියක ඇති 'Quick Time Movie' (Special multimedia file format) වර්ගයේ චලන වීඩියෝ පටයක් නැරඹීමට ඔබට සිදුවුවහොත් ඔබගේ වෙබ් බ්‍රවුසරයන්ට එම හැකියාව නැතත් එම හැකියාව ඇති උපකාරක යෙදුම් (Quick Time Movie Player) ඔබ පරිගණකයට ලබාගෙන එම හැකියාව ඔබගේ වෙබ් බ්‍රවුසරයන්ට ලබා දිය හැකියි.

වෙබ් අඩවියට ප්‍රවේශ වී එය පරිශීලනය කරන පුද්ගලයෝ එම වෙබ් අඩවියේ ග්‍රහකයන් ලෙස හැඳින්වේ. උදාහරණයක් ලෙස ලෝකය පුරා විසිර සිටින වෙබ් අඩවිය භාවිතා කරලා පුද්ගලයෝග්‍රහක වෙති. වෙබ් අඩවියක් නිර්මාණය දී ග්‍රාහක විවිධත්වය පහත අයුරින් හඳුනාගත හැක.

1. භාෂාව හා සංස්කෘතියය
2. වයස් මට්ටම
3. අධ්‍යාපනික මට්ටම
4. ස්ත්‍රී පුරුෂ බව
5. රුචිකත්වය

ග්‍රහක කණ්ඩායම් විශ්ලේෂණය යනු ඉලක්කගත ග්‍රාහක කණ්ඩායම් පිළිබඳව හොඳින් විමසා බලා ඔවුන් සතු සුවිශේෂ ලක්ෂණ සහ වෙනස්කම් වුණා ගැනීමයි
මේ සඳහා විවිධ ක්‍රම අනුගමනය කළ හැකිය නිදසුනක් ලෙස

- ඉලක්කගත ග්‍රාහකයන් පිළිබඳව අධ්‍යයනයක යෙදීම
- මෙවැනිම වූ වෙනත් වෙබ් අඩවි පිළි පිළිබඳව විමසා බැලීම

වෙබ් අඩවිය ඵලදායිතා යථා පරිදි වූ තොරතුරු පිරිසැලසුම් නිර්මාණය

මින් අදහස් වනුයේ වෙබ් පිටු සඳහා යොදානු ලබන පසුතල වගුහා රාමු , වර්ණ ,අකුරු වර්ග, අකුරු ප්‍රමාණ ,අකුරු හැඩ යනාදිය මෙහිදී සැලකිය යුතු කරුණු රාශියක් හඳුනාගත් හැක ඉන් කිහිපයක් නම්,

01. සෑම පිටුවක්ම වෙබ් අඩවියේ අනන්‍යතාවය පවත්වා ගැනීම

මෙහිදී එකම වෙබ් අඩවියට අයත් පිටු සමූහයේම ව්‍යුහය හා පිරිසැලසුම ඒකාකාරව පවත්වා ගැනීම අදහස් කෙරේ.

02.වැදගත් කරුණු උද්දීපනය මේ සඳහා එකී තොරතුරට අදාළව අකුරු වර්ගය, අකුරු ප්‍රමාණය වර්ණය වෙනස් විය හැකි ක්‍රම භාවිතා කළ හැකියි හැකිය.

03.ඕනෑම විවිධ වර්ගයේ පරිගණක වලින් වෙබ් අඩවිය පරිශීලනයය කළ හැකි වීම

විවිධ වර්ගයේ පෞද්ගලිකපරිගණක, ජංගම දුරකතන ආදී විවිධ උපාංග වෙබ් අඩවි පරිශීලනය සඳහා භාවිතා කෙරේ මෙහිදී වෙනස් ප්‍රමාණවල තිර යොදාගැනෙන අතර ඒවායේ තිර විභේදන වෙනස් බැවින් ඇතැම් වෙබ් පිටු එක් එක් යුගවලදී වෙනස් ආකාරයට දර්ශනය වීම වැළැක්වීමට තිරයේ ප්‍රමාණයට අනුව වෙබ් පිටුවේ ප්‍රමාණය සකස්වන ආකාරය වෙබ් අඩවි නිර්මාණය කළ යුතුය.

වෙබ් අඩවියක පිටු හඳුනා ගැනීම

වෙබ් අඩවියන් සඳහා තෝරාගත් අන්තර්ගතය ක්‍රමවත් පිළිවෙලකට පෙළගැස්වීම සන්ධාර සංවිධානයෙන්ද සිදුකරනු ලබයි.මෙහි ප්‍රධාන අරමුණ වන්නේ ග්‍රාහකයන්ට කාර්යක්ෂම ලෙස වෙබ් පිටු පරිශීලනයන් ට පහසුකම් සැලසීමයි.

වෙබ් අඩවියක් තුළ මුල් පිටුවක් හා එයට සම්බන්ධ කරන ලද තවත් පිටු ගණනාවකින් සමන්විත වෙයි. එමෙන්ම වෙබ් අඩවියක් පහසුවෙන් හැසිරවීමට හැකි විය යුතුය. වෙබ් අඩවියක පිටු සැකසීම වෙබ් අඩවියේ වර්ගය හා අන්තර්ගත කරුණු අනුව වෙනස් වේ.

මුල් පිටුව

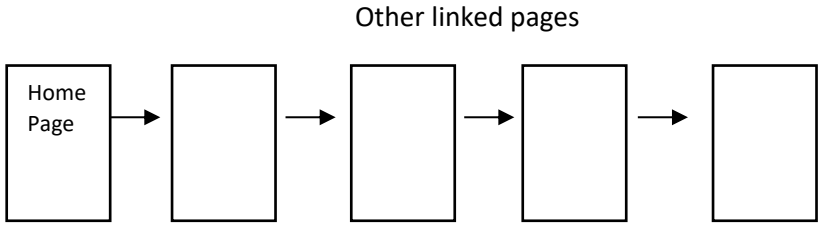
වෙබ් බ්‍රවුසරයක් හරහා වෙබ් අඩවියකට පිවිසෙන විට මුලින්ම දිස්වන පිටුවවෙබ් අඩවියේ ප්‍රධාන පිටුව හෙවත් මුල් පිටුව ලෙස හඳුන්වනු ලැබේ.

ඇඳුම් පිටුව

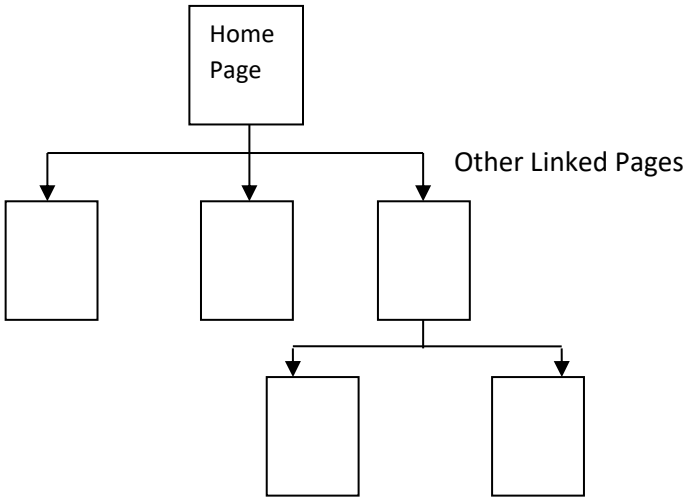
මුල් පිටුව සමග සම්බන්ධ වී ඇති අනෙකුත් පිටු ඇඳුම් පිටු නමින් හඳුන්වයි. වෙබ් පිටුවක තැනුම් ඒකකයක් වන්නේ රූපක, ශ්‍රව්‍ය දෘශ්‍යයන් වැනි බහු මාධ්‍යය අංගයන්ය.

වෙබ් අඩවියක මුලි පිටුව හා ඇඳුම් පිටු සම්බන්ධ කළ හැකි වින්‍යාස කිහිපයකි

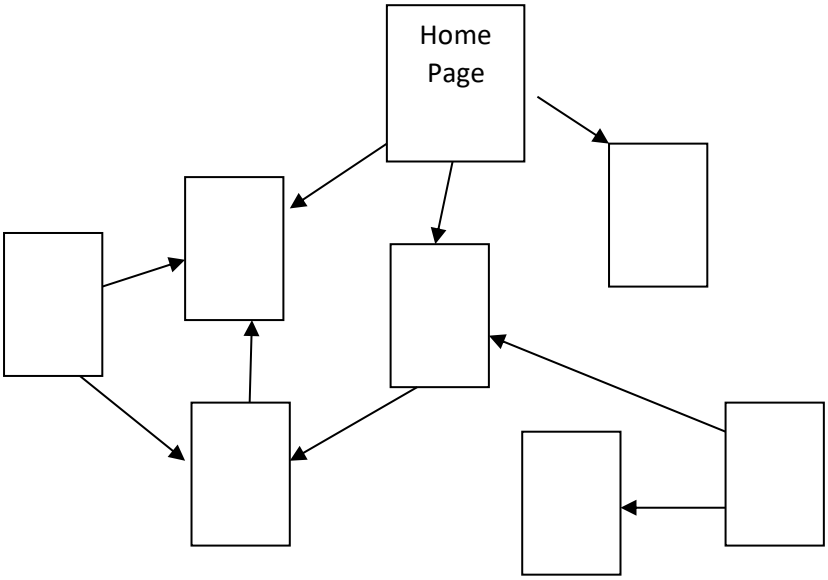
1 .රේඛීය වින්‍යාසය



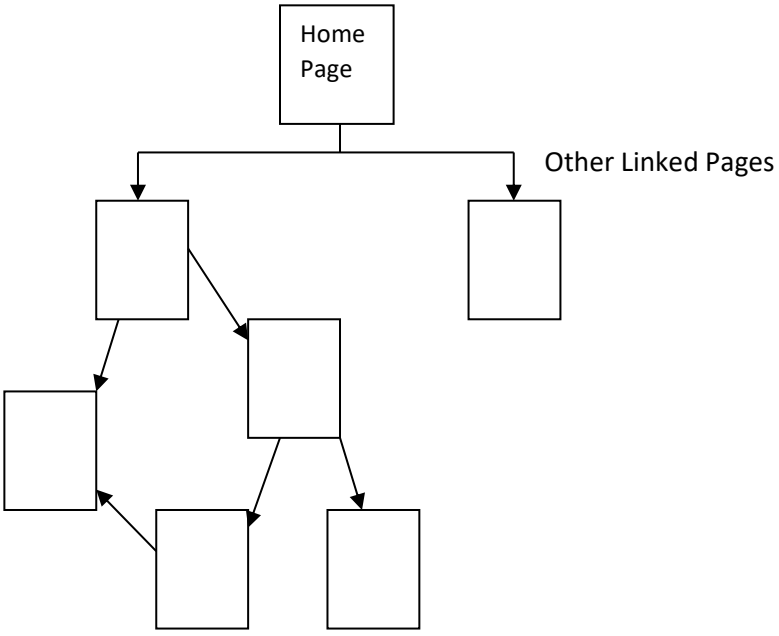
2.ධූරාවලි වින්‍යාසය



3. ජාල විනාසය



4. සම්බන්ධතාවය



වෙබ් පිටුවක අන්තර්ගතය හා හඳුනාගැනීම

ඉහත දැක්වූ පරිදි වෙබ් අඩවියක් තුළ ඇති ශව්‍ය දෘශ්‍ය තොරතුරු සහ සේවා සියල්ල එහි අන්තර් අන්තර්ගතය ලෙස හැඳින්වේ.

වෙබ් අඩවිය නිර්මාණය කුඩුදුන් අවශ්‍යතාව අවශ්‍යතා සහ ග්‍රාහක කණ්ඩායම් කණ්ඩායම් විශ්ලේෂණයේදී හඳුනාගත් ග්‍රාහක අවශ්‍යතා සැලකිල්ලටට ගනිමින් නව වෙබ් අඩවියේ අන්තර්ගතය තෝරාග ගත යුතුවේ.ඒ තුළ ග්‍රාහකයාට සන්නිවේදනය කළ යුතු සියලු තොරතුරු මෙන්ම වෙබ් අඩවියෙන් පරීශීලනයට පහසු කිරීම සඳහා ග්‍රාහකයා වෙත සැපිය යුතු සේවා ද අඩංගු කළ යුතුය

වෙබ් පිටුවක තොරතුරු ඉදිරිපත් කිරීමේදී පහත කරුණු සලකනු ලැබේ.

- අවශ්‍ය තොරතුරු පමණක් ඇතුලත් කිරීම.
- කෙටි කාලයක දී තේරුම් ගැනීමට හැකි පරිදි ප්‍රමාණාත්මක තොරතුරු ගොනු කිරීම මෙහිදී වගු, ලැයිස්තු ආදිය යොදා ගනී.
- විවිධ පුද්ගලයන්ට ග්‍රහණය කරගත හැකි පරිදි නොයෙක් ආකෘති වලින් තොරතුරු සරල භාෂාවකින් දැක්වීම.
- තේරුමක් ඇති පැහැදිලි නිවැරදි කෙටි මාතෘකා, කෙටි වාක්‍ය සහ කුඩා ෙඡ්ද යෙදීම.
- භාවිතා කරන්නාට කාර්යක්ෂමව තොරතුරු සෙවීම සඳහා තේරීම් වාර ගණන හා පිටු ගණන අවම කිරීම.
- කරුණු සංවිධානය කිරීමේදී භාවිතා කරන්නාට පහසුවෙන් තේරුම් ගත හැකි පරිදි සම්බන්ධයක් ඇති තොරතුරු එකට කාණ්ඩගත කිරීම.
- තොරතුරුවල කාලීන හා අපක්ෂපාතී බව.
- වෙබ් පිටු මුහුණත ආකර්ෂණීය වීම හා ඒ සඳහා අකුරුවල ප්‍රමාණය, හැඩය හා පැහැය ආදී කරුණු කෙරෙහි අවධානය යොමු කිරීම.
- උචිත වලන හා නිසල රූප යොදා ගැනීම.
- කරුණු යාවත්කාලීන කිරීම.

10.3 වෙබ් පිටු නිර්මාණය සඳහා HTML භාවිතා කරයි.

විශ්ව විසිරි වියමන (World Wide Web) හෙවත් WWW හි භාවිත කරනු ලබන වෙබ් අඩවි වල ඇති වෙබ් පිටු සකස් කිරීම සඳහා භාවිත කරනු ලබන ක්‍රම පිළිවෙළ හෝ ආකාරය (Format) සඳහන් වූ නියමයන් හෙවත් භාෂාව 'Hypertext Markup Language' හෝ කෙටියෙන් 'HTML' ලෙස හඳුන්වනු ලබන බව කලින් සඳහන් කළෙමු.

වෙබ් බ්‍රවුසරයන් මගින් සරල ආකාරයට සකස් කරනු ලබන වෙබ් ලිපි (Web pages in plain text) නිරායාසයෙන් ම වෙබ් අඩවි තුළින් ලබා ගෙන ඔබගේ පරිගණක තිරය මත පතිත කරනු ලබන අතර, ඉතාමත් ම සංකීර්ණ වූ වගු සකස් කිරීම, රූප, ශබ්ද හා වීඩියෝ පට වැනි බහුමාධ්‍ය මෙවලම් ඇතුළු කිරීම වැනි කාර්යයන් සඳහා ද 'HTML' භාෂාව යොදු ගනු ලැබේ.

එපමණක් නොව අන්තර්ජාලය තුළ ඇති අනෙකුත් වෙබ් ලිපි සහ වෙබ් අඩවි සඳහා අවශ්‍ය වන සම්බන්ධතා (Links) හෙවත් 'Hyper - Links' ඇති කිරීම වැනි කාර්යයන් සඳහා ද 'HTML' භාවිත කරනු ලැබේ. 'Hyper - Links' යනු අතිශය විශාල ප්‍රදේශයක විසිරී ඇති පරිගණකයන් හි ගොනු කර ඇති පරිගණක ලිපි, වෙබ් ලිපි හෝ වෙබ් අඩවි සඳහා පමණක් නොව විවිධ බහුමාධ්‍ය මෙවලම් සඳහා ද සම්බන්ධතාව ඇති කිරීම සඳහා යොදු ගනු ලබන ක්‍රමයයි.

වෙබ් අඩවි සඳහා සුදුසු ආකාරයට පරිගණක ලිපිගොනු සකස් කිරීමේ දී එම ලිපි සකස් කිරීම සඳහා භාවිත කරනු ලබන නියමයන් හෙවත් උපදේශ කේත (Tags හෙවත් HTML Codes) යොදු ගනිමින් ලිපියෙහි කොටස් විදහා දැක්විය යුතු ආකාරය කෙසේද යන්න 'HTML' මගින් හසුරුවනු ලබයි.

HTML File එකක් හඳුනාගනිමු

- HTML මගින් Hyper Text Markup Language යන්න කෙටිකර දක්වයි.
- HTML වෙබ් පිටු නිර්මාණය කිරීමට භාවිතා කරන භාෂාවකි
- HTML File එකක් Text File එකක් වන අතර එහි කුඩා Markup Tags අන්තර්ගත වේ.
- Web Browser එකෙන් Page එක පෙන්වන ආකාරය Markup Tags මගින් දක්වයි.

- HTML File එකක Extension එක htm හෝ html විය යුතුයි.
- Text Editor එකක් භාවිතා කොට HTML File එකක් නිර්මාණය කල හැක.
- HTML ක්‍රමලේඛන භාෂාවක් (Programming Language) නොවේ.

HTML Elements හා HTML Tags

HTML File එකක් HTML Elements හා Tags වලින් සමන්විත වෙයි. උදාහරණයක් ලෙස HTML වලදී අකුරු Bold කිරීමට Tag එකත් හිස් පේළියක් සඳහා
 Tag එකත් යොදයි. මේවායේ පිලිවෙලින් b, br යන්න HTML Elements වේ.HTML Tags ආරම්භයේ දී "<" ද අවසානයේදී ">" ද යෙදෙන අතර මේවා Angle Brackets ලෙස හඳුන්වයි.

- සාමාන්‍යයෙන් Tags යුගල ලෙස පවතී. උදා - සහ
- පළමු Tag එක Start Tag එක ලෙසද දෙවැන්න End Tag එක ලෙස ද හඳුන්වයි.
- මේ දෙක අතර පවතින Text, Element Content ලෙස හඳුන්වයි.
- HTML Tags, Case Sensitive නොවේ. එනම් හා <B> මගින් එකම Tag එක නිරූපණය වේ.
- Browser එකට Tag එකක් හඳුනාගත නොහැකි නම් එය නොසලකා හරී.

සරල 'HTML' ලිපියක් නිර්මාණය කිරීම

මින් ඉදිරියට සඳහන් කරනු ලබන උදාහරණ මගින් 'HTML' භාවිතයත් 'HTML' මගින් සකස් කරනු ලැබූ වෙබ් ලිපියක අඩංගු වී ඇති උපදේශ කේත (Tags) සහ ඒ සමග භාවිත වන විවිධ වූ අනු උපදේශ කේත භාවිත වන යොදා ඇති ආකාරය පෙන්වා දී ඇත. ඔබගේ අවධානය යොමු විය යුතු ප්‍රධාන උපදේශ කේත කිහිපයක් පිළිබඳ ව
මෙහි දී සඳහන් කරනු කැමැත්තෙමු.

```
<html>
<head>
<title>This is my first example</title>
</head>
<body>
<h1>What is HTML</h1>
<p>Hyper Text Markup Language</p>
</body>
</html>
```

ඉහත දී ඇති කොටස Notepad හි Type කර File Name කොටුවේ mypage.htm ලෙස type කර, Save as type වලට All Files තෝරා Save කරන්න.

මෙම උදාහරණයෙහි පිළිවෙලින් <HTML>, <TITLE></TITLE>, <H1></H1>, <P></P> සහ </HTML> යන උපදේශ කේත භාවිත වී ඇත. ලිපිය සකස් කිරීමේ දී භාවිත කරන ලද උපදේශ කේත 4 පිළිබඳ කෙටි හැඳින්වීමක්
පහතින් දක්වා ඇත.

<HTML> - අන්තර්ජාල වෙබ් පිටුවක ආරම්භය දක්වනු ලැබේ.
<TITLE> - අන්තර්ජාලය තුළ ඔබගේ වෙබ් අඩවිය හඳුන්වා දීම සඳහා යොදනු ලබන ශීර්ෂ මාතෘකාව
සටහන් කිරීම සඳහා යොදා ගනු ලබයි.
<H1> - මෙය භාවිත කරනු ලබන්නේ වෙබ් ලිපිය තුළ ඇති ප්‍රධාන මාතෘකා සටහන් කිරීම සඳහා ය.
<P> - අන්තර්ජාලය තුළ ඇති ලිපි වල ඡේද වෙන්කිරීම සඳහා භාවිත කරනු ලැබේ.

මූලික HTML Tags

<html> - HTML File එකක ආරම්භය දක්වයි
<body> - HTML File එකක අන්තර්ගතයේ ආරම්භය දක්වයි
<h1> to <h6> - මාතෘකා වර්ග 6ක් නිරූපණය කරයි (1-6)
<p> - ඡේදයක ආරම්භය නිරූපණය කරයි

 - හිස් පේළියක් ලබා දෙයි. මේ Tag එකේ වෙනත් කිසිවක් ඇතුලත් නොවේ. (empty tag)
<hr> - තිරස් ඉරක් ලබා දෙයි
<!-- comment --> - Web පිටුවේ නොපෙන්වන දෙයක් සඳහා යොදයි
 - ලයිස්තුවක් (List) සෑදීමට යොදයි
 - අංක සහිත ලයිස්තුවක් සාදයි
 - අංක රහිත ලයිස්තුවක් සාදයි
<a> -තවත් යමකට සම්බන්ධයක් නිරූපණය කරයි

'HTML' ලිපි සඳහා 'BODY', 'HEAD' සමඟ ශීර්ෂ පාඨ සඳහා 'TITLE' උපදේශ කේත යොදා ගැනීම

වෙබ් ලිපියක සෑම විට ම ආරම්භයේ දී <HTML> සහ අවසානයේ දී </HTML> යන යුගල උපදේශ කේත භාවිත වන අතර එමගින් එම ලිපිය 'Hypertext Markup Language' යට අනුව සකස් වූ ලිපියක් බව හඳුනා ගනී.

Example

```
<HTML>
<HEAD>
<TITLE>Test Example </TITLE>
</HEAD>
<BODY>
<H1>This is where the body of the document...</H1>
<H2>This is Heading One</H2>
<H3>This is Heading Two</H3>
<H4>This is Heading Three</H4>
</BODY>
</HTML>
```

'HTML' ලිපියක ශීර්ෂය

<HTML> යොදු ගැනීමෙන් අනතුරුව මෙම ලිපියෙහි අඩංගු කළ යුතු මූලික තොරතුරු හඳුනා ගැනීම සඳහා <HEAD> උපදේශ කේතය භාවිත කරනු ලබයි. එය අවසන් කිරීම සඳහා </HEAD> භාවිත කරනු ලබන අතර එම උපදේශ කේත දෙක අතර පමණක් <TITLE> උපදේශ කේතය භාවිත කළ යුතු ය.

<TITLE>—</TITLE> වෙබ් ලිපියෙහි ශීර්ෂ මාතෘකාව

අන්තර්ජාල වෙබ් අඩවි තිරය මත පතිත වීමේ දී එම වෙබ් ලිපියේ ශීර්ෂ මාතෘකාව වෙබ් බ්‍රවුසරයෙහි (Web browser) ශීර්ෂය මත දිස්විය යුතු වේ. ඒ සඳහා <TITLE> සහ </TITLE> උපදේශ කේත අතර ලිපියේ ශීර්ෂ මාතෘකාව සඳහන් කළ හැකි ය.

<BODY>—</BODY>- වෙබ් ලිපියෙහි බඳ කොටස

වෙබ් අඩවියක ඇති වෙබ් පිටුවක බඳ (Body) හෙවත් ප්‍රධානතම තොරතුරු ඇති කොටස 'Body' යනුවෙන් හඳුන්වනු ලබන අතර, එම පරාසය හඳුනාගැනීම සඳහා <BODY> සහ </BODY> උපදේශ කේත යුගලය භාවිත කරනු ලැබේ. ඉන් අනතුරුව මෙම උපදේශ කේත යුගල අතර වෙබ් ලිපියෙ හි අඩංගු කළයුතු අනෙකුත් තොරතුරු රූප, වීඩියෝපට සහ චලන චිත්‍ර විවිධ ආකාරයට වෙනත් වෙනත් උපදේශ කේත භාවිත කරගනිමින් ඉදිරිපත් කළ හැකි අතර එම තොරතුරු සියල්ල වෙබ් අඩවි භාවිත කරන්නාට දැකගත හැකි ය.

මාතෘකා සහ උපමාතෘකා

<H1>—</H1> <H2>—</H2> **<H3>—</H3><H4>—</H4>** **<H5>—</H5> <H6>—</H6>**	The six levels of text headings ranging from the largest (<H1>) to the smallest (<H6>). Text headings appear in the bold face font.

ALIGN=Option The alignment of the heading. (LEFT|RIGHT|CENTER)

වෙබ් අඩවියක ඇති වෙබ් පිටුවල තොරතුරු, මාතෘකා (Heading) සහ උප මාතෘකා (Sub Heading) යටතේ වෙන් වෙන් වශයෙන් දැක්විය හැකි ය. එසේ බෙදු වෙන් කරන ලද කොටස් සඳහා මාතෘකා හෝ උප මාතෘකා යොදගන්නේ නම් ඒ සඳහා භාවිත කරනු ලබන උපදේශ කේත මෙයින් හඳුන්වනු ලබයි. විශේෂයෙන් ම ලිපියක ව්‍යුහය සාර්ථකව පවත්වාගෙන යාම සඳහා මාතෘකා සහ උප මාතෘකා භාවිත කළයුතු අතර එසේ ලිපි සකස් කිරීමේ දී ඒ සඳහා භාවිත කළ හැකි උපදේශ කේත 6ක් <H1>, <H2>, <H3>,<H4>, <H5>, සහ <H6> වශයෙන් නිර්මාණය කර ඇත.
<H1> ප්‍රධාන මාතෘකාව සඳහා යොදු ගන්නා අතර ඊට අනු පිළිවෙලින් <H2> වල සිට <H6> දක්වා වූ උපදේශ කේත උප මාතෘකා සඳහා යොදු ගත හැකි ය. <H1> හි යොදු ඇති අනු ලක්ෂණය ප්‍රමාණයෙන් විශාලවන අතර <H6> හි යොදු ගනු ලබන අනු ලක්ෂණය ප්‍රමාණයෙන් කුඩා වේ. එසේම මෙම මාතෘකා තද කළු අකුරින් දිස් වේ.

<H1> First Level Heading </H1>
<H2> Second Level Heading </H2>
<H3> Third Level Heading </H3>
<H4> Fourth Level Heading </H4>

<H5> Fifth Level Heading </H5>
<H6> Sixth Level Heading </H6>

මාතෘකා සහ උප මාතෘකා සඳහා මෙම උපදේශ කේත භාවිත කිරීමේ දී එම උපදේශ කේත අතරේ වෙනත් කිසිදු උපදේශ කේතයක් භාවිත කළ නොහැකි ය. මාතෘකා හෝ උප මාතෘකා වම් පසට, දකුණු පසට හෝ මධ්‍යයට සමාන්තරව පෙළ ගැස්වීම 'ALIGN' අනු උපදේශ කේතය මගින් කළ හැකි අතර මධ්‍යයට සමාන්තරව පෙළ ගැස්වීමට අවශ්‍ය වූ විටක දී එය <H1 ALIGN = CENTER> මාතෘකාව </H1> ලෙස යොදාගත යුතුය. අනු උපදේශ කේත සඳහා සුදුසු අගයන් ආදේශ කළ යුතු බව සැලකිල්ලට ගතයුතු අතර <P> උපදේශ කේතය සඳහා භාවිත කළ හැකි 'ALIGN' අනු උපදේශ කේතය සඳහා 'LEFT', 'RIGHT' සහ 'CENTER' ලෙස අගයන් 3 ක් ඇත.

ඡේද සහ පේළි වෙන් කිරීම

<P>—</P> - The <P> tag defines the beginning and ending of a paragraph of text.

ALIGN=Option The alignment of the text in the paragraph (LEFT| RIGHT| CENTER)

සාමාන්‍යයෙන් භාවිත කරනු ලබන 'Line Break' (පේළි වෙන්කිරීම) හෝ 'Tab', 'HTML' ලිපියක දී යොදා ගත නොහැකි වන අතර ඒ සඳහා විශේෂිත වූ උපදේශ කේත භාවිත කිරීමට සිදු වේ. ඒ අනුව ඡේද වෙන් කිරීමේ දී <P> (for Paragraph) උපදේශ කේතය භාවිත කළ යුතු වේ. <P> උපදේශ කේතය භාවිත කිරීමේ දී ඡේද අතර හිස් පේළියක් ඇති කරනු ලබයි. මෙයින් අදහස් කරනු ලබන්නේ ඡේද අතර ඉඩක් ඇති කරනු ලැබීමයි.

```
<HTML>
<HEAD>
<TITLE> Test page </TITLE>
</HEAD>
<BODY>
<H1> Paragraph and line Demo </H1>
<P> This is first Paragraph  </P>
<H2> This is heading 2 </H2>
<P> The paragraph two<BR>The paragraph three </P>
</BODY>
</HTML>
```


 - පේළි වෙන් කිරීම

එක් පේළියකින් තවත් පේළියකින් වෙන්කර ගැනීමට අවශ්‍ය වූ විටක දී එම පේළි වෙන් කිරීම සඳහා
 (BR for Line BReak) යොදා ගනු ලබන අතර, මෙහි දී හිස් පේළියක් ඇති නොකර ඊළඟ පේළියට ලිපියේ ඉතිරි කොටස ගෙන යනු ලබයි. <P> මගින් ඡේද දෙක අතර හිස් පේළියක් ඇති කල ද
 මගින් එවැනි හිස් පේළියක් ඇති කරනු නොලබයි.

<P> සහ
 අනු පිළිවෙලින් හුදෙක් ඡේද සහ පේළි වෙන් කිරීම සඳහා යොදා ගනු ලබන අතර ඒ නිසා ම අනෙකුත් උපදේශ කේත සඳහා අවශ්‍ය වන අවසාන කිරීමේ දී භාවිත කරනු ලබන </P> හෝ </BR> මේ සඳහා අනිවාර්ය නොවේ.

සටහන් ඃ

1. වෙබ් පිටු නිර්මාණය කිරීමේ දී එම වෙබ් පිටුව සකස් කරන අවස්ථාවේ ඔබගේ පහසුව තකා හිස් පේළි හෝ ඉඩ හැකි ප්‍රමාණයට තබා ගතහැකි වන අතර එම හිස් පේළි හෝ ඉඩ වෙබ් අඩවි තුළ දී ඔබට දැක ගත හැකිවන්නේ නැත. ඒ නිසා ම ඔබට හිස් පේළි තබා ගැනීමට අවශ්‍යවන අවස්ථාවල දී <P> හෝ
 භාවිත කළ හැකිවන අතර එමගින් ලිපියේ යථා ස්වරූපය විදහා දක්වන ආකාරයට අවශ්‍ය වන හිස් පේළි හා අනෙකුත් ඉඩ තැබීම සඳහා අවශ්‍ය ආකාරයට <P> සහ
 යොදා ගැනීමට අමතක නොකරන්න.
2. ඡේද සකස් කිරීමේ දී හෝ එම ඡේද පෙළ ගස්වා පිහිටුවීමේ දී මධ්‍යයට සමාන්තරව (Align Center) හෝ දකුණු පැත්තට සමාන්තරව (Align Right) හෝ එසේ නැතිනම් වම් පැත්තට සමාන්තරව (Align Left) පෙළ ගැස්සවීමට සිදු වේ. ඒ සඳහා අවශ්‍යවන උපදෙස් සඳහා <P> උපදේශ කේතය සමඟ යොදා ගත හැකි අනු උපදේශ කේත යොදා ගත හැකි ආකාරය පහත දක්වා ඇත.

```
<P ALIGN = "RIGHT">
<P ALIGN = "CENTER">
<P ALIGN = "LEFT">
```

(මෙහි දී ඔබ මතක තබාගත යුතු දෙයක් වන්නේ භාවිත කරන ඉංග්‍රීසි අමෙරිකානු ඉංග්‍රීසි විය යුතු බවය. උදාහරණයක් ලෙස "CENTRE" භාවිත කළ නොහැක. මෙය මහා බ්‍රිතාන්‍ය ඉංග්‍රීසියෙ හි භාවිත වන ආකාරයයි.)

සැලකිය යුතුයි ඃ-
• වෙබ් පිටුව නිර්මාණය කිරීමේ දී ඔබ විසින් භාවිත කරන ලද තොරතුරු ඔබ කැමති සිංහල, දෙමළ හෝ ඉංග්‍රීසි භාෂාවෙන් සකස් කළ හැකි ය.
• ඔබ විසින් දෘඩ තැටියෙහි තැන්පත් කරගනු ලැබූ වෙබ් ලිපියෙහි දිගුව (File Extension) .html හෝ .htm විය යුතු ය. අන්තර්ජාල සමාජය තුල බෙහෙවින් ම භාවිත වන්නේ .html බව ද සඳහන් කළ යුතු ය.

<u>තිරස් රේඛා යොදා ගැනීම</u>

```
<HTML>
<HEAD>
<TITLE>Test Page </TITLE>
</HEAD>
<BODY>
<H1>Horizontal Rules</H1>
<HR COLOR=#FF0000>
<P><HR size= 4 width=80% COLOR=#CCFFCC>
<P><HR size=10 width=40>
<P><HR size=10 width=40 ALIGN=”LEFT”>
<P><HR size= 5 width=20% ALIGN=”RIGHT”>
<P><HR size= 4 width=80% NOSHADE>
<P><HR size=10 width=40 NOSHADE>
</BODY></HTML>
```

වෙබ් ලිපියක බාහිර ස්වරූපය අලංකාර කිරීම සඳහා තිරස් රේඛා (Horizontal Rules) යෙදීම බොහෝ අවස්ථාවල දී ඔබ දැක ඇත. මේ සඳහා භාවිත කළයුතු උපදේශ කේතය ලෙස <HR> (Horizontal

Rule) යොදගත හැකි ය. මෙම තිරස් රේඛා වෙබ් ලිපි අලංකාර කිරීම සඳහාත් එක් එක් කොටස් වෙන් කර දැක්වීම සඳහාත් භාවිත කෙරේ.

මෙහිදී ද උපදේශ කේත අවසාන කිරීමේ දී භාවිත කරනු ලබන </HR> අවශ්‍ය නොවේ. 'Internet Explorer' (අන්තර්ජාල තොරතුරු ලබාගැනීමට භාවිත කරන 'Microsoft' ආයතනය මගින් නිෂ්පාදිත පරිගණක යෙදුම) මගින් විවිධ වූ තිරස් ඉරි (රේඛා) නිර්මාණය කර ගත හැකි ය. ඒ සඳහා භාවිත කළ හැකි අනු උපදේශ කේත 4ක් ඇත. එනම්, 'SIZE' (රේඛාවේ ඝනකම හෙවත් ප්‍රමාණය), 'WIDTH' (රේඛාවේ දිග), 'ALIGN' (රේඛාවේ පෙළගැස්වීම) සහ 'SHADOW' (රේඛාවේ සෙවනැලි යෙදීම) මෙම අනු උපදේශ කේත වේ. මෙයට අමතරව රේඛාවේ වර්ණය වෙනස් කිරීම සඳහා 'COLOR' නම් වූ අනු උපදේශ කේතය ද යොදා ගනු ලැබේ.

තිරස් රේඛාවේ ඝනකම

තිරස් රේඛාවේ මහත වෙනස් කිරීමේ හැකියාව 'SIZE' අනු උපදේශ කේතය මගින් දැක්විය හැකි ය.

උදාහරණ : <HR SIZE = 5>.

තිරස් රේඛාවේ දිග

'WIDTH' අනු උපදේශ කේතය මගින් රේඛාවේ දිග සඳහන් කළ හැකි ය. මෙය ආකාර දෙකකින් පෙන්වා දිය හැකි වන අතර එක් ක්‍රමයක් වන්නේ එම රේඛාවේ දිග 'Pixel' වලින් හෙවත් පරිගණක තිරය මත ඇති තිත් ප්‍රමාණයෙන් දැක්වීම වන අතර අනෙක් ක්‍රමය වන්නේ පරිගණක තිරය මත වෙබ් පිටුව දිස්වන දුර ප්‍රමාණයෙහි ප්‍රතිශතයක් (%) ලෙස දැක්වීම ය.

උදාහරණ :

<HR SIZE = 8 WIDTH = 50> (රේඛාවේ දිග 'Pixel' වලින්)
<HR SIZE = 8 WIDTH = 50%> (රේඛාවේ දිග ප්‍රතිශතයක් ලෙස)

තිරස් රේඛාවේ පෙළගැස්වීම

පරිගණක තිරය මත පතිත වන තිරස් රේඛාවේ පෙළගැස්වීම 'ALIGN' අනු උපදේශ කේතය මගින් සකස් කරනු ලබන අතර එම රේඛාව මධ්‍යයට (CENTER) අනුව සකස් කර ගත හැකි වන අතර, දකුණු පැත්තට (RIGHT) හා වම් පැත්තට (LEFT) අනුව ද පෙළගැස්වීම සිදු කළ හැකි ය.

උදාහරණ :

<HR SIZE = 8 WIDTH = 50 ALIGN = "CENTER">
<HR SIZE = 8 WIDTH = 50% ALIGN = "LEFT">
<HR SIZE = 8 WIDTH = 50 ALIGN = "RIGHT">

මෙම 'ALIGN' අනු උපදේශය කේත යොදා නොගන්නා අවස්ථාවල දී එම තිරස් රේඛාව මධ්‍යයට සමාන්තරව සකස් කරනු ලැබේ.

තිරස් රේඛාවේ සෙවනැලි ඇති කිරීම

'NOSHADE' අනු උපදේශ කේතය භාවිත කරනු ලබන රේඛාව තද කළු පාටින් සටහන් කරනු ලබන අතර සමහර අවස්ථාවල දී සෙවනැලි සහිත ව සකස් කරනු ලැබේ.

උදාහරණ :

<HR SIZE = 10 WIDTH = 40 NOSHADE>

අනුලක්ෂණ මෝස්තර

- <B>—</B> - The <B> tag displays the enclosed text in the bold type.
- <BIG>—</BIG> -The <BIG> tag increased the size of the enclosed text. The exact appearance of the text depends on the browser and the default font size.
- <EM>—</EM> - The <EM> is used to emphasize text. The enclosed text is usually displays in italics
- <I>—</I> - The <I> tag italicizes the enclosed text.
- <SMALL>—</SMALL> - The <SMALL> tag decreases the size of the enclosed text. The exact appearance of the text depends on the browser and the default font size.
- _— - The <SUB> tag displays the enclosed text as a subscript.
- [—] - The <SUP> tag displays the enclosed text as a superscript.
- <TT>—</TT> - The <TT> tag displays text in a fixed width, teletype style font.
- <U>—</U> - The <U> tag underlines the enclose text. The <U> tag should be avoided because it will confide users with hypertext, which is typically underlined.

Example

```
<HTML><HEAD><TITLE>Test Page</TITLE></HEAD>
<BODY>
<H1>Character Emphasis </H1>
<P>this is <EM>Emphasis (Usually Italic)
</EM> text
<P>this is <STRONG>Stronger emphasis
(Usually bold)</STRONG> text
<P>this is <TT>Teletext </TT> text
<P>this is <B>Bold</B> text
<P>this is <I>Italic</I> text
<P>this is <U>Underline</U> text
<P>this is <BIG>BIGGER</BIG> text
<P>this is <SMALL>SMALLER</SMALL> text
<P>this is <SUB>Subscripted</SUB> text
<P>this is <STRIKE>Strike through</STRIKE> text
<P>this is <SUP>Superscripted</SUP> text
</BODY></HTML>
```

'HTML' භාවිතයෙන් සකස් කරන වෙබ් පිටු සඳහා භාවිත කරනු ලබන අනුලක්ෂණ විවිධ ආකාරයෙන් වෙනස් කරමින් වෙබ් පිටුවල බාහිර පෙනුම අලංකාර කළ හැකි ය. මේ සඳහා අනුලක්ෂණ උපදේශ කේත (Character Tag) භාවිත කළ හැකි ය. මෙම උපදේශ කේත අවශ්‍ය වන අක්ෂරයට හෝ

වචනයට හෝ වාක්‍ය කාණ්ඩයකට හෝ වාක්‍යයකට හෝ ඡේදයකට හෝ යොදු ගත හැකි ය. මෙම අනුලක්ෂණ උපදේශ කේත භාවිත කිරීමේ දී එහි අවසානය කොතනදැයි දැක්විය අනිවාර්ය ය.

සමහර ලේඛන මෝස්තර (Document Styles) භාවිත කරනු ලබන ආකාරය අනුව වෙනස් වන අතර සමහර අවස්ථාවල දී මෙම කේත තර්කානුකූල (Logical) ආකාරයකට ක්‍රියා කරනු ලැබේ.

අකුරු වර්ගයෙහි ප්‍රමාණය සහ වර්ණය

<FONT>—</FONT> - The <FONT> tag used to control the appearance of the text it encloses.
SIZE - Value Size of the font in points, it can be absolute or relative. Specifying
SIZE = 5 - sets the font size to 5 points. Specifying
SIZE - +5 set the font size 5 points larger than default tag.
COLOR - Colour The colour of the enclosed text.
FACE - List The font faces of the text. Multiple font face can be specified, separated by
commas. The browser will try to render the text in the order specified by the list.

'HTML' හි <FONT> උපදේශ කේතය සමග එහි අනු උපදේශ කේත භාවිත කරගනිමින් අක්ෂරයන් හි ප්‍රමාණය (Size) සහ වර්ණය (Color) වෙනස් කළ හැකි ය. මෙම අමතර අනු උපදේශ කේත භාවිත කරගනිමින් වර්ණවත් අක්ෂර සහිත වෙබ් පිටු සකස් කිරීමට පහසුකම් සැලසී ඇත.

අකුරු වර්ගයෙහි ප්‍රමාණය

<FONT SIZE = n >

මේ සඳහා ආකාර දෙකක් භාවිත කළ හැකි ය. පළමු ක්‍රමය අක්ෂරයේ ප්‍රමාණය නිශ්චිත වශයෙන් සඳහන් කිරීම වේ. මෙහි 'n' ප්‍රමාණයේ අගය 1 සිට 7 දක්වා වෙනස් වේ.

n හි අගය 1 සහ 2 කුඩා ප්‍රමාණ සඳහාත් (Small Size)
n හි අගය 3 සාමාන්‍ය ප්‍රමාණය සඳහාත් (Normal Size)
n හි අගය 4 සිට 7 දක්වා විශාල ප්‍රමාණ සඳහාත් (Large Size) යොදු ගැනේ.

Example

```
<HTML><HEAD><TITLE>Test Page</TITLE></HEAD>
<BODY>
<H1>Font size and colour</H1>
<FONT SIZE=4 COLOR="#0000FF">
<P>This font is in Blue</FONT>
<FONT SIZE=6>
<P>This font is in size 6</FONT>
<FONT SIZE=3 COLOR="black">
<P>This font is in size 3 and Black
color</FONT>
</BODY></HTML>
```

අනුලක්ෂණ වර්ණය

අක්ෂරවල වර්ණය <FONT> උපදේශ කේත සමඟ 'COLOR' අනු උපදේශ කේතය භාවිත කරමින් වෙනස් කළ හැකි ය.

<FONT COLOR ="#RRGGBB">

අක්ෂරයන් හි වර්ණය වෙනස් කිරීම සඳහා ෂඩ් දශම (Hexadecimal) සංඛ්‍යා 3ක් භාවිත කළ යුතු ය. රතු, කොළ සහ නිල් වර්ණය සඳහා වෙන වෙනම ෂඩ් දශම සංඛ්‍යා භාවිත කළ යුතු වන අතර එම සෑම වර්ණයකම අගය දශම සංඛ්‍යා 0 සිට 255 දක්වා වෙනස් කළ හැකිවන අතර එම දශම සංඛ්‍යා 'HTML' ලිපි සකස් කිරීමේ දී ෂඩ් දශමසංඛ්‍යා වලින් දැක්විය යුතු ය.

ඉහත සඳහන් කරන ලද දශම සංඛ්‍යා 0 සිට 255 දක්වා වූ අගයයන් ෂඩ් දශම සංඛ්‍යා ක්‍රමයට අනුව '00' සිට 'FF' දක්වා වෙනස් වන අතර, ඉහත සඳහන් කරන ලද නිදසුනට අනුව 'RR' (රතු වර්ණය සඳහා), 'GG' (කොළ වර්ණය සඳහා) සහ 'BB' (නිල් වර්ණය සඳහා) සඳහා වෙන් වෙන් ෂඩ් දශම සංඛ්‍යා අගයයන් ආදේශ කළ යුතු ය.

උදාහරණයක් ලෙස ෂඩ් දශම අගයක් වන '0000FF' (0,0 සහ 255 දශම සංඛ්‍යා අගයයන් අනුපිළිවෙළට) නිල් වර්ණය සඳහා භාවිත කරනු ලබන අතර එහි '00' අගය රතු වර්ණය සඳහා ද දෙවන ෂඩ් දශම සංඛ්‍යාව වන '00' අගය කොළ වර්ණය සඳහා ද තුන්වන ෂඩ් දශම සංඛ්‍යාව වන 'FF' අගය නිල් වර්ණය සඳහා ද යොදා ගැනේ.

මෙහි දී රතු වර්ණය සහ කොළ වර්ණය සඳහා '00' යොදා ගෙන තිබීමෙන් එම වර්ණ දෙක යොදා ගෙන නැති බව දැක්වෙන අතර නිල් වර්ණය පමණක් යොදා ගැනේ. නිල් වර්ණය අනෙකුත් වර්ණ සමග සංකලනය වී නොමැති බව සැලකිය යුතු ය.

එසේ ම මෙම ෂඩ් දශම සංඛ්‍යාව වෙනුවට වර්ණයට හිමි නාමය ද භාවිත කළ හැක

උදාහරණයක් ලෙස,

<FONT COLOR ="#8F8FBD"> (Light Steel Blue වර්ණය සඳහා)
<FONT COLOR ="blue"> (නිල් වර්ණය සඳහා)

සටහන් ;

<FONT> උපදේශ කේතය යටතේ අක්ෂරයන් හි වර්ණය සහ එහි ප්‍රමාණය වෙනස් කිරීම සඳහා අදාළ අනු උපදේශ කේත පහත දැක්වෙන ආකාරයට භාවිත කළ හැකි ය.

<FONT COLOR ="White" SIZE = 5>

Pre formatted text භාවිතය

Example

```
<html><head><title> pre formatted </title></head><body>
<pre>
Hello
        Welcome to the world of programming
                                with HTML
                    With the help of this guide
</pre></body></html>
```

10.4 වෙබ් පිටුවක් වැඩිදියුණු කිරීම සඳහා HTML භාෂාවේ ඇති දියුණු ලක්ෂණ භාවිතා කරයි

Tables

Table එකක් සඳහා <table> Tag එක භාවිතා කරයි. Table එකක පේළි (Row) අන්තර්ගත වන අතර පේළි
Data Cell වලින් සමන් විත වේ. පේළි සඳහා <tr> Tag එකද Data Cell සඳහා <td> Tag එකද යොදයි. td
මගින් "Table Data" නිරූපණය කරන අතර Data Cell එකක අකුරු පින්තූර ජේද ආදී දේවල් තිබිය හැක.

Table Tags

Tag විස්තරය

<table> - Table එකක් නිරූපණය කරයි.
<th> - Table එකේ මාතෘකාව සඳහා භාවිතා කරයි.
<tr> - තීරු නිරූපණය කරයි.
<td> - Table එකේ කොටුවක් දක්වයි.

```
<html><head><title>HTML Tables</title></head>
<body>
<table border="1">
<tr>
<td>Row 1, Column 1</td>
<td>Row 1, Column 2</td>
</tr>
<tr>
<td>Row 2, Column 1</td>
<td>Row 2, Column 2</td>
</tr>
</table></body></html>
```

Browser එක තුලින් දකින ආකාරය

Row 1, Column 1	Row 1, Column 2
Row 2, Column 1	Row 2, Column 2

```
<html><head><title>HTML Table Header</title></head>
<body>
<table border="1">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
```

```
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table></body></html>
```

Browser එක තුලින් දකින ආකාරය

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

Cellpadding and Cellspacing Attributes

```
<html><head><title>HTML Table Cellpadding</title></head>
<body>
<table border="1" cellpadding="5" cellspacing="5">
<tr>
<th>Name</th>
<th>Salary</th>
</tr>
<tr>
<td>Ramesh Raman</td>
<td>5000</td>
</tr>
<tr>
<td>Shabbir Hussein</td>
<td>7000</td>
</tr>
</table></body></html>
```

Browser එක තුලින් දකින ආකාරය

Name	Salary
Ramesh Raman	5000
Shabbir Hussein	7000

Colspan and Rowspan Attributes

```
<html>
<head>
<title>HTML Table Colspan/Rowspan</title>
</head>
<body>
<table border="1">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
<tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row
1 Cell 3</td></tr>
<tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>
<tr><td colspan="3">Row 3 Cell 1</td></tr>
</table>
</body>
</html>
```

Browser එක තුලින් දකින ආකාරය

Column 1	Column 2	Column 3
Row 1 Cell 1	Row 1 Cell 2	Row 1 Cell 3
	Row 2 Cell 2	Row 2 Cell 3
Row 3 Cell 1		

Table Backgrounds

```
<head><title>HTML Table Background</title></head><body>
<table border="1" bordercolor="green" bgcolor="yellow">
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
</tr>
<tr><td rowspan="2">Row 1 Cell 1</td><td>Row 1 Cell 2</td><td>Row 1 Cell
3</td></tr>
```

```
<tr><td>Row 2 Cell 2</td><td>Row 2 Cell 3</td></tr>
<tr><td colspan="3">Row 3 Cell 1</td></tr>
</table></body></html>
```

ලැයිස්තු (HTML Lists)

වෙබ් අඩවි සැකසීමේ දී බෙහෙවින් ම අවශ්‍ය වන දෙයක් ලෙස ලැයිස්තු සකස් කිරීම (List) පෙන්වා දිය යුතු ය. තොරතුරු වාර්තා කිරීමේ දී මෙම ලැයිස්තු සකස් කිරීම අත්‍යවශ්‍ය දෙයක් ලෙස බොහෝ දෙනා සලකති.

- අංක සහිත ලැයිස්තු (Ordered Lists)
- අංක රහිත ලැයිස්තු (Unordered Lists)
- විස්තර සහිත ලැයිස්තු (Definition Lists)

List Tags

Tag විස්තරය

<ol> - අංක සහිත ලැයිස්තුවක් දක්වයි

<ul> - අංක රහිත ලැයිස්තුවක් දක්වයි

<li> - ලැයිස්තුවේ ඇති එක අයිතමයක් දක්වයි

<dl>- විස්තර සහිත ලැයිස්තුවක් දක්වයි

<dt> - විස්තර කෙරෙන අයිතමය දක්වයි

<dd> - විස්තරය දක්වයි

අංක සහිත ලැයිස්තු

ලැයිස්තුවේ අයිතම අංක සහිතව දක්වයි. ** Tag** එකෙන් ආරම්භ කරයි. අයිතමයක් ** Tag** එකෙන් ආරම්භවේ.

```
<ol>
<li>Coffee</li>
<li>Milk</li></ol>
```

Browser එක තුලින් දකින ආකාරය

1. Coffee
2. Milk

The type Attribute

<ol type="1"> - Default-Case Numerals.
<ol type="I"> - Upper-Case Numerals.
<ol type="i"> - Lower-Case Numerals.
<ol type="a"> - Lower-Case Letters.
<ol type="A"> - Upper-Case Letters.

Eg 1 .

```
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="1">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ol>
</body>
</html>
```

Eg 2.

```
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="I">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ol>
</body>
</html>
```

Eg 3.

```
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="i">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ol>
</body>
</html>
```

Eg 4

```
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="A">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ol>
</body>
</html>
```

Eg 5

```
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="a">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ol>
</body>
</html>
```

The start Attribute

<ol type="1" start="4"> - Numerals starts with 4.
<ol type="I" start="4"> - Numerals starts with IV.
<ol type="i" start="4"> - Numerals starts with iv.
<ol type="a" start="4"> - Letters starts with d.
<ol type="A" start="4"> - Letters starts with D.

Eg.

```
<html>
<head>
<title>HTML Ordered List</title>
</head>
<body>
  <ol type="i" start="4">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ol>
</body>
</html>
```

අංක රහිත ලයිස්තු

මෙම අයිතම Bullet සමග දක්වයි. Tag එකෙන් ආරම්භවේ. සෑම අයිතමයක්ම <li> Tag එකෙන් ආරම්භවේ.

```
<ul>
<li>Coffee</li>
<li>Milk</li></ul>
```

Browser එක තුලින් දකින ආකාරය

- Coffee
- Milk

ඉහත ලැයිස්තු දෙකේම අයිතම තුලට පේ්ද, රූප ආදිය ඇතුලත් කල හැක.

විස්තර සහිත ලැයිස්තු

මෙය අයිතම ලැයිස්තුවක් නොවන අතර වචන ලැයිස්තුවක් හා ඒවායේ විස්තර අඩංගුවූ ලැයිස්තුවකි. <dl> Tag එකෙන් ආරම්භවේ. <dt> විස්තර කෙරෙන අයිතමය ආරම්භ කරයි. <dd> Tag එකෙන් විස්තරය ආරම්භ වන අතර මේ තුලට පේ්ද, රූප ආදිය ඇතුලත් කල හැක.

```
<dl>
<dt>Coffee</dt>
<dd>Black hot drink</dd>
<dt>Milk</dt>
<dd>White cold drink</dd>
</dl>
```

Browser එක තුලින් දකින ආකාරය

Coffee
Black hot drink
Milk
White cold drink

The type Attribute

```
<ul type="square">
<ul type="disc">
<ul type="circle">
```

Eg. 1

```
<html>
<head>
<title>HTML Unordered List</title>
</head>
<body>
  <ul type="square">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ul>
</body>
</html>
```

Eg. 2

```
<html>
<head>
<title>HTML Unordered List</title>
</head>
<body>
  <ul type="circle">
  <li>Beetroot</li>
  <li>Ginger</li>
  <li>Potato</li>
  <li>Radish</li>
  </ul>
</body>
</html>
```

Html උසුලනය

 උසුලනය මගින් HTML මූලාංග සමූහයක් බාණ්ඩයක් ලෙස සැකසීමට යොදා ගනී.මෙම උසුලනය වෙබ් පිටුවක ඇති අනෙකුත් මූලාංග දැවටිය හැකි බහාලුමක් ලෙස ක්‍රියා කරන අතරම ලේඛනය අංශ වලට බෙදයි.ඔබේ වෙබ් පිටුවෙහි තිබෙන <p >,<h2 >,<img > වැනි උසුලන අනෙකුත් උසුලන වලින් වෙන් කොට තබාගත යුතු නම් ඔබට උසුලනය භාවිත කළ හැකිය.

බොහෝ විට උසුලනය භාවිත කරන්නේ පාදකය(footer), ශීර්ෂකය(header), වම් කොටස , දකුණු කොටස වැනි කොටස්වලට සම්පූර්ණ පිටුවම බෙදීමටයි.

```
<html>
<body>

        <div style="background:#FFCAFF">
         <h3>Heading inside a div</h3>
         <p>Text inside a div.</p>
        </div>

        <div style="background:#66FFFF">
         <h3>Heading inside a div</h3>
         <p>Text inside a div.</p>
        </div>
 </body>
 </html>
```

Heading inside a div

Text inside a div.

Heading inside a div

Text inside a div.

HTML Forms

HTML forms දත්ත server එකට යොමු කිරීමට යොදා ගනී. form එකක ආදාන සංඝටක(input elements) වන checkboxes, radio-buttons, submit buttons වැනි දෑ අඩංගු වේ.

Text Fields

```
<html>
    <form>
        First name: <input type="text" name="firstname" /><br />
        Last name: <input type="text" name="lastname" />
</form>
</html>
```

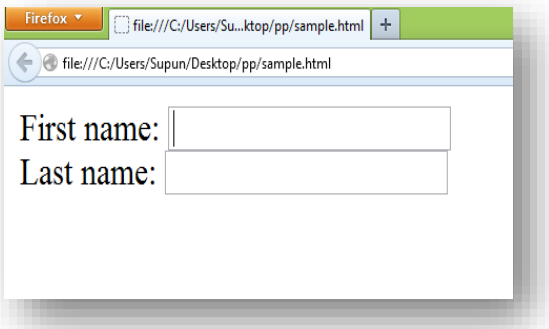
Radio Buttons

```
<form>
        <input type="radio" name="sex" value="male" /> Male<br />
        <input type="radio" name="sex" value="female" />          Female
</form>
```



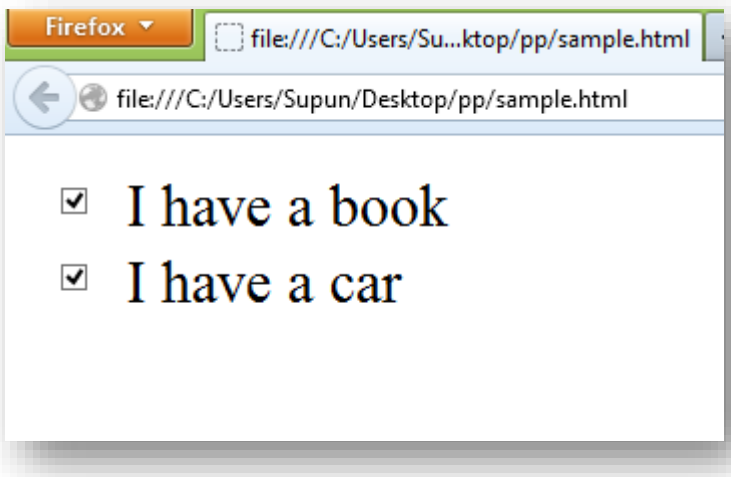
Check boxes

(Checkboxes එකේදී ඔබට එක් පිළිතුරක් හෝ පිළිතුරු කීපයක් තෝරාගත හැකිය)




```
<form>

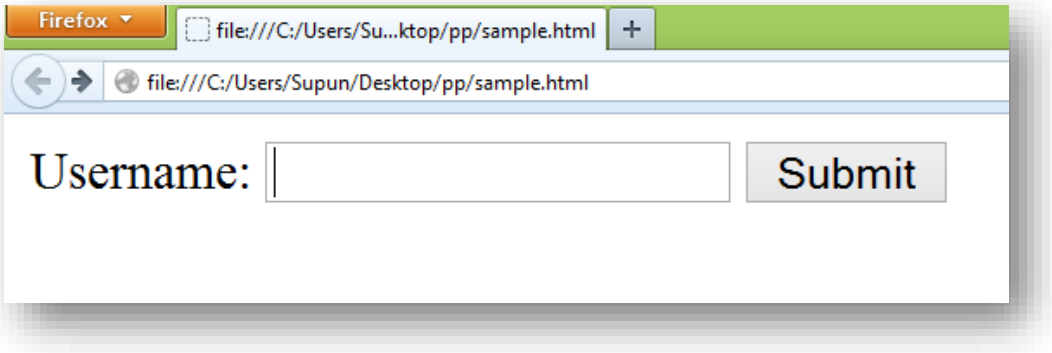
<input type="chec kbox" name="book"
value="book" />
        I have a book<br />
                        <input type="chec kbox" name="vehicle" value="Car"
                        I have a car />
                </form>
```



Submit Button

(Server එකට දත්ත යැවීමට Submit Button එක යොදා ගනී. PHP අන්තර්ගත කොටසකි.)

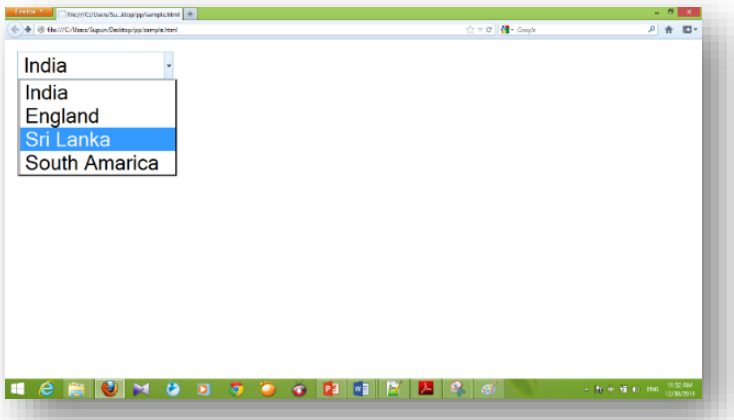
```
<html>
    <form name="input" action="uoc.php" method="get">
        Username: <input type="text" name="user" />
        <input type="submit" value="Submit" />
    </form>
</html>
```



Drop down list

(පහත උදාහරණය සලකා බැලූ විට මෙය තේරුම්ගැනීමට හැකිවේවි.මේ සඳහා select යන එක tag යොදා ගනී.)

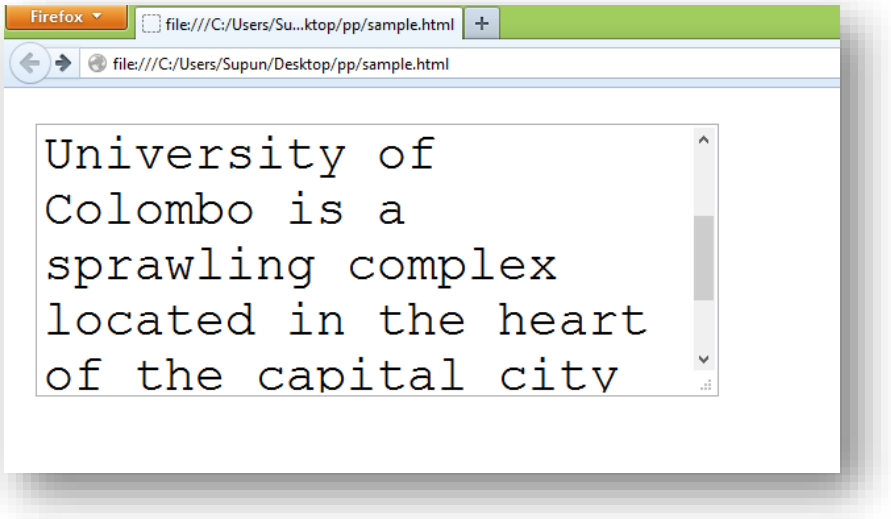
```
<html>
    <select>
        <option>India</option>
        <option>England</option>
        <option>Sri Lanka</option>
        <option>South Amarica</option>
    </select>
</html>
```



Text area

(Text එකක යමක් ලිවීමට ඇති ඉඩ කොටසයි)

```
<html>
    <textarea rows="2" cols="20">
        The oldest University in Sri Lanka, the University of Colombo is a
    sprawling complex located in the heart of the capital city of Colombo.
    </textarea>
    <html>
```



HTML <meta> Tag

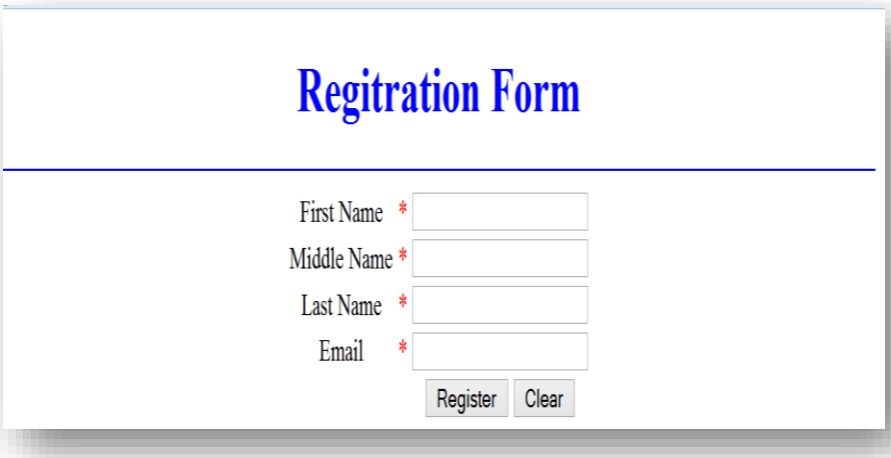
HTML Meta data පිලිබඳ විස්තර කරයි.
මෙහි දන්න මගින් out put එකක් web browser තුලින් පෙන්නුම් නොකරයි.
උදාහරණ

```
<head>
    <meta name="description" content="Free Web tutorials">
    <meta name="keywords" content="HTML,CSS,XML,JavaScript">
    <meta name="author" content="Stale Refsnes">
    <meta charset="UTF-8">
    <meta http-equiv="refresh" content="30">
</head>
```

Form සඳහා උදාහරණයක්.

```
<html> <head>
    <title>Registration Form</title>
    <link href="main.css" rel="stylesheet" />
    <script src="main.js"></script>
        </head>
    <body>
        <form action="tt.php" id="regForm" onsubmit="return validate();">
        <div id="mainDiv">
        <h1 class="heading1">Regitration Form</h1>
        <hr color="blue" />
```

```
<table align="center">
        <tr>
                <td>First Name</td>
                <td><span class="req">*</span></td>
                <td><input type="text" id="fName" /></td>
        </tr>
        <tr>
                <td>Middle Name</td>
                <td><span class="req">*</span></td>
                <td><input type="text" id="mName" /></td>
        </tr>
        <tr>
                <td>Last Name</td>
                <td><span class="req">*</span></td>
                <td><input type="text" id="lName" /></td>
        </tr>
        <tr>
                <td>Email</td>
                <td><span class="req">*</span></td>
                <td><input type="text" id="email" /></td>
        </tr>
        <tr>
                <td></td>
                <td></td>
                <td>
                        <input type="submit" value="Register" />
                        <input type="reset" value="Clear" />
                </td>
        </tr>
</table></div></form></body></html>
```



වෙබ් පිටු නිර්මාණයට සම්පාදන මෙවලම් භාවිතය

වෙබ් පිටු සම්පාදන මෙවලම්

වෙබ් අන්තර්ගතය නිර්මාණය කිරීම සඳහා වෙබ් කතෘ මෙවලම් භාවිතා කරන අතර ඔබේ පරිගණකයට බාගත කළ හැකි හෝ මාර්ගගතව පිවිසිය හැකි පුළුල් පරාසයක මෘදුකාංග වැඩසටහන් ආවරණය කරයි. ලෝක ව්‍යාප්ත වෙබ් සම්මේලනය නොහොත් W3C, වෙබ් ප්‍රවේශය සඳහා මූලික කර්මාන්ත ප්‍රමිතියක් නිර්මාණය කරන වෙබ් කතෘ මෙවලම් සඳහා මාර්ගෝපදේශ නිකුත් කරයි.

මෙය ආකාර කිහිපයක් යටතේ බෙදා වෙන් කළ හැක

- Word Processors
- Desktop Publishing Programs
- Online Web Page Builders
- HTML Editors
- Plain Text Editors

Word Processors

මයික්‍රොසොෆ්ට් වර්ඩ්, වර්ඩ්පර්ෆෙක්ට් හෝ ඕපන් ඔෆිස් රයිටර් වැනි වර්ඩ් ප්‍රොසෙසර ලබා ගත හැකි වඩාත් ජනප්‍රිය වෙබ් කතෘ මෙවලම් වේ. පරිශීලකයින්ට මුද්‍රණය කළ හැකි ලේඛනයක් සේම වෙබ් පිටුවක් නිර්මාණය කර එය HTML ආකෘතියෙන් සුරැකිය හැකි අතර ඉක්මන් හා පහසු වෙබ් පිටුවක් නිර්මාණය කළ හැකිය. පරිශීලකයින් සාමාන්‍යයෙන් ඔවුන්ගේ පරිගණකවල වචන සකසනය ගැන හුරුපුරුදු බැවින්, එකම වැඩසටහනක් සමඟ HTML පිටු නිර්මාණය කිරීම අඩු ඉගෙනුම් වක්‍රය නියෝජනය කරයි. මේවා සාමාන්‍යයෙන් ඔබ දකින දෙයක අන්තර්ගතය වන්නේ ඔබට ආකෘතිය ලැබීමයි, නැතහොත් WYSIWYG යන්නයි, එයින් අදහස් වන්නේ තිරය මත පිටුව දිස්වන ආකාරය එය සබැඳිව පෙනෙන විට එය දිස්වන ආකාරයයි.

Desktop Publishing Programs

Adobe InDesign සහ Scribus වැනි ඩෙස්ක්ටොප් ප්‍රකාශන වැඩසටහන් සැලසුම් කර ඇත්තේ පුවත්පත්, සඟරා, පොත් සහ වෙබ් පිටු වැනි ද්‍රව්‍ය නිෂ්පාදනය කිරීම සඳහා ය. වචන සකසනයන් මෙන්, ඩෙස්ක්ටොප් ප්‍රකාශන වැඩසටහන් WYSIWYG අතුරු මුහුණතක් සපයයි. පිටු සැකැස්ම සහ ශෛලීය අංග වැනි ඔවුන්ගේ උසස් වෙබ් කතෘ විකල්පයන් පරිශීලකයින්ට පිටුවේ පෙනුම කෙරෙහි වැඩි පාලනයක් ලබා දෙයි. මෙම වැඩසටහන් රූප, ග්‍රැෆික් හෝ ශ්‍රව්‍ය ලිපිගොනු වැනි බහුමාධ්‍ය වස්තු සඳහා ද සහාය වේ. සම්පූර්ණ කරන ලද පිටු HTML සහ CSS ගොනු දෙකටම පරිවර්තනය කළ හැකිය.

Online Web Page Builders

වෙබ් අඩවි සත්කාරක අඩවි සාමාන්‍යයෙන් තම ගනුදෙනුකරුවන්ට ඔවුන්ගේ වෙබ් පිටු මාර්ගගතව නිර්මාණය කිරීම සහ පවත්වාගෙන යාම සඳහා බොහෝ වෙබ්-කර්තෘ මෙවලම් විකල්ප ලබා දෙයි. මෙවලම් වලට වෙබ් පිටු සාදන්නන්, සාප්පු පද්ධති, ශ්‍රව්‍ය / දෘශ්‍ය සංස්කාරකවරුන් සහ වසම් විකල්ප ඇතුළත් කළ හැකිය. වචන සැකසීම, ග්‍රැෆික් සංස්කරණය, සැකිලි සහ පිරිසැලසුම් යෝජනා ක්‍රම ඇතුළු බොහෝ වෙබ් කතෘ මෙවලම් තනන්නන් විසින් අන්තර්ගත කර ඇත. වෙබ් පිටු සාදන්නන්ට ප්‍රධාන සංස්කරණ විකල්ප දෙකක් ඇත: HTML හෝ HTML නොවන අතුරු මුහුණතක්. සීමිත HTML දැනුමක් ඇති පරිශීලකයින්ට HTML නොවන අතුරුමුහුණත භාවිතා කර පිරිසැලසුම් නිර්මාණය කිරීම සඳහා අයිතම අතහැර දැමීමට සහ අන්තර්ගතය ටයිප් කිරීමට පෙළ විකල්පය භාවිතා කළ හැකිය.

HTML Editors

ඇඩෝබ් ඩ්‍රීම්වීවර් වැනි HTML සංස්කරණ වැඩසටහන් යනු ලබා ගත හැකි වඩාත්ම ප්‍රබල වෙබ් කතෘ මෙවලම් වේ. වාණිජ වෙබ් අඩවි නිර්මාණය කිරීම සඳහා ඒවා සාමාන්‍යයෙන් වෘත්තීය වෙබ් නිර්මාණකරුවන් විසින් භාවිතා කරනු ලැබේ. බොහෝ HTML කතුවරුන් පරිශීලකයින්ට HTML හෝ HTML නොවන අතුරුමුහුණත් ලබා දීමේදී වෙබ් පිටු සාදන්නන්ට සමාන ය. HTML නොවන අතුරුමුහුණත පරිශීලකයාට වෙබ් පිටුව අන්තර්ජාලයට උඩුගත කරන විට එය පෙනෙන්නේ කෙසේදැයි බැලීමට ඉඩ දෙයි. HTML, CSS, JavaScript හෝ XML ඇතුළුව

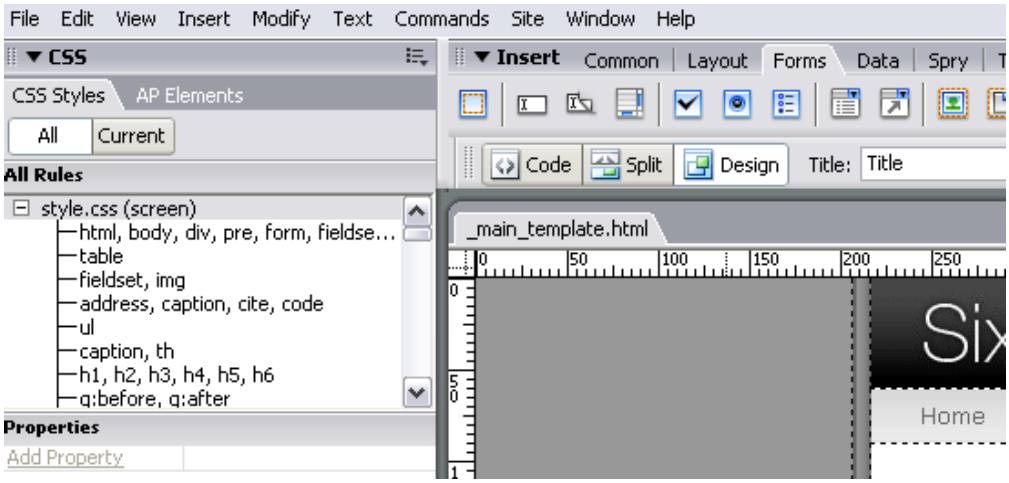
වචන සකසනයක් වැනි සරල පෙළ ලේඛනයක අමු කේත ටයිප් කිරීමට HTML සංස්කාරකවරුන් භාවිතා කරයි. බොහෝ කාර්යයන් සිදු කරනු ලබන්නේ සාදන ලද පෙළ සංස්කාරකයක් භාවිතා කරමිනි. HTML සංස්කාරකවරුන්ට HTML වලංගු කිරීමේ පරීක්ෂකවරුන් සිටින අතර ඒවා වෙබ් පිටුවක් හරහා ක්‍රියාත්මක වන අතර සලකුණු දෝෂ සහ ප්‍රවේශ්‍යතා වලංගු කිරීමේ ගැටළු සඳහා පරීක්ෂා කරයි.

Plain Text Editors

නොට්පෑඩ් වැනි මූලික පෙළ සංස්කාරකවරු කේතය හුරුපුරුදු අයට ප්‍රයෝජනවත් වෙබ් කතෘ මෙවලමක් ද වේ. වචන සකසන හෝ ඩෙස්ක්ටොප් ප්‍රකාශන වැඩසටහන් මෙන් නොව, සරල පෙළ සංස්කාරකවරු ලේඛනයේ පෙනෙන දේට අතිරේක කේත අදාළ නොකරති. යාවත්කාලීනයන් අවශ්‍ය වන සම්පූර්ණ කරන ලද පිටුවලට ඉක්මණින් සංස්කරණ සිදු කිරීම සඳහා සරල පෙළ සංස්කාරකවරුන් ද ප්‍රයෝජනවත් වේ.

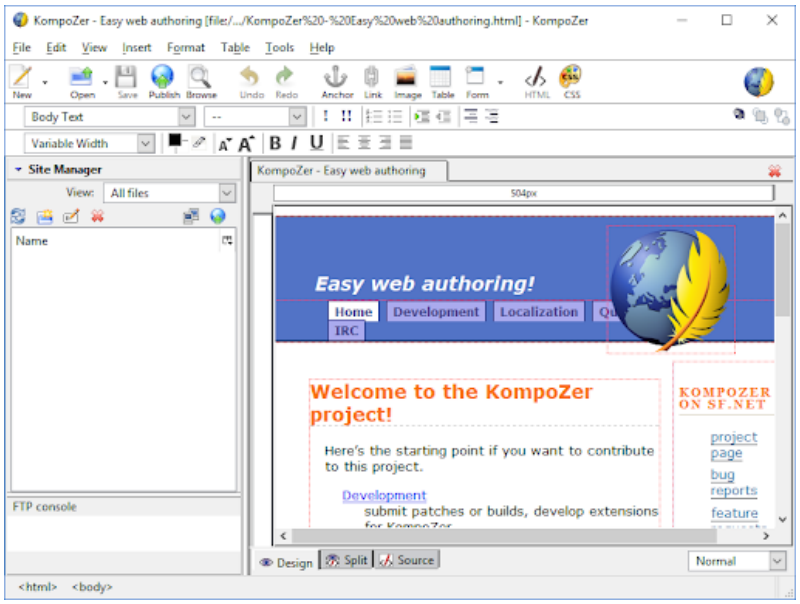
වෙබ් පිටු සම්පාදන මෘදුකාංග

1. Dreamweaver



ඇඩෝබ් ඩ්‍රීම්වීවර් යනු මැක් සහ වින්ඩෝස් මෙහෙයුම් පද්ධති සඳහා ලබා ගත හැකි වෙබ් සංවර්ධනය සඳහා වන වාණිජ යෙදුමකි.

මෙම වෙබ් පිටු සැලසුම් කිරීමේ මෙවලම මෙවලම් සහ විකල්පයන්ගෙන් සමන්විත වන අතර ඩ්‍රීම්වීවර් ෆොටෝෂොප් වැනි ජනප්‍රිය ඇඩෝබ් නිෂ්පාදන සමඟ තදින් සම්බන්ධ වන අතර ග්‍රැෆික් සංරචක ඉක්මන් හා පහසුවෙන් යාවත්කාලීන කිරීම සහ සංස්කරණය කිරීම සඳහා ස්මාර්ට් වස්තු බෙදා ගැනීමට ඔබට ඉඩ සලසයි.



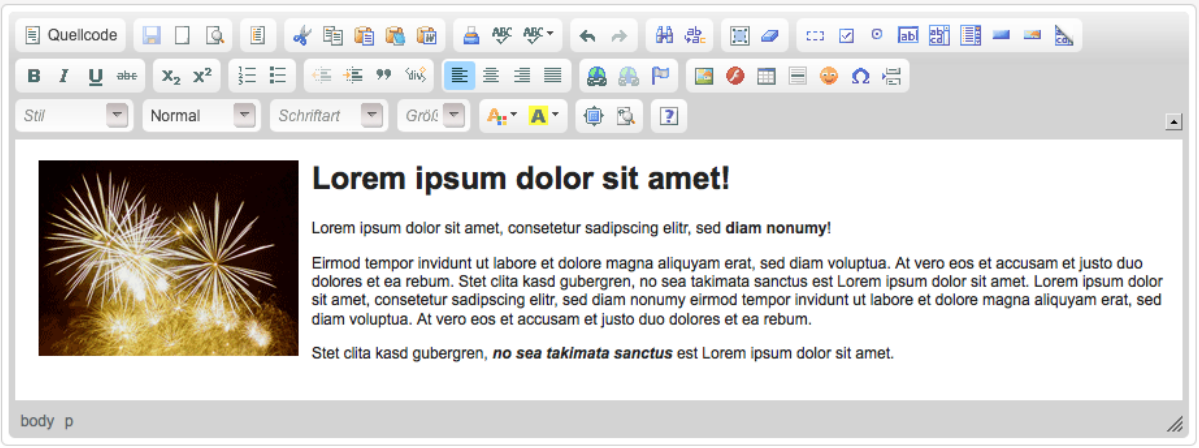
2.Kompozer

KompoZer යනු Nvu සංස්කාරකය මත පදනම් වූ අත්හිටවූ විවෘත මූලාශ්‍ර WYSIWYG HTML සංස්කාරකයකි, එය මොසිල්ලා යෙදුම් කට්ටලයේ රචනා සංරචකයෙන් උපුටා ගන්නා ලද්දකි. කොම්පෝසර් ප්‍රජාව විසින් මෙහෙයවන ලද ව්‍යාපෘතියක් ලෙස සංවර්ධනය කරන ලද අතර

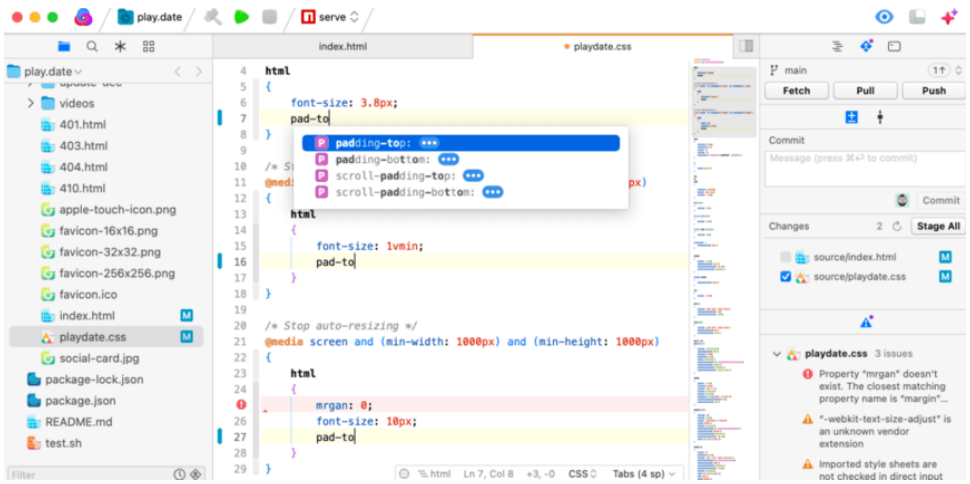
එය Sourceforge හරහා සංවර්ධනය සම්බන්ධීකරණය කරයි

3.CKEditor

CKEditor යනු වෙබ් පිටු හෝ සබැඳි යෙදුම් තුළ කෙලින්ම අන්තර්ගතය ලිවීමට ඉඩ සලසන WYSIWYG පොහොසත් පෙළ සංස්කාරකයකි. එහි මූලික කේතය ජාවාස්ක්‍රිප්ට් හි ලියා ඇති අතර එය වැඩි දියුණු කර ඇත්තේ සීකේසෝර්ස් විසිනි. CKEditor විවෘත මූලාශ්‍ර සහ වාණිජ බලපත්‍ර යටතේ ලබා ගත හැකිය



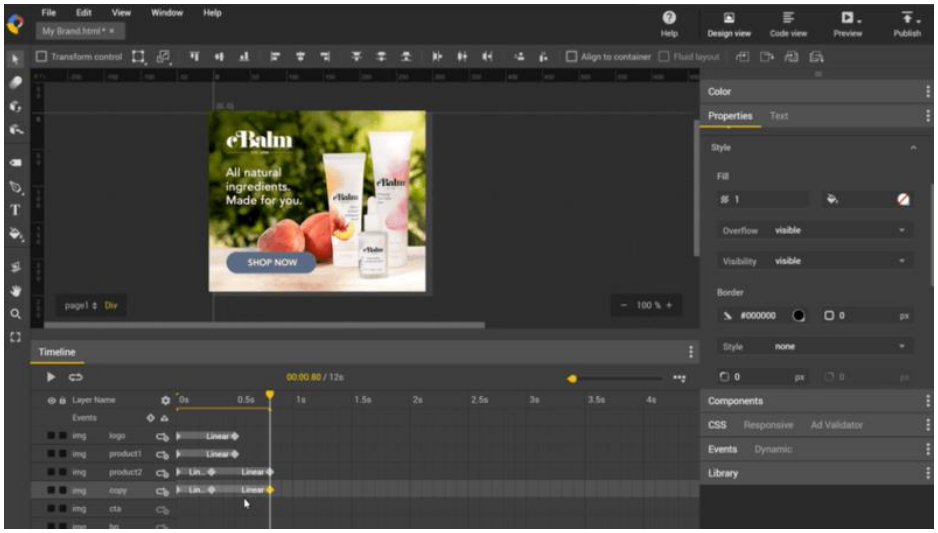
4. Nova



නෝවා යනු වෙබ් සංවර්ධනයක් වන Panic හි නව සහ වැඩිදියුණු කළ අනුවාදයකි

5.Google Web Designer

ගූගල් වෙබ් නිර්මාණකරු යනු අන්තර්ක්‍රියාකාරී හා ආකර්ශනීය HTML5 මත පදනම් වූ මෝස්තර නිර්මාණය කිරීම පහසු කරන නිදහස් වෙබ් පිටු සැලසුම් කිරීමේ මෙවලමකි. ඊටත් වඩා හොඳයි, ඩෙස්ක්ටොප් සිට ස්මාර්ට්



ෆෝන් දක්වා ඕනෑම උපාංගයක ඔබේ නිර්මාණ (සහ චලන ග්‍රැෆික්ස්) ක්‍රියාත්මක වන බව මෙවලම සහතික කරයි. ඔබට ගූගල් වෙබ් නිර්මාණකරු සමඟ කෙලින්ම HTML, CSS සහ JavaScript සංස්කරණය කළ හැකිය

ප්‍රපතිත විලාස පත්‍ර (CSS - Cascading Style Sheets)

- CSS කියන්නේ Cascading Style Sheets කියන එකයි.
- HTML elements display කරන්නේ කොහොමද කියලා මෙහිදී කතා කෙරයි.
- මෙමගින් එක css file එකක් වෙනස් කිරීමෙන් ඕනෑම වෙබ් පිටුවල පෙනුමේ වෙනස් කම් ඇති කල හැක. එය ඉතාම වාසිදායකවේ.
- ප්‍රධාන වශයෙන් style sheet 3කි.

1. Author styles
   - External style
   - Embedded style
   - Inline style
2. User Style
   (Viewer)

3. User Agent Style

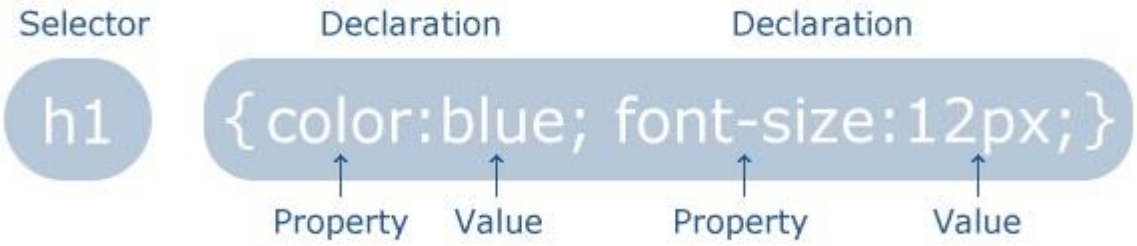
- css rule වල ප්‍රධාන කොටස් 2ක් තිබේ.
  1.1 Selector
  1.2 declaration
- selectorකියන්නේ ඔබට අවශ්‍ය HTML element(body,h1,p,font)එකයි.
- Declaration සෑම එකකම property එකක් හා value එකක් ඇත.
- Property යනු ඔබට වෙනස් කිරීමට අවශ්‍ය style attribute (වර්ණය,අකුරු වල ප්‍රමාණය යන ආදී)එකයි.
- සෑම property එකකටම අගයක් (value) ඇත.
- සෑමවිටම declaration අවසන් විය යුත්තේ semicolon (;)එකකිනි.
- Declaration group එකක් සඟල වරහන් වල අන්තර්ගත විය යුතුය.

```
p {color:red;text-align:center;}
```

- ඉහත code එක කියවීමට තරමක් අපහසු නිසා පහත පරිදි ලිවීම සිදු කරයි.

```
p {
color:red;
text-align:center;
```

- පහත දැක්වෙන්නේ css rule එකකි.



CSS Comments

- css comment කරන අයුරු පහත දැක්වේ.

```
p
{
text-align:center;
/*This is a comment*/
color:black;
font-family:arial;
}
```

Id Selector

- Id එකක් මගින් යම් element එකක් සඳහාම style එකක් අර්ථ දක්වයි.
- මෙය අර්ථ දැක්වීමට #ලකුණ යොදා ගැනේ.
- Id එක සඳහා නම ඉලක්කමකින් ආරම්භ නොකරන්න.
- එය Firfox Browser එක හරහා ක්‍රියාත්මක නොවන බැවිනි.
- පහත උදාහරණයේ id එක "para1" වේ

```
#para1
{
text-align:center;
color:red;
}
```

class Selector

- elements කීපයකට (element group එකකට) style එකක් අන්තර්ගත කිරීමට යොදා ගනී.

- මෙය අර්ථ දැක්වීමට " . " ලකුණ යොදා ගැනේ.
- පහත උදාහරණයේ class එක " center " වේ.

```
.center {
text-align:center;
}
```

එලෙසම එක එකක් වෙනුවෙන්ම මෙය යොදා ගත හැකි අයුරු පහත දැක්වේ.

```
p.center {
text-align:center;
}
```

- ඉහත උදාහරණයට අනුව සියලු p elements මැදට කර වෙබ් පිටුව සැදෙයි.
- Class එක සඳහා නම ඉලක්කමකින් ආරම්භ නොකරන්න..මන්ද එය Internet Explorer එක හරහා පමණක් ක්‍රියාත්මක වන බැවිනි.

Insert CSS

style sheet එකක් අන්තර්ගත කිරීමේ ක්‍රම 3ක් අැත.

- External style sheet
- Internal style sheet
- Inline style

1- External Style Sheet

- මෙය වඩාත්ම සුදුසු වන්නේ එක්තරා style එකක් වෙබ් පිටු කීපයකට යොදා ගන්නේ නම්ය.
- සම්පූර්ණ වෙබ් අඩවියේ පෙනුම එක style sheet file එකකින් වෙනස් කිරීමට ලැබීම මෙහි ප්‍රධානතම වාසියයි.
- <link> යන tag එක මගින් style sheet එක සම්බන්ධ කිරීම සිදුකරයි.

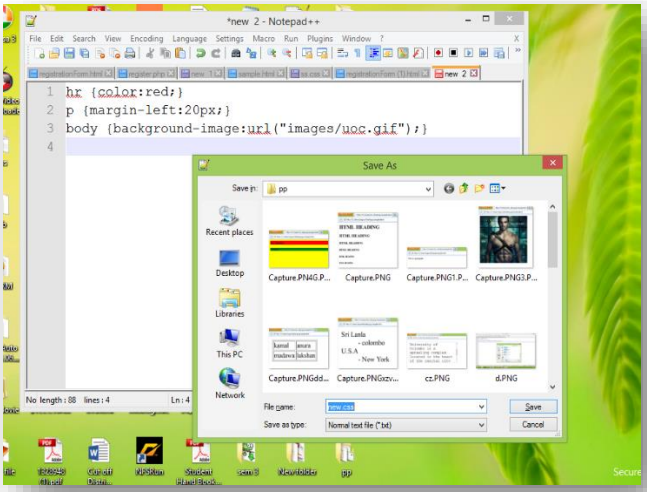
```
<head>
 <link rel="stylesheet" type="text/css"
href="uoc.css"/>
</head>
```

- css ඕනෑම text editor එකක ලිවිය හැකිය.මෙහි කිසිදු html tags අඩංගු නොවේ.ඔබ css file එක.css ලෙස ඔබ යොදන නමට පසුව යොදා save කර ගත හැක.

- margin යන්න සරලව කිවහොත්,අපි යමක් ලියන විට පිටුවක මුල සිටම(කොනේ සිට)ලිවීම සිදුනොකරයි.යම් කිසි ඉඩක් තියා ගැනීමට margin යොදා ගනී.margin ගැන අපි පසුව කතා කරනු ලබයි.

- යම් තැනක ඔබට නොතෙරුනිනම් ඉදිරියේදී ඇති උදාහරණ මගින් සාකච්චා කිරීම මගින් ඔබට ඒවා පැහැදිලි වේවි.

- CSS file එකක් save කිරීම පහතින් පෙන්වා දී ඇත.



2- Internal Style Sheet

- Internal Style Sheet යොදාගනුයේ එක style එකක් යොදා ගන්නේනම් ය.
- HTML page එකකhead section(<head>) එක යටතේ මෙය ලියනු ලැබේ.
- මේ සඳහා <style> tag එක යොදා ගනී.
- පහත උදාහරණය සලකන්න.

```
<head>
<styletype="text/css">
hr {color:blue;}
p {margin-left:20px;}
body {background-
image:url("images/back40.gif");}
</style>
</head>
```

3- Inline Styles

- පහත උදාහරණය මගින් ඔබට මෙය හොදින් වටහා ගත හැක.
- එකම ස්ථානයේ style sheet එක හා අපිට අවශ්‍ය HTML code එකද ඇත.

```
<p
style="color:red ;
margin-left:20px">university
</p>
```

CSS Background Color

වෙබ් පිටුවේ background color එක යොදන ආකාරය මෙහි දැක්වේ.

```
body {
background-color:#b0c4de;
}
```

style sheet එකක නිතරම පහත ආකාර වලින් color දක්වනු ලබයි.

- a HEX value - "#ff0000" ලෙස
- a RGB value -"rgb(255,0,0)" ලෙස
- a color name - "red" ලෙස

Color code කිපයක්

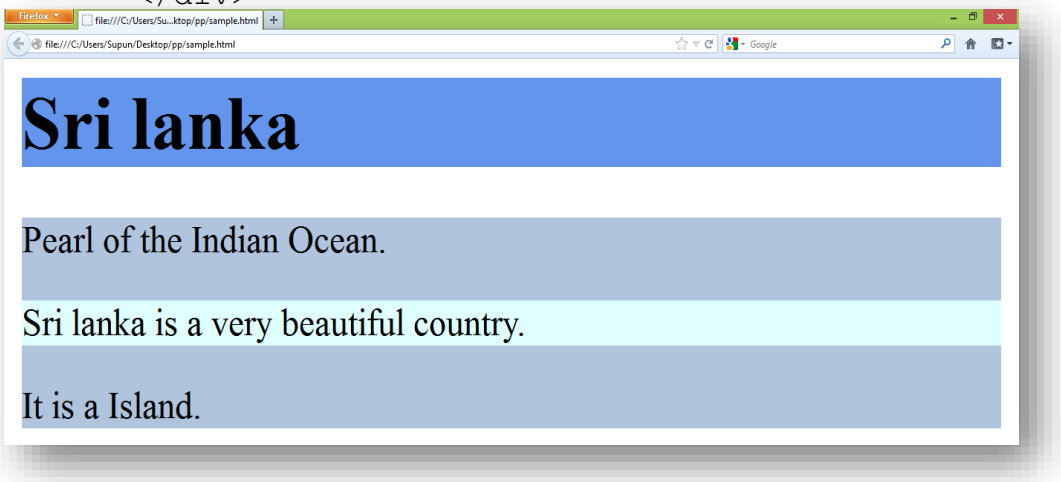
Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)

```
<html>

    <head>
        <style type="text/css">
                h1{
                background-color:#6495ed;}
                p{
                background-color:#e0ffff;}
                div{
                background-color:#b0c4de;}
        </style>
    </head>


        <body>
                <h1>Sri lanka</h1>
                    <div>
                        Pearl of the Indian Ocean.
                        <p>Sri lanka is a very beautiful
                        country.</p>
```

```
                It is a Island.
            </div>
        </body>
</html>
```



CSS Background Image

- යම් කිසි element එකක පසුබිමට පින්තූරයක් යොදන අයුරු මෙහිදී සාකච්ඡා කරනු ලබයි.
- ඔබ යොදනBackground Image එක නිතැතින්ම සම්පූර්ණ element එක පුරාවටම යෙදේ.

Table properties

Purpose	Table Attribute	CSS Property(ies)
Borders	`border`	`border` properties
Spacing inside cell	`cellpadding`	`padding` properties
Spacing between cells	`cellspacing`	`border-spacing`
Width of a table	`width`	`width` and `table-layout` properties
Table framing	`frame`	`border` properties
Alignment	`align`, `valign`	`text-align`, `vertical-alignment` properties

©2007, University of Colombo School of Computing

CSS Background Image සඳහා පහත උදාහරණය සලකන්න(අන්තර්ජාලය මගින් උපුටා ගන්නා ලද උදාහරණයකි.)

```
<html>
    <head>
        <style type="text/css">
body {background-image:url(http://www.cmb.ac.lk/wp-content/uploads/uoc_logo_moto.jpg);}
        </style>
    </head>
    <body>
    <h1> The oldest University in Sri Lanka</h1>
    </body>
```



- ඉහත උදාහරණය කල ඔබට මුළු වෙබ් පිටුව පුරාම background image එක පැතිරී ඇති බව දක්නට ලැබේ.
- ඔබට තිරස්ව,සිරස්ව හෝ background image එක එක වතාවක් පමණක් යොදා ගැනීමටද පුළුවන් කමක් ඇත.
- එලෙසම පින්තූරයේ ස්ථානයද වෙනස් කල හැක.
- පහත උදාහරණය සලකන්න.

ඔබ පින්තූරය තොර ගත්තේ name.jpg නම්

තිරස්ව දැක්වීමට :

```
<style type="text/css">
        body {background-image:url(name.jpg);}
        background-repeat:repeat-x;
   </style>
```

සිරස්ව දැක්වීමට :

```
<style type="text/css">
        body {background-image:url(name.jpg);}
        background-repeat:repeat-y;
   </style>
```

එක වතාවක් පමණක් දැක්වීමට :

```
<style type="text/css">
        body {background-image:url(name.jpg);}
        background-repeat:no-repeat;
   </style>
```

ස්ථානයද වෙනස් කිරීමට :

```
<style type="text/css">
     body {background-image:url(name.jpg);}
     background-repeat:no-repeat;
     background-position:right top;
</style>
```

- ඉහත එක එක උදාහරණය අධ්‍යනය කර වෙනස හොඳින් වටහා ගත යුතුය.
- ඉහත properties වර්ග ගොඩක් කතා කර ඇති අතරේ ඔක්කම properties එකට ලියන්න පුළුවන් විදියක් ඇත.
- එයට shorthand property ලෙස හඳුන්වයි.

```
     body {
     background:#ffffff url('uoc.jpg) no-repeat right
     top;}
```

පහත උදාහරණය සලකන්න.

```
<html> <head>
<style type="text/css">
body{
background:#ffffff url('uoc.jpg') no-repeat right top;
margin-right:100px;}
</style>
</head>
<body>
<h1>UOC</h1>
<p> The oldest University in Sri Lanka.</p>
<p>University of Colombo School of Computing (UCSC) </p>
</body> </html>
```

Short hand property ලිවීමේදී පහත අනුපෙලට ලිවිය යුතුය.

1. background-color
2. background-image
3. background-repeat
4. background-attachment
5. background-position

CSS Text

පහත ඇති CSS color වර්ග ගැන අප ඉහතදීත් කතා කර ඇත.

- a HEX value - "#ff0000" ලෙස
- an RGB value -"rgb (255,0,0)" ලෙස
- a color name - "red" ලෙස

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)

- එමගින් id වෙනම නම් යොදා හදුන්වා ද් ත්බෙ.
- එනම් ඒවාට එකම වර්ණයක් නොදී වෙනස් වර්ණයන් ලබා දේ

```
<html>
<head>
<style type="text/css">
body {color:red;}
h1 {color:#00ff00;}
p.ex {color:rgb(0,0,255);}
p.tr {color:rgb(255,0,255);}
</style>
</head>
<body>
<h1>Sri lanka</h1>
<p>Colombo in Sri lanka</p>
<p class="ex">html in sinhala.</p>
<p class="tr">php in sinhala.</p>
</body>
</html>
```

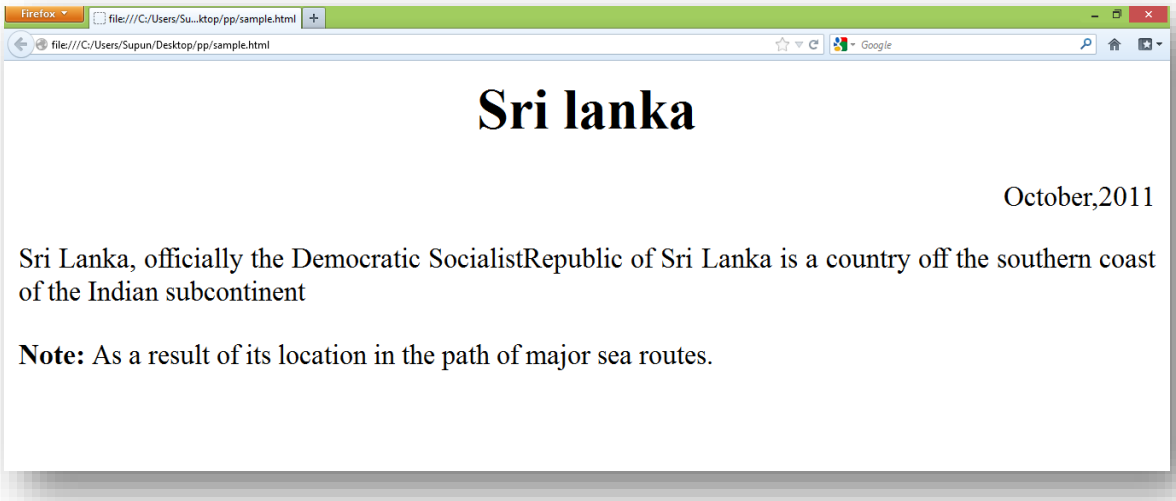


- හොදින් බලන්න.ඉහත උදාහරණයේ යන්න Colombo in Sri lanka රතු පැහැ ගත්තේ කෙසේද?එම id එකට නමක් ලබා දී නැත.එම නිසා body එකෙහි පාට එයට ලැබේ.

Text Alignment

පහත උදාහරණය බලන්න. එහිදී සිදු කර ඇත්තේ අප ලියන text වෙබ් පිටුවේ මැදට ගැනීමට ඒක යොදා ගැනීමයි.

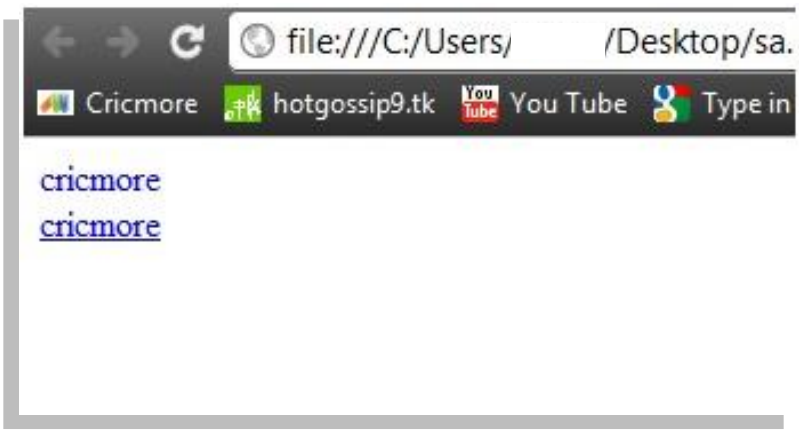
```
<html>
<head>
<style type="text/css">
h1 {text-align:center;}
p.date {text-align:right;}
p.main {text-align:justify;}
</style>
</head>
<body>
<h1>Sri lanka</h1>
<p class="date">October,2011</p>
<p class="main"> Sri Lanka, officially
the Democratic SocialistRepublic of Sri Lanka is a country off
the southern coast of the Indian subcontinent </p>
<p>
<b>Note:</b> As a result of its location in the path of major
sea routes. </p>
</body> </html>
```



Text Decoration

ඇත්ත වශයෙන්ම මෙහිදී සිදුවන්නේ ඔබ ලියන text එක හැඩ කිරීමයි. මෙහිදී වඩාත්ම වැදගත් වන්නේ,ඔබ link එකක් දෙන විට එය underline වී තිබීම මෙමගින් නැති කර දමයි.

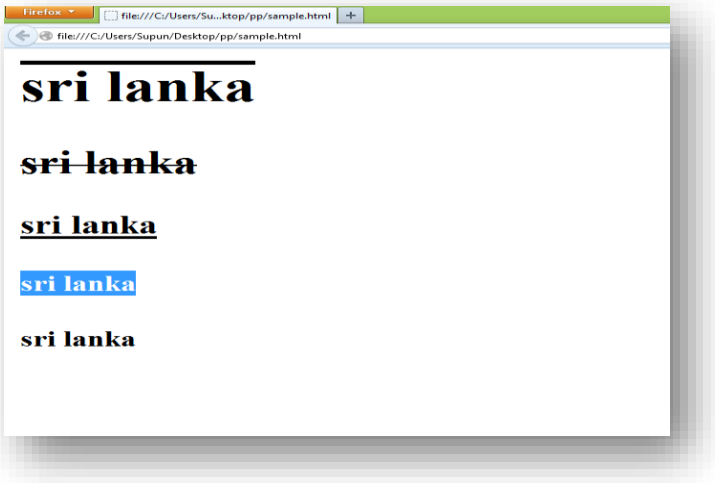
```
<html>
<head>
<style type="text/css">
a.uoc {text-decoration:none;}
</style>
</head>
<body>
<a class="uoc" href="http:// cricmore.com">cricmore</a>
<br/>
<a href="http:// cricmore.com">cricmore</a>
</body>
</html>
```



පහත උදාහරණය සලකන්න.

```
<html>
    <head>
        <style type="text/css">

            h1 {text-decoration: overline;}
            h2 {text-decoration: line-through;}
            h3 {text-decoration: underline;}
            h4 {text-decoration: blink;}
        </style></head>
    <body>
        <h1> sri lanka </h1>
        <h2> sri lanka </h2>
        <h3> sri lanka </h3>
        <h4> sri lanka </h4>
        <p><b> sri lanka </b></p>
    </body>
</html>
```



- CSS කේත ඛණ්ඩ හරහා වෙබ් පිටු තුල අන්තර්ගතයන් වඩා ආකර්ශවත් ලෙස ඉදිරිපත් කිරීමේ හැකියාව පවතී.
- CSS කේත ඛණ්ඩ හරහා html මගින් නැවත නැවතත් ලියන ලද කේත ඛණ්ඩයන් අවම කරගැනීමේ හැකියාවද පවති.

- CSS කේත ඛණ්ඩ හරහා වෙබ් පිටුවක ඇති අන්තර්ගතයන් දර්ශනය විය යුතු ස්ථානයද වෙනස් කර ගත හැකිය

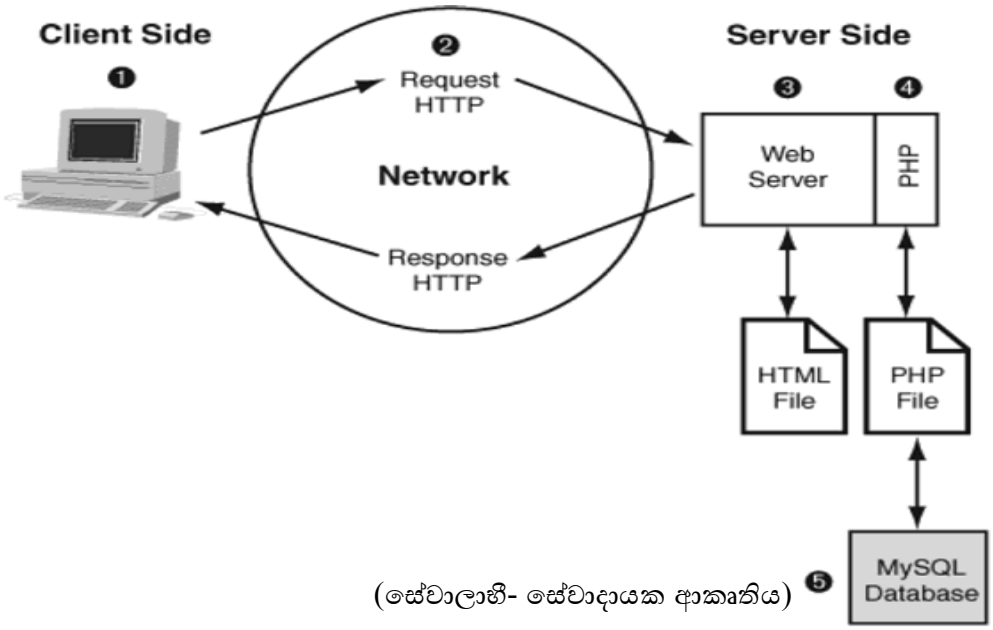
PHP ක්‍රමලේඛණය

මෙම කොටසෙන් PHP (Hypertext Pre Processor) ක්‍රමලේඛන භාෂාවේ අවශ්‍යතාවය, PHP කේත ලිවීම, PHP භාවිතයෙන් දත්ත සමුදායක් කළමනාකරණය සිදු කිරීම, PHP මගින් වෙබ් පිටු නිර්මාණය කිරීම ආදිය පිලිබඳ සාකච්ඡා කෙරේ.

PHP ක්‍රමලේඛන භාවිතයේ අවශ්‍යතාව -

අන්තර්ජාලය ජනප්‍රිය වීමත් සමගම වෙබ් පාදක යෙදුම් සංවර්ධනය කිරීම හා භාවිතය වැදගත් අංගයක් බවට පත්වී ඇත. වෙබ් පාදක යෙදුම් සේවා ලාභී - සේවා දායක ආකෘතිය මත පදනම් වී ඇත.

සේවා දායක පරිගණකය තුල සේවා දායක යෙදුම් මෘදුකාංගය ඇති අතර සේවා ලාභී පරිගණක තුල සේවා ලාභී යෙදුම් මෘදුකාංග ඇත.



(සේවාලාභී- සේවාදායක ආකෘතිය)

මීට පෙර ඉගෙනගත් HTML ක්‍රමලේඛන භාෂාව මගින් සිදු කරනුයේ වෙබ් අතිරික්සුවක් හරහා දත්ත ඉදිද්රිපත් කිරීම පමණි. නමුත් ක්‍රමලේඛයක් මගින් සිදු කරන ගණනය කිරීම්, දත්ත සමුදායක් සමග සම්බන්ධ වී සිදු කරන විවිධ ක්‍රියා ආදිය සිදු කිරීමට HTML භාෂාව යොදාගත නොහැකිය. මෙම මෙහෙයුම් සිදුවන්නේ සේවා දායක පරිගණක අන්තයේ ය.

PHP යනු මෙවැනි යෙදුම් සංවර්ධනය සඳහා යොදාගත හැකි ක්‍රමලේඛන භාෂාවකි. එනම් PHP ක්‍රමලේඛන භාෂාව සේවා දායක පරිගණක පරිසරයක ක්‍රියාත්මක වේ. තවද PHP ගතික වෙබ් පිටු නිර්මාණය කිරීම සඳහා යොදාගන්නා ක්‍රමලේඛන භාෂාවකි.

1. *PHP - පරිගණකයේ ස්ථාපනය කරගැනීම -*

PHP ක්‍රමලේඛන භාෂාව යොදාගෙන කේතකරණය කිරීම සඳහා පළමුවෙන්ම PHP, පරිගණකය තුල ස්ථාපනය කර ගත යුතුය.

www.php.net මගින් PHP බාගතකර ස්ථාපනය කිරීමට හැකි වුවත් PHP, MYSQL හා Apache තනි පැකේජයක් ලෙස පැමිණෙන XAMPP Server හෝ WAMP Server පරිගණකය තුල ස්ථාපනය කරගැනීම වඩාත් පහසු වේ.

XAMPP - www.apachefriends.org

WAMP - www.wampserver.com

එසේම PHP පහසුවෙන් කේතකරණය කිරීම සඳහා කේතකරණ මෘදුකාංගයක් (IDE එකක්) ස්ථාපනය කර ගැනීම අවශ්‍ය වේ. ඒ සඳහා Notepad++, Brackets, Sublime Text යොදාගත හැකිය.

2. *PHP ක්‍රමලේඛනයක මූලික ව්‍යුහය -*

PHP කේතයක් ආරම්භ කරනුයේ <?php උසුලනයෙන් වන අතර අවසන් වනුයේ ?> යන උසුලනයෙනි. අනෙකුත් සියලු කේත ඛණ්ඩ මෙම උසුලන අතර යෙදිය යුතුය. ක්‍රමලේඛනයේ ඕනෑම වගන්තියක් අවසන් කරනුයේ ; මගිනි.

උදා:-

<?php

?>

PHP ක්‍රමලේඛයක ප්‍රතිදාන ලබාගැනීම සඳහා “echo” යන්න භාවිත කරයි.

උදා:-

<?php

echo “Hello World”;

?>

3. *PHP ක්‍රමලේඛයක් ක්‍රියාත්මක කිරීම*

PHP ක්‍රමලේඛයක් පරිගණකය තුල සුරැකිය යුත්තේ .php යන ගොනු දිගුව සහිතවය. එසේම පෙර සඳහන් කල පරිදි php ගොනුවක් ක්‍රියාත්මක වනුයේ සේවාදායක පරිගණක පරිසරයක බැවින් එය අප පෙර ස්ථාපිත කරගත් Wamp Server තුල හෝ XAMPP Server තුල නියමිත පරිදි තැන්පත් කර වෙබ් අතිරික්සුව (Web Browser) හරහා හෝ Command Prompt භාවිතයෙන් ක්‍රියාත්මක කල යුතුය.

උදා:- Wamp Server සැලකූ විට එහි www ෆෝල්ඩරය තුල php ගොනුව තැන්පත් කර Wamp Server ක්‍රියාත්මක කර,

i. Command Prompt මගින් නම් php file_name.php ලෙස යොදා Enter යතුර තද කිරීමෙන්.
ii. වෙබ් අතිරික්සුව (Web Browser) තුල ලිපින තීරුවෙහි localhost/ file_name.php ලෙස යොදා Enter යතුර තද කිරීමෙන්.

4. *php ක්‍රමලේඛයක් HTML ක්‍රමලේඛයක් තුල ඇතුලත් කිරීම*

php කේත ඛණ්ඩයක් ඕනෑම html ලේඛනයක් තුලට ඇතුලත් කර ක්‍රියාත්මක කල හැක.

උදා:-

```
<html>
    <body>
        <h1><? php echo “php code”; ?></h1>
    </body>
</html>
```

5. *PHP ක්‍රමලේඛන භාෂාවේ සංරචක*

5.1 හඳුන්වන (Identifires) -

පරිශීලකයා විසින් ක්‍රමලේඛය තුල භාවිත කරන නම් මේ යටතට ගැනේ.

උදා - විචල්‍ය නාම, ශ්‍රිත සඳහා යෙදූ නම්

පහත නීති වලට අනුව මේවා යෙදිය යුතුය.

1. හඳුන්වනයක් සමන්විත විය යුත්තේ අකුරු, ඉලක්කම්,යටි ඉර සහ අනෙකුත් ASCII අක්ෂර වලිනි.
2. අක්ෂර එකක් හෝ කීපයකින් සමන්විත විය හැකි අතර ආරම්භ විය යුත්තේ ඉංග්‍රීසි අක්ෂරයකින් හෝ යටි ලකුණ මගිනි (Underscore).
3. පෙර යෙදූ පදයන් ගෙන් වෙනස් විය යුතුය.
4. Case Sensitive වේ.

5.2 විචල්‍ය (Variables) -

මෙහෙයුම් සඳහා යොදා ගන්නා දත්ත සසම්භාවී ප්‍රවේශ මතකය (Random Access Memory) තුල තාවකාලිකව ගබඩා කර ගැනීම සඳහා විචල්‍ය භාවිතා කරයි.

විචල්‍යයක් php කේතයක් තුලට ඇතුලත් කරනු ලබන්නේ $ ලකුණ භාවිතයෙනි.

උදා:- $F_Name, $_name

ඇතුලත් කිරීමේදී $ ලකුණට පසුව හිස් තැනක් නොතැබීමට වග බලා ගත යුතුය.

5.3 ආදේශන (Assignments) –

PHP ක්‍රමලේඛයක් තුල භාවිතා වන දත්ත සහ ගණනය කිරීම් වල ප්‍රතිපල විචල්‍යයන් වෙත පැවරීමට යොදාගැනේ. මේ සඳහා ‘=’ ලකුණ භාවිතා කරයි.

උදා:-

```
<?php
$f_name=”Thanuja Bandara”;
$x=12;
?>
```

සංඛ්‍යාත්මක අගයන් ආදේශයෙදී “” යෙදීම අවශ්‍ය නොවේ.

5.4 ප්‍රකාශන (Expressions) –

echo යන විධානය භාවිත කොට ප්‍රතිදාන ලබා ගත හැකි බව අප දනිමු. යම් කිසි ප්‍රකාශයක් එලෙසම ප්‍රතිදානය කර ගැනීම සඳහා එය “” තුල යෙදිය යුතුය. විචල්‍යයන් තුල ඇති අගය ප්‍රතිදානය කර ගැනීමට එය “” භාවිත නොකර යොදයි.

උදා:-

```
<?php
$f_name=”Thanuja Bandara”;
$age=18;
echo “Name is”, $f_name, “and”,”age is”,$age;
?>
```

ප්‍රතිදානය වනුයේ -

Name is Thanuja Bandara and age is 18

යන්නයි.

5.5 විවරණ(Comments) –

කේතකරණයේදී එක් එක් කේත ඛණ්ඩය ලියා තිබෙන්නේ කුමන අරමුණකින්ද යන්න දැක්වීම සඳහා විවරණ භාවිතා කරයි.

ආකාර දෙකකින් යෙදිය හැක.

1. තනි පෙළ විවරණ (Single Line Comments).
   උදා:- //Single Line Comment හෝ
   # Single Line Comment
2. බහු පෙළ විවරණ (Multi Line Comments).
   උදා:- /*Multi line comments
   can be written like this*/

5.6 දත්ත ප්‍රරූප (Data Types) –

ක්‍රමලේඛයක් තුල භාවිතා කල හැකි දත්ත ඒවායේ භාවිතය අනුව ඛාණ්ඩ ගත කල හැක. මේවා දත්ත ප්‍රරූප ලෙස හැඳින් විය හැක.

PHP ක්‍රමලේඛනයේදී භාවිත වන දත්ත ප්‍රරූප ආකාර 4 කි.

1. Integer - පූර්ණ සංඛ්‍යා.
2. Float - ඉපිලුම් ස්ථාන අංක.
3. String - අනුලකුණු වැල.
4. Boolean - බූලියානු අගයන්.

1. Integer - පූර්ණ සංඛ්‍යා. -

   දශමස්ථාන නොමැති ධන හෝ ඍණ පූර්ණ සංඛ්‍යා.

උදා:- 123, -123, 4B(ෂඩ් දශමය), 1011001(ද්විමය).

2. Float - ඉපිලුම් ස්ථාන අංක. -

   දශමය හෝ බලයන් ආකාරයෙන් දැක්විය හැක.

   උදා:- 2.546, 5E-10

3. String - අනුලකුණු වැල. -

තනි හෝ ද්විත්ව Inverted commas තුල යෙදීමෙන් දත්තයක් string බවට පත් වේ.

   උදා:- “Hello World”, “123”

4. Boolean - බූලියානු අගයන්. -

True සහ False ලෙස එකිනෙකට ප්‍රතිවිරුද්ධ තත්ත්ව දෙකක් මෙයින් අදහස් කෙරේ. (ද්විමය අංකනයෙදී 0 සහ 1 වැනි).

මෙම පද Case sensitive නොවේ.

උදා:- True, true, TRUE, False, false, FALSE.

5.7 PHP මෙහෙයුම් (Operators) -

1. අංක ගණිත මෙහෙයුම් -

Operator	Name	Example	Result
+	Addition	$x + $y	$x සහ $yවල එකතුව
-	Subtraction	$x - $y	$x සහ $yවල අන්තරය
*	Multiplication	$x * $y	$x සහ $yවල ගුණිතය
/	Division	$x / $y	$x $yවලින් බෙදීම
%	Modulus	$x % $y	$x $yවලින් බෙදා ශේෂය
**	Exponentiation	$x ** $y	$x නැමති සංඛ්‍යාවේ$yවෙනි බලය

තාර්කික මෙහෙයුම් -

Operator	Name	Example	Result
and	And	$x and $y	$x සහ $yසත්‍යවේ නම් ප්‍රතිඵලය සත්‍යවේ
or	Or	$x or $y	$x හෝ $yසත්‍යවේ නම් ප්‍රතිඵලය සත්‍යවේ
xor	Xor	$x xor $y	$x හෝ $yදෙකෙන් එකක් පමණක් සත්‍යවේ නම් ප්‍රතිඵලය සත්‍යවේ
&&	And	$x && $y	$x සහ $yසත්‍යවේ නම් ප්‍රතිඵලය සත්‍යවේ
||	Or	$x || $y	$x හෝ $yසත්‍යවේ නම් ප්‍රතිඵලය සත්‍යවේ
!	Not	!$x	$x අසත්‍යවේ නම් ප්‍රතිඵලය සත්‍යවේ

පාලන ව්‍යුහ (Control Structures) –

යම් කිසි කොන්දේසියක් තෘප්ත වීම හා නොවීම මත එකිනෙකට වෙනස් තත්ත්ව දෙකක් ලබා ගැනීමට / එකිනෙකට වෙනස් කේත ඛණ්ඩ දෙකක් ක්‍රියාත්මක කිරීමට පාලන ව්‍යුහ භාවිත කරයි.

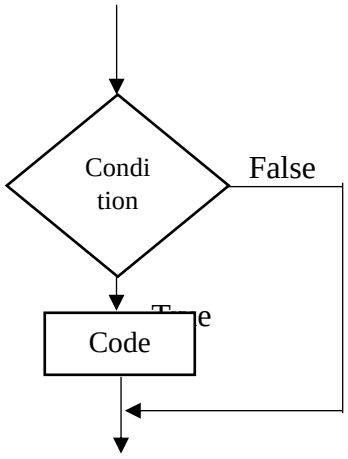
PHP හි භාවිත වන පාලන ව්‍යුහ 4 කි.

1. if
2. if...else
3. if...elseif...else
4. switch

1. if ව්‍යුහය.

සරලතම if ප්‍රකාශය වේ. යම් කිසි එක කොන්දේසියක තෘප්ත දිශාව පමණක් සලකයි.

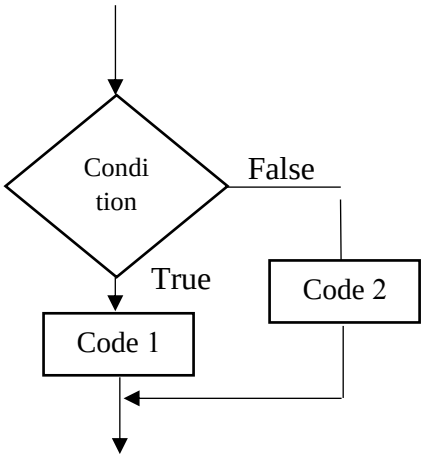
Syntax
If(condition)
{
Code;
}



2. if...else ව්‍යුහය.

යම් කිසි එක කොන්දේසියක තෘප්ත දිශාව සහ තෘප්ත නොවන දිශාව සලකා එකිනෙකට වෙනස් කේත ඛණ්ඩ 2 ක් ක්‍රියාත්මක කිරීමට යොදාගැනේ.

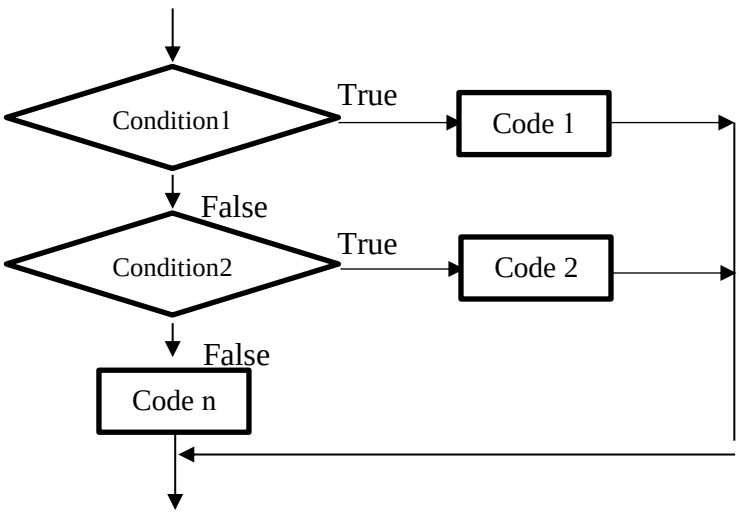
```
Syntax
If  (condition)
{
Code  1;
}
else
{
Code  2;
}
```



3. if...elseif...else ව්‍යුහය.

කොන්දේසි කිහිපයක තෘප්ත දිශාවන් සහ තෘප්ත නොවන දිශාව සලකා එකිනෙකට වෙනස් කේත ඛණ්ඩ ගණනාවක් ක් ක්‍රියාත්මක කිරීමට යොදාගැනේ.

```
Syntax
If  (condition)
{
Code  1;
}
else  if
{
Code  2;
}.....
else
{
Code  n;
}
```



4. Switch ව්‍යුහය.

5.8 PHP පුනර්කරණ (PHP Loops) –

ක්‍රමලේඛනයක යම් කිසි කොන්දේසියක් තෘප්ත වී පවතින තාක් හෝ නොවී තිබෙන තාක් එකම කේත ඛණ්ඩය නැවත නැවතත් ක්‍රියාත්මක කිරීමට අවශ්‍ය වූ විට පුනර්කරන භාවිත වේ.

1. While –

Syntax -
While(condition is true)
{
Code to be executed;
}
	උදා:-
	<?php
	$x=12;
	while($x<=12)
	{
	echo $x;
	$x++;
	}
	?>

2. Do – While –

Syntax -
do
{
Code to be executed;
}
while(condition is true);
	උදා:-
	<?php
	$x=12;
	do
	{
	echo $x;
	$x++;
	}
	while($x<=12)
	?>

3. For –

Syntax -
for(initialization;check;post execution)
{
Code to be executed;
}
	උදා:-
	<?php
	for($x=0;$x<=12;$x++)
	{
	echo $x;
	}
	?>

5.9 PHP ශ්‍රිත (PHP Functions) –

කිසියම් ක්‍රමලේඛ ඛණ්ඩයක් විවිධ පරාමිති වලට අනුව ධාවනය කිරීම සඳහා ශ්‍රිත භාවිත වේ. PHP තුල අවශ්‍ය පරිදි ශ්‍රිත ගොඩනගාගත හැකි අතර php තුල ශ්‍රිත විශාල ප්‍රමාණයක් ඇතුලත් කර ඇත.

උදා:- fopen(), fread(), date()

ශ්‍රිතයක කාරක නීතිය -

Syntax -
function function_name()
{
Code to be executed;
}
	උදා:-
	<?php
	function errormsg()
	{
	echo “Unexpected error”;
	}
	?>

ඉහත පරිදි ලියන ලද ශ්‍රිතයක් කේතයේ වෙනත් ස්ථානයක call කරනුයේ මෙසේය

<?php

errormsg();

?>

5.10 PHP දත්ත ව්‍යුහ (Data Structures) –

අරාවන්(Arrays) -

ක්‍රමලේඛනයක විචල්‍යන්ගේ බහු අගයන් ගබඩා කිරීමට අරාවන් භාවිත කරයි.

උදා-

<?php

$team = array(“Thanuja”,”Anjana”,”Sunali”);

var_dump($team);

?>

foreach භාවිතයෙන් අරාවක් තුල ඇති වටිනාකම් ප්‍රතිදානය කිරීම.

<?php

$colors=array(“red”,” blue”,”Green”);

foreach($colors as $value);

?>

5.11 PHP ගොනු පරිහරණය (File Handling) –

සේවාදායක පරිගණක පරිසරයක ඇතුලත් ගොනුවක ඇති දත්ත කියවීම, ලිවීම සඳහා පහත ක්‍රියාවලි අනුගමනය කෙරේ.

1. ගොනුව විවෘත කිරීම.
2. අඩංගු දත්ත කියවීම.
3. දත්ත ලිවීම.
4. ගොනුව වැසීම.

1. ගොනුව විවෘත කිරීම.

- මේ සඳහා fopen විධානය භාවිත කරයි.
- පහත පරිදි යොදනු ලැබේ.

```
$privatefile=fopen("filename","Mode");
```

මෙහිදී mode යන්නෙන් ගොනුවේ මාදිලිය අදහස් කෙරේ. එනම් ගොනුව විවෘත කරන්නේ කුමන අරමුණක් සඳහා ද යන්න දැක්වේ.

මාදිලිය	විස්තරය	ගොනු දර්ශකය
r	ගොනුව කියවීමට පමණක් විවෘත කිරීම.	ගොනුවේ මුල සිට ඇරඹේ.
w	ගොනුවක් ලිවීම සඳහා පමණක් විවෘත කිරීම මෙහිදී සිදුවේ. ගොනුවක අන්තර්ගතය නොමැතිනම් අලුත් ගොනුවක් නිර්මාණය වේ.	ගොනුවේ මුල සිට ඇරඹේ.
a	ගොනුවක් ලිවීම සඳහා පමණක් විවෘත කිරීම මෙහිදී සිදුවේ. ගොනුවක අන්තර්ගතය නොමැතිනම් අලුත් ගොනුවක් නිර්මාණය වේ.	ගොනුවේ අවසානයේ සිට ඇරඹේ.

2. අඩංගු දත්ත කියවීම.

මේ සඳහා fgets සහ fputs විධාන භාවිත වේ.

1. fgets(file variable) - ගොනුවක එක පේළියක් කියවයි.
2. fputs(file variable,string) - ගොනුවකට පාඨ එක්කරයි.

3. දත්ත ලිවීම.

උදා:-

පහත දැක්වෙන කේත ඛණ්ඩය මගින් "This is a new line" යන පාඨ කොටස abc.txt ගොනුවට ලියනු ලබයි.

abc.txt ලෙස ගොනුවක් නොමැති වේලාවකදී fopen විධානය මගින් abc.txt ලෙස නව ගොනුවක් නිර්මාණය වේ.

```
<?php
        $myfile = fopen("abc.txt","w");
        fputs($myfile,"This is a new line");
        fclose($myfile);
?>
```

පහත ක්‍රමලේඛය මගින් abc.txt හි පළමු පේළිය කියවා ආදානය html කේතය තුලට ප්‍රතිදානය කරයි.

```
<html>
	<body>
		<?php
		$myfile = fopen(“abc.txt”,”r”);
		$data = fgets($myfile);
		echo $data;
		fclose($myfile);
		?>
	</body>
</html>
```

ගොනුවේ අන්තර්ගතය වරකට එක පේළියක් බැගින් කියවා html තුලට ඇතුලත් කිරීමේදී feof() ශ්‍රිතය භාවිත කරයි. එමගින් ගොනුව අවසානයටම කියවූ බව තහවුරු කරගත හැක.

```
<html>
	<body>
		<?php
		$myfile = fopen(“abc.txt”,”r”);
		while(!feof($myfile)
		{
		$data = fgets($myfile);
		echo $data,”<br>”;
		}
		fclose($myfile);
		?>
	</body>
</html>
```

4. **ගොනුව වැසීම**.

fclose යන විධානය භාවිත කරයි. – fclose($file variable)

උදා:-

```
fclose($myfile);
```

5.12 PHP මගින් දත්ත සමුදායක් කළමනාකරණය කිරීම -

1. **HTML පෝරමයකින් දත්ත සොයා ලබාගැනීම (Retrieving data from HTML forms)-**

$_GET සහ $_POST යන විධාන දත්ත එකතු කිරීම සඳහා භාවිත කරයි.

```
<html>
	<body>
		<form action=”welcome.php” method=”post”>
		Name:<input type=”text” name=”name”><br>
		Email:<input type=”text” name=”email”><br>
		<input type=”submit”>
		</form>
	</body>
</html>
```

ඉහත කේතය ක්‍රියාත්මක කිරීමෙන් පසු අදාළ ක්ෂේත්‍ර වලට සම්පූර්ණ කල තොරතුරු welcome.php වෙත යැවීම සඳහා http post method යොදා ගැනේ. echo විධානය සමග විචල්‍යන් ඇතුලත් කිරීමෙන් ඇතුලත් කල දත්ත බලාගත හැක.

```
<html>
    <body>
        Hi<?php echo $post[“name”];?><br>
        Email address is: <?php echo $post[“email”];?>
    </body>
</html>
```

ඉහත ක්‍රියාවලිය සඳහා http post method වෙනුවට http get method ද යොදාගත හැක.

2. සේවා දායක පරිගණකය හ සම්බන්ධ වීම.

Procedural

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Create connection
$conn = mysqli_connect($servername, $username,
$password);

// Check connection
if (!$conn) {
   die("Connection failed: " .
mysqli_connect_error());
}
echo "Connected successfully";
?>
```

මෙම කේත ඛණ්ඩය ක්‍රියාත්මක වූ වහාම සේවාදායක පරිගණකය සමග ඇති සම්බන්ධතාවය විසන්ධි වී යයි. ඊට පෙර එය සිඳලීමට

$conn->close(); යොදයි.

3. දත්ත සමුදායක් නිර්මාණය කිරීම -

Procedural

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
```

```
// Create connection
$conn = mysqli_connect($servername, $username,
$password);
// Check connection
if (!$conn) {
   die("Connection failed: " .
mysqli_connect_error());
}

// Create database
$sql = "CREATE DATABASE myDB";
if (mysqli_query($conn, $sql)) {
   echo "Database created successfully";
} else {
   echo "Error creating database: " .
mysqli_error($conn);
}

mysqli_close($conn);
?>
```

4. **දත්ත සමුදාය වගු සෑදීම -**

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
   die("Connection failed: " . $conn->connect_error);
}

// sql to create table
$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP
)";

if ($conn->query($sql) === TRUE) {
   echo "Table MyGuests created successfully";
```

```
} else {
   echo "Error creating table: " . $conn->error;
}

$conn->close();
?>
```

5. දත්ත සමුදායට දත්ත ඇතුලත් කිරීම -

Procedural

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = mysqli_connect($servername, $username,
$password, $dbname);
// Check connection
if (!$conn) {
   die("Connection failed: " .
mysqli_connect_error());
}

$sql = "INSERT INTO MyGuests (firstname,
lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";

if (mysqli_query($conn, $sql)) {
   echo "New record created successfully";
} else {
   echo "Error: " . $sql . "<br>" .
mysqli_error($conn);
}

mysqli_close($conn);
?>
```

6. දත්ත සමුදායේ දත්ත නැවත ලබා ගැනීම -

Procedural

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
   die("Connection failed: " . mysqli_connect_error());
}

$sql = "SELECT id, firstname, lastname FROM MyGuests";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {
   // output data of each row
   while($row = mysqli_fetch_assoc($result)) {
      echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " .
$row["lastname"]. "<br>";
   }
} else {
   echo "0 results";
}

mysqli_close($conn);
?>
```

නිපුණතාව 11 - සාර්ව ද්‍රව්‍ය අන්තර්ජාලය (Internet of Things - IOT)

විවිධ සංවේදක, මෘදුකාංග හා ඉලෙක්ට්‍රොනික උපාංග දත්ත සන්නිවේදනය කළ හැකි ආකාරයට සකස් කොට ජාලගත කළ විට සාර්ව ද්‍රව්‍ය අන්තර්ජාලය ලෙස හැඳින්වෙන සුහුරු පරිසරය නිර්මාණය වෙයි.

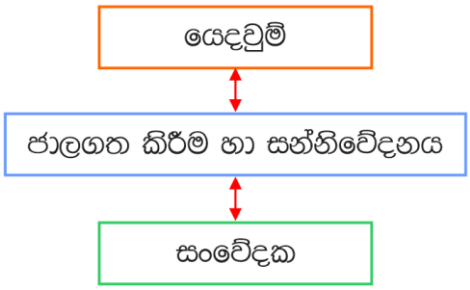
මෙහි උපාංග ජාලගත කිරීම සඳහා සාම්ප්‍රදායික ජාලගත ක්‍රම හෝ මීටම සුවිශේෂී ක්‍රම භාවිතා කරන්නට පුළුවන. එසේ ඇති සුහුරු පරිසරය සුහුරු දුරකථන හෝ පරිගණක හරහා හෝ දුරස්ථව අධීක්ෂණය කිරීම හෝ පාලනය කිරීමද කළ හැකි වුවත්, එසේ අධීක්ෂණයකින් හෝ පාලනයකින් තොරව ක්‍රියාත්මක වෙන ආකාරයට සුහුරු පද්ධති නිර්මාණය කිරීමට පුළුවන. බොහෝ සුහුරු පද්ධති දවසේ පැය 24 තුළම ක්‍රියාත්මක වෙයි. මේ නිසා මේවා පහසුවෙන් අක්‍රීය නොවන පද්ධති ලෙසද හඳුන්වා දෙන්නට පුළුවන.

සාර්ව ද්‍රව්‍ය අන්තර්ජාලය උපාංගයක් ලෙස උෂ්ණත්ව සංවේදකයක් වැනි සරල උපකරණයක සිට මෝටර් රථයක්, නිවසක්, ගොඩනැගිල්ලක්, නගරයක් වුවද අයත් විය හැකිය.

සාර්ව ද්‍රව්‍ය අන්තර්ජාලය සඳහා තවම එක් නිර්වචනයක් නැත. එබැවින් විවිධ ආයතන, විවිධ ප්‍රමිතීන් සහ හැඳින්වීම් කීපයක්ම මේ සම්බන්ධව ඉදිරිපත්ව තිබේ. ඒවායින් ජනප්‍රිය හැඳින්වීම් කීපයක් මෙසේය.

1. විදුලි සහ ඉලෙක්ට්‍රොනික ඉංජිනේරු ආයතනය - IEEE

IEEE (Institute of Electrical and Electronic Engineers) හී අර්ථ දැක්වීමට අනුව උපාංග සියල්ල ජාලගතව තිබිය යුතුය. උපාංගවලට අන්තර්ජාල සම්බන්ධතාවය තිබිය යුතුය. එමෙන්ම සංවේදක අන්තර්ගත විය යුතුය. මෙම නිර්වචනයට අනුව IOT සඳහා අන්තර්ජාලය අත්‍යවශ්‍ය බව කියැවුනද ඇතැම් නිර්වචනවල එසේ අන්තර්ජාලයේ අත්‍යවශ්‍යතාවයක් දක්වන්නේ නැත. ඊට හේතුව IOT යන යෙදුමේ Internet යන්නෙන් සාමාන්‍ය අන්තර්ජාලය නොව එකිනෙක සබැඳුනු ජාලයක් යන්න අර්ථ දක්වෙන නිසාය. ඉහත සටහන අනුව මෙම සාර්ව ද්‍රව්‍ය අන්තර්ජාල උපාංග තිබිය යුතුය.



2. යුරෝපීය විදුලි සංදේශ ප්‍රමිති ආයතනය - ETSI

ETSI (European Telecommunications Standards Institute) විසින් IOT යන්න හඳුන්වන්නේ මිනිසුන්ගේ සෘජු මැදිහත්වීමකින් තොරව උපාංග දෙකක් හෝ වැඩි ගණනක් අතර සන්නිවේදනයක් සිදුවීම ලෙසය. මෙහිදී සන්නිවේදනය හා තීරණ ගැනීම ස්වයංක්‍රියව වෙයි. මෙම නිර්වචනයේ සාමාන්‍ය අන්තර්ජාලයේ අවශ්‍යතාවයක් ගැන දැක්වෙන්නේ නැත.

3. අන්තර්ජාල ඉංජිනේරු කාර්යසාධක බලකාය - IETF

IETF (Internet Engineering Task Force) මඟින් IOT යන්න හඳුන්වන්නේ අප වටා වූ විවිධ විද්‍යුත්, ඉලෙක්ට්‍රොනික හෝ ඉලෙක්ට්‍රොනික නොවන උපාංග, ඒවා අතර අඛණ්ඩ සන්නිවේදනයක් සහ ඒවා ආශ්‍රිත සේවාවන් සැපයෙන ආකාරයට සම්බන්ධ කර ඇති පද්ධතියක් ලෙයසි. මෙම නිර්වචනයේ සාමාන්‍ය අන්තර්ජාලයේ අවශ්‍යතාවයක් ගැන දැක්වෙන්නේ නැත.

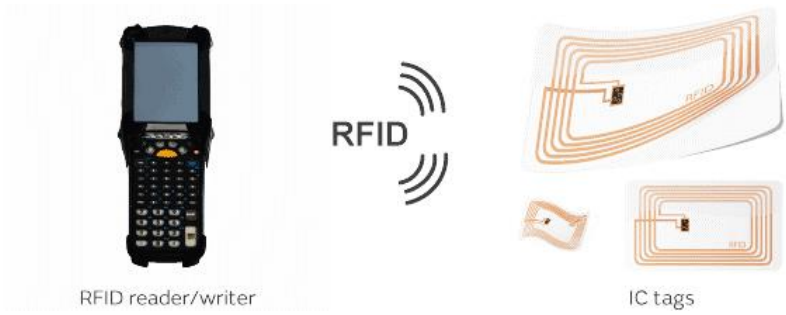
4. අන්තර්ජාතික දුරස්ථ සන්නිවේදන සංගමය - ITU

ITU (International Telecommunication Union) යනු එක්සත් ජාතීන්ගේ සංවිධානයේ කොටසකි. මෙම ආයතනය විසින් 2005 වසරේදී IOT පිළිබඳව ඉදිරිපත් කර ඇති වාර්තාවලට අනුව IOT යනු සාර්වවර්ති ජාලයකි. (ubiquitous network) සාර්වවර්ති ජාලයක් යනු ඕනෑම මොහොතක, ඕනෑම කෙනෙකුට, ඕනෑම තැනක සිට ජංගම දුරකථන වැනි ඕනෑම උපාංගයක් හරහා සම්බන්ධ විය හැකි ජාලයකි. ඒ අනුව මෙම නිර්වචනයේදී සාමාන්‍ය අන්තර්ජාලයේ අවශ්‍යතාවය ගැන අදහසක් ඉදිරිපත් කර ඇත.

මේවාට අමතරව IOT වල වර්ධනය සඳහා ප්‍රධාන වශයෙන් සාධක වී ඇති තාක්ෂණ කීපයක් ITU ආයතනය විසින් හඳුන්වාදී ඇත. ඒවා මෙස්ය.

i. වස්තු හඳුනා ගැනීම

මේ සඳහා ප්‍රධාන වන්නේ RFID තාක්ෂණයයි. මෙහිදී ගුවන්විදුලි සංඛ්‍යාත හඳුනා ගැනීම කරනු ලබයි. ඒ අනුව යම් වස්තුවක හැඳුනුම් සංකේත අලවා එය හඳුනා ගැනීම මෙහි ක්‍රමයයි.



ii. සංවේදක

විවිධ වර්ගවල සංවේදක සාර්ව ද්‍රව්‍ය අන්තර්ජාලය සඳහා යොදා ගන්නා බව දනිමු. උෂ්ණත්වය, ගොවිපලක පසෙහි තෙතමනය මැනීම, ගෑස් කාන්දුවක් හඳුනාගැනීමේ සංවේදක, ආලෝකය හඳුනාගැනීමේ සංවේදක ආදිය මීට උදාහරණයි.

iii. සුහුරු ද්‍රව්‍ය (Smart materials)

සුහුරු ද්‍රව්‍ය, බුද්ධිමත් ද්‍රව්‍යය හා ප්‍රතිචාරාත්මක ද්‍රව්‍යය යන නම් වලින්ද හැඳින්වෙයි. සංවේදක හරහා ලැබෙන බාහිර උත්තේජනවලට ප්‍රතිචාර දක්වීම සඳහා තම ගුණාංග එකක් හෝ කීපයක කැපී පෙනෙන වෙනස්කම් කරන ආකාරයට නිර්මාණය කර ඇති උපාංග මෙනමින් හැඳින්වෙයි. මෙම සුහුරු ද්‍රව්‍ය ආකාර තුනකි. ඒවා නම්,

- නිශ්ක්‍රීය සුහුරු ද්‍රව්‍ය

යම්කිසි උත්තේජනයක් යෙදූ විට සංඥාව විශ්ලේෂණය නොකර ඒකාකාර ලෙස සහ කෙලින්ම ප්‍රතිචාර දක්වන ද්‍රව්‍ය මෙලෙස හැඳින්වෙයි. මීට උදාහරණයක් ලෙස විකෘති වූ දත්වල හැඩය නිවැරදි කිරීමට යොදා ගන්නා නිකල්-ටයිටේනියම් කම්බි දැක්විය හැකිය. එම කම්බියේ ආරම්භක හැඩය දත් ඇන්දක නිවැරදි හැඩය ලෙස සකසයි. දැන් එය විකෘති වූ දත් ඇන්දට සවි කරයි. එවිට දත් ඇන්ද නිවැරදි හැඩයට සකස් වෙන තුරු කම්බිය මඟින් ඒකාකාර ප්‍රත්‍යාබලයක් දත් ඇන්දට යොදනු ලබයි. අවසානයේ කම්බිය නැවත මුල් හැඩයට එයි. යම් වස්තුවක් තද කළ විට හැඩය වෙනස් වීමත්, එම තද කිරීම අත හැරිය විට නැවත මුල් හැඩයට පත්වීමත් මීට උදාහරණ ලෙස දැක්වීමට පුළුවන.

- සක්‍රීය සුහුරු ද්‍රව්‍ය

සංඥාවක් ලබාගෙන, දුරස්ථ පාලකයක පාළනය යටතේ සුදුසු පරිදි ප්‍රතිචාර දක්වන ද්‍රව්‍ය මෙලෙස හඳුන්වයි. යම්කිසි පීඩනයක් යෙදූ විට කුඩා විභව අන්තරයක් නිපදවන සහ විද්‍යුත් ක්ෂේත්‍රයක් යෙදූ විට හැඩය වෙනස් කර ගන්නා පීඩ විද්‍යුත් ද්‍රව්‍ය මීට උදාහරණයි. පීඩ විද්‍යුත් ද්‍රව්‍යට උදාහරණ ලෙස විද්‍යුත් ඔරලෝසු, පියන විවෘත කළ විට වාදනයක් ඇසෙන සුභ පැතුම්පත්, මයික්‍රෝෆෝන ආදිය දක්වන්නට පුළුවන.

- ස්වතන්ත්‍ර සුහුරු ද්‍රව්‍ය

සංවේදක, පාලක සහ ප්‍රතික්‍රියක යන සියල්ල සහිත ද්‍රව්‍ය මෙනමින් හැඳින්වෙයි.

IV සූක්ෂමකරණය

විවිධ උපාංග වලට සංවේදක සහ පරිපථ සම්බන්ධ කිරීමේ දී එම සංවේදක සහ අනෙකුත් පරිපථ ඉතා කුඩා විය යුතුය. මෙවැනි කුඩා උපාංග තැනීමට නැනෝ තාක්ෂණය භාවිතා කරයි.

සුහුරු උපාංගවල දියුණුව IOT සඳහා බලපෑම් කර ඇත. මෙහිදී සුහුරු උපාංග ලෙස හැඳින්වෙන්නේ බිලූටූත්, වයිෆයි වැනි රැහැන් රහිත සම්මුතීන් හරහා වෙනත් උපාංගවලට හෝ ජාලවලට සම්බන්ධ විය හැකි අන්තර්ක්‍රියාකාරීත්වයක් සහිත ස්වාධීන ඉලෙක්ට්‍රොනික් උපාංගයි.

IOT හි අවශ්‍යතාවයන්

පරිගණක නොවන වෙනත් උපාංග අන්තර්ජාලය හා සම්බන්ධ කිරීමට උවමනා වන්නේ ඒවා අන්තර්ජාලය හරහා පාළනය කිරීමට බව දනිමු. මීට උදාහරණයක් ලෙස නිවසේ විදුලි බල්බයක් අන්තර්ජාලය හරහා පාළනය කිරීම, නිවසේ ඇති විදුලි පංකාවක් දුරස්ථ ස්ථානයක සිට අක්‍රීය කිරීම ආදිය දැක්වීමට පුළුවන. මේ ආකාරයේ විවිධ යෙදුම් ගණනාවක් වර්තමානය වෙන විට භාවිතා වෙයි. එම යෙදුම් පහත ආකාරවලින් වර්ග කරන්නට පුළුවන.

1) ගොඩනැඟිලි සහ නිවෙස් සඳහා වන යෙදුම්.

මේවාට උදාහරණ ලෙස සුහුරු බල්බ්, සුහුරු ප්ලග්, සුහුරු අගුළු ආදිය හඳුන්වා දෙන්නට පුළුවන. මේවාට අමතරව පහත දැක්වෙන උපාංගද මෙම යෙදුම්වලට ගැනෙයි.

Air Quality Egg (AQE)



මෙමඟින් ගෘහස්ථ පරිසරය තුළ පවතින කාබන්ඩයොක්සයිඩ් ප්‍රමාණය, නයිට්‍රජන් ඩයොක්සයිඩ් ප්‍රමාණය වැනි දේ හඳුනාගන්නට පුළුවන. මේ හරහා ලබාගන්නා දත්ත වයිෆයි හරහා සේවාදායක පරිගණකයකට යවයි. ඒ හරහා අවශ්‍ය විස්ලේෂණයන් කළ හැකිය.

Amazon Echo

මෙය යොදා ගෙන කටහඬ මඟින් උපාංග පාලනය කිරීමට පුළුවන. වෙනත් බාහිර ශබ්ද සමඟ වුවද මෙම උපාංගයට මුවින් පිටවෙන විධාන වෙන් කර හඳුනාගෙන ඒවාට අදාළව උපාංග පාලනය කිරීම කළ හැකිය. මෙහි සංවේදක හා මයික්‍රෆෝන ගණනාවක් අන්තර්ගත කර ඇත. උපාංග ක්‍රියාත්මක කිරීමට අමතරව සංගීතයට සවන් දීම, පොත් කියවා ගැනීම වැනි දේද මේ හරහා කළ හැකිය.

2) වෛද්‍ය හා සෞඛ්‍යය ආරක්ෂක පද්ධතිවල යෙදවුම්

හදිසි දැනුම් දීම්, දුරස්ථ සෞඛ්‍ය නිරීක්ෂණ උපාංග සක්‍රීය කිරීම වැනි දේ සඳහා IOT උපාංග බහුලව භාවිතා වේ. මේ අතර හෘද ස්පන්ධනය, රුධිර පීඩනය, ශරීර උෂ්ණත්වය වැනි තත්වයන් හඳුනාගැනීමට ඒවා පාලනය කිරීමට අවශ්‍ය උපදෙස් සැපයීමට IOT උපාංග භාවිතා වෙයි. මෙවැනි උපාංග කීපයක් මෙසේය.

Medication Dispensing Service



මෙම උපාංගය මඟින් නියමිත බෙහෙත් මාත්‍රාව අවශ්‍ය කාලයට නිකුත් කරනු ලබයි. ඒ සඳහා මුලින්ම මෙම උපාංගය වෙත අවශ්‍ය බෙහෙත් ඇතුළු කර තැබිය යුතුය. එවිට නියමිත වේලාවට බෙහෙත් ගැනීම මතක් කරවන සංඥා නිකුත් කර බෙහෙත් වර්ගද පිටතට ලබාදීම මෙමඟින් කරයි.

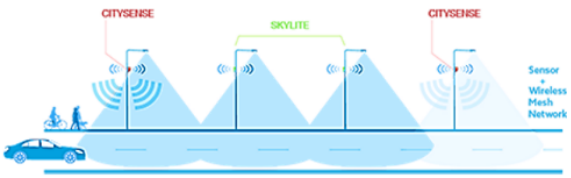
Jawbone UP2



මෙය අත් පළඳනාවකි. මේ හරහා ශරීර යෝග්‍යතා මිනුම් නිරීක්ෂණය කිරීම, ආහාර ගැනීමේ වාර්තා පවත්වා ගෙන යෑම, ඇවිදීම හා ව්‍යායාම කිරීමට අදාළව වාර්තා පවත්වා ගෙන යෑම වැනි දේ කරනු ලබයි.

3) සුහුරු නගර සඳහා වන යෙදවුම්

කසල කළමනාකරණය සහ ප්‍රතිචක්‍රීකරණය සඳහා නිපද වූ සුහුරු උපාංග තිබේ. bigbelly smart waste and recycling system යනු එවැන්නකි. මේවා හරහා කසල බඳුන් පිරීම් ආදිය විස්තර සේවකයින්ට පහසුවෙන් දැනගත හැක. මෙවැනි තවත් උපාංග මෙසේය.



- වීථි සඳහා සුහුරු ආලෝකකරණය

අඳුර වැටෙන විට දැල්වෙන පහනක සිට, පදිකයින් සහ මෝටර් රථ හඳුනාගෙන ඉබේ ක්‍රියාත්මක වන ආලෝක පද්ධති දක්වා විවිධාකාර ආලෝක පද්ධති නිර්මාණය කිරීම කළ හැකිය.

4) කෘෂිකර්මාන්ත සඳහා වන යෙදවුම්

ආර්ද්‍රතාවය, වායුගෝලීය උෂ්ණත්වය සහ පාංශු ගුණාත්මකභාවය වැනි කරුණු අධීක්ෂණය කළ හැකි සුහුරු උපාංග ඇත. එමෙන්ම වගාවට ජල ය සැපයීම, අස්වනු නෙලීම, පැල සිටුවීම, වල්පැල ඉවත් කිරීම වැනි බොහෝ දේ කළ හැකි විවිධ උපාංග ඇත.

ඉහත විස්තර කළ ක්ෂේත්‍රවලට අමතරව ප්‍රවාහන ක්ෂේත්‍රය, කර්මාන්ත ක්ෂේත්‍රය, බලශක්ති කළමනාකරණය වැනි සෑම ක්ෂේත්‍රයකම පාහේ සුහුරු උපාංග භාවිතය වර්තමානය වන විට ප්‍රචලිත වෙමින් පවතී.

ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධති හැඳින්වීම

ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධති Microprocessor Development Systems (MDS) යනු කුඩා ප්‍රමාණයේ මයික්‍රෝ පරිගණකයකි. ක්ෂුද්‍ර සකසනයක් පාදක කර ගෙන නිර්මාණය කරන පරිපථ වර්තමානයේ බහුලව භාවිතා වෙයි. ගුවන් යානා, රොබෝ යන්ත්‍ර, වාහන, කර්මාන්ත ශාලාවල තිබෙන යන්ත්‍ර වැනි සංකීර්ණ යන්ත්‍ර ස්වයංක්‍රීයව ක්‍රියාත්මක වෙන්නේ ඒවා සමඟ ඊට ආවේනික MDS සවි කර තිබෙන නිසාය.

එමෙන්ම නිවසේ බල්බ අන්තර්ජාලය හරහා ක්‍රියාත්මක කිරීම, අනතුරු හැඟවීමේ සංඥා පද්ධති ක්‍රියාත්මක කරවීම, සොර සතුරන්ගෙන් නිවෙස් ආරක්ෂා කරගත හැකි පද්ධති නිර්මාණය කිරීම වැනි බොහෝ දේ සඳහා දැන් MDS භාවිතා කරයි.

බැලූ බැල්මට MDS එකක් පෙනෙන්නේ මුද්‍රිත පරිපථයක් ආකාරයෙනි. එවැනි පරිපථයක විවිධ අංග අන්තර්ගත වේ. එසේ ඇති අංග MDS වර්ගයට අනුව වෙනස් වෙයි. නමුත් පොදුවේ ගත්කල සෑම MDS එකකම පාහේ අන්තර්ගත අංග කීපයක් තිබේ. ඒවා මෙසේය.

- **ක්ෂුද්‍ර සකසනය (microprocessor)** - සාමාන්‍ය පරිගණකයක මෙන් සියලු සැකසුම් කටයුතු කරන්නේ මෙමඟිනි. ඇතැම් ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධතිවල ක්ෂුද්‍ර සකසනය වෙනුවට මයික්‍රෝකන්ට්‍රෝලර් (Microcontroller) හෙවත් ක්ෂුද්‍ර පාලකයක් අන්තර්ගත වේ. ක්ෂුද්‍ර පාලකයක් යනු සාමාන්‍ය ක්ෂුද්‍ර සකසනයක මෙන් පාලන ඒකකයක් (Control Unit), ගණිත හා තාර්කික ඒකකයක් (arithmetic and logic unit) යන අංගවලට අමතරව රෑම් (RAM) මතකයක්, රොම් (ROM) මතකයක් සහ සැනෙලි මතකයක් හෙවත් ෆ්ලැෂ් (flash) මතකයක් ඇති උපාංගයකි. මේ නිසා ක්ෂුද්‍ර සකසනයට මෙන් නොව ක්ෂුද්‍ර පාලකයට පරිගණකයක් සේ තනිව ක්‍රියාත්මක විය හැකිය. නමුත් බැලූ බැල්මට ක්ෂුද්‍ර පාලකයක්ද පෙනෙන්නේ සාමාන්‍ය චිප් එකක් ලෙසටය.

- **ප්‍රතිසම ආදාන (Analog input)** - ආලෝකය, සුළඟ, උෂ්ණත්වය, හඬ වැනි ඇනලොග් හෙවත් ප්‍රතිසම සංඥා ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධති වෙත ඇතුළු කර ගන්නේ මෙම ආදාන හරහාය. MDS උපාංගයක පිටත දාරයට ආසන්නව පින් හෙවත් තුඩු වශයෙන් හෝ සම්බන්ධක වශයෙන් මේවා දැකගත හැක.

- **අංකිත ආදාන (Digital input)** - විදුලිය ඇත, නැත යන්න මත පාදක වූ අංකිත සංඥා ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධති වෙත ඇතුළු කර ගන්නේ මෙම ආදාන හරහාය. මේවාද MDS උපාංගයක ප්‍රතිසම ආදාන සමඟ පිටත දාරයට ආසන්නව පින් හෙවත් තුඩු වශයෙන් හෝ සම්බන්ධක වශයෙන් දැකගත හැක.
- **අංකිත ප්‍රතිදාන (Digital Output)** - බල්බයක් දැල්වීමට, මෝටරයක් ක්‍රියාත්මක කිරීමට, යම් සංවේදකයක් සක්‍රීය කිරීමට වැනි කාර්යයක් සඳහා අවශ්‍ය අංකිත සංඥා ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධති මඟින් පිටතට ලබාදෙන්නේ මෙම ප්‍රතිදාන හරහාය. මේවාද MDS උපාංගයක ඉහත විස්තර

කළ ආදාන සමඟ පිටත දාරයට ආසන්නව පින් හෙවත් තුඩු වශයෙන් හෝ සම්බන්ධක වශයෙන් දැකගත හැක.

- **Tx සහ Rx පින්** - මේවා භාවිතා කරන්නේ ශ්‍රේණිගත (Serial) සන්නිවේදනය සඳහාය. යූඑස්බී ද ශ්‍රේණිගත සන්නිවේදනයකි. මේ නිසා බොහෝ විට MDS උපාංගය යූඑස්බී කෙවෙනිය හරහා පරිගණකයකට සම්බන්ධ කර ඇති විටදී Tx සහ Rx පින් භාවිතා කරන්නේ නැත.

- **විදුලි සැපයුම (Power supply)** - MDS එක වෙත බාහිරින් විදුලි බලය ලබාදෙන්නේ මේ හරහාය.

- **යූඑස්බී (USB)** - පරිගණකයකින් MDS එක වෙත විවිධ වැඩසටහන් ඇතුළු කරන්නේ මේ හරහාය. යූඑස්බී පෝට් එක හරහා උපාංගය පරිගණකයට සම්බන්ධ කර තිබියදී ඉහත විස්තර කළ විදුලි සැපයුම හරහා විදුලිය දෙන්නට අවශ්‍ය නැත.

- **ප්‍රත්‍යාරම්භක ස්විචය (Reset Switch)** - සාමාන්‍යයෙන් MDS උපාංගයකට විදුලිය සැපයූ සැනින් එහි ඇති මෘදුකාංග ක්‍රියාත්මක වෙයි. පසු අවස්ථාවක වුව එහි ඇති මෘදුකාංග නැවත මුල සිට ක්‍රියාත්මක කළ යුතු නම් විදුලි සැපයුම විසන්ධි කර නැවත විදුලි සැපයුම ලබාදිය යුතුය. නමුත් එම අවශ්‍යතාව ඉවත් කිරීම සඳහා මෙම ප්‍රත්‍යාරම්භක ස්විචය MDS උපාංගය මත තිබෙයි. මෙය වරක් තද කළ පසු වැඩසටහන මුල සිට ක්‍රියාත්මක වීම ඇරඹෙයි.

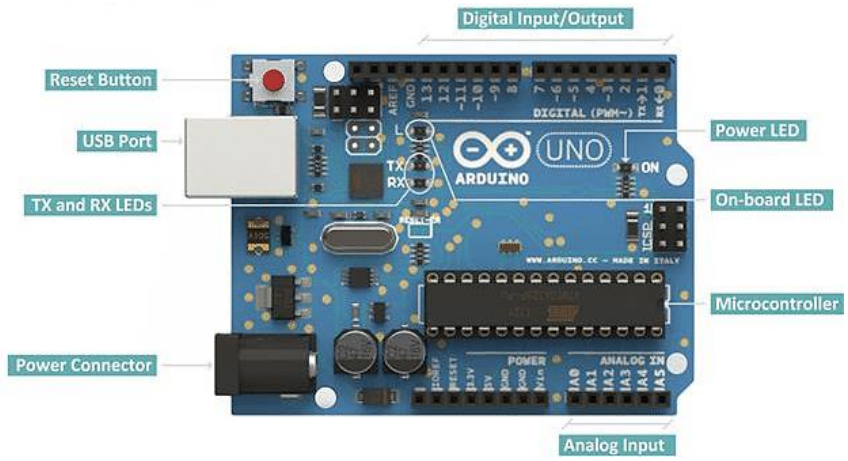
මෙම මූලික කොටස්වලට අමතරව එක් එක් MDS උපාංගවල ඒවාටම අනන්‍ය වූ ලක්ෂණ අන්තර්ගත වෙයි.

බහුලව භාවිතා වෙන ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධති

1) ආඩුර්නෝ (Arduino)

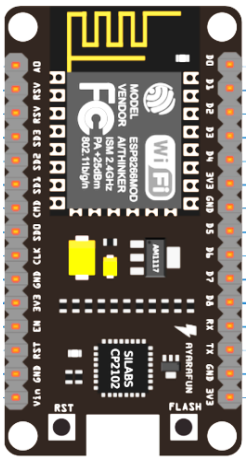
ආඩුර්නෝ යනු මයික්‍රෝකන්ට්‍රෝලර් හෙවත් ක්ෂුද්‍ර පාලකයක් සහිත ක්ෂුද්‍ර සකසන සංවර්ධන පද්ධතියකි. ආඩුර්නෝ බෝඩ් වර්ග කීපයකි. ආඩුර්නෝ උෆනෝ (Arduino UNO) යනු එම බෝඩ් අතරින් මිල අඩු එකකි. එමෙන්ම බහුලව භාවිතා කරන ආඩුර්නෝ බෝඩ් එක වෙන්නේ මෙයයි.

- ආඩුර්නෝ බෝඩ් එකෙහි අන්තර්ගතය



ආඩුර්නෝ බෝඩ් එක මත මයික්‍රොකන්ට්‍රෝලර් (Microcontroller) එක, බාහිර බල සැපයුමක් (Power Connector), යූඑස්බී කනෙක්ටරයක් (USB Port), ඇනලොග් හා ඩිජිටල් ඉන්පුට් / අවුට්පුට් පින් (I/O Pin), සහ තවත් උපාංග තිබෙණු පෙනේවි. සරල දැනුමක් ලබාගනු පිණිස පමණක් ආඩුර්නෝ බෝඩ් එකක නම් කරන ලද ඡායාරූපයක් මෙහි දක්වා ඇත.

2) නොඩ් එම්සීයූ (Node MCU)



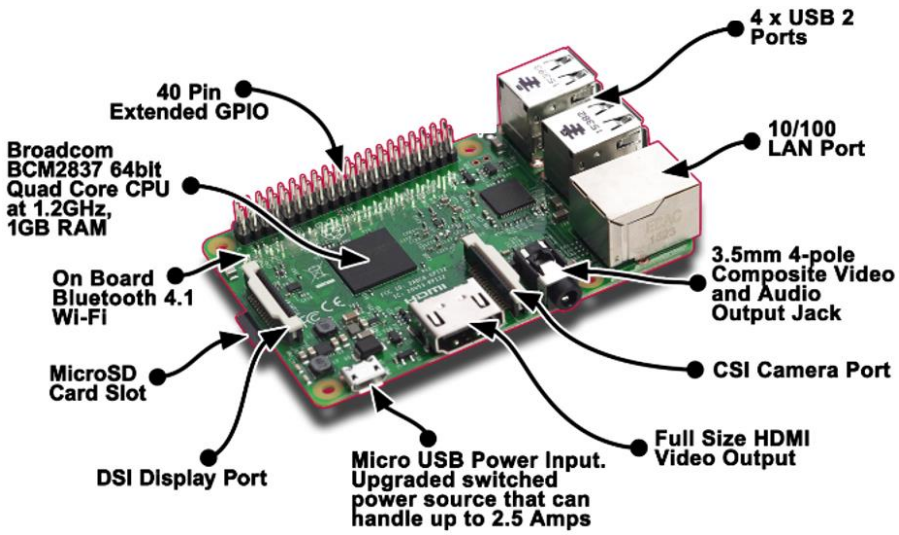
මෙයද ආඩුඊනෝ බෝඩ් එකෙහිම කුඩා ආකාරයකි. ප්‍රමාණයෙන් කුඩා වුවත් මෙහි ඇති ඇතැම් පහසුකම් ආඩුඊනෝ බෝඩ් එකෙහි ඇති පහසුකම්වලට වඩා වැඩිය. මෙහි ඇති ප්‍රධාන විශේෂත්වය නම් වයිෆයි පහසුකම අන්තර්ගත වීමය. මේ නිසා පහසුවෙන් අන්තර්ජාලය හරහා උපාංග හැසිරවීම කළ හැකිය. මිලෙන්ද අඩු බැවින් වර්තමානයේ බහුලව මෙය භාවිතා කොට IOT උපාංග තනා තිබේ. නොඩ් එම්සීයූ බෝඩ් එක මත Microcontroller එක, මයික්‍රෝ යූඑස්බී කනෙක්ටරයක් , ඇනලොග් හා ඩිජිටල් ඉන්පුට් / අවුට්පුට් පින් (I/O Pin), සහ තවත් උපාංග තිබෙණු පෙනේවි. සරල දැනුමක් ලබාගනු පිණිස පමණක් නොඩ් එම්සීයූ බෝඩ් එකක නම් කරන ලද ඡායාරූපයක් මෙහි දක්වා ඇත.

3) රාස්බෙරි පයි (Raspberry Pi)

ප්‍රමාණයෙන් කුඩා නමුත් සාමාන්‍ය පරිගණකයකින් කරගත හැකි වැඩ සියල්ලක්ම පාහේ කරගත හැකි පරිගණකයක් ලෙස රාස්බෙරි පයි පරිගණකය හැඳින්විය හැකිය. මෙය තනි පුවරුවේ පරිගණකයකි. (Single Board Computer) බැලූබැල්මට රාස්බෙරි පරිගණකය පෙනෙන්නේ කුඩා පරිපථයක් ලෙසය. වර්තමානයේ ප්‍රචලිත වෙමින් පවතින නිවෙස් ස්වයංක්‍රීය පද්ධති (Home automation system) ක්‍රියාත්මක කිරීම සඳහා බහුලව රාස්බෙරි පරිගණක භාවිතා කරයි.

මෙහි වයිෆයි, බ්ලූටූත්, NIC වැනි පහසුකම් අන්තර්ගත වෙන නිසා, අන්තර්ජාලය හරහා උපාංග හැසිරවීමට පහසුවෙන් හා අමතර උපාංග රහිතව වුව කළ හැකිය.

රාස්බෙරි බෝඩ් එකෙහි අන්තර්ගතය

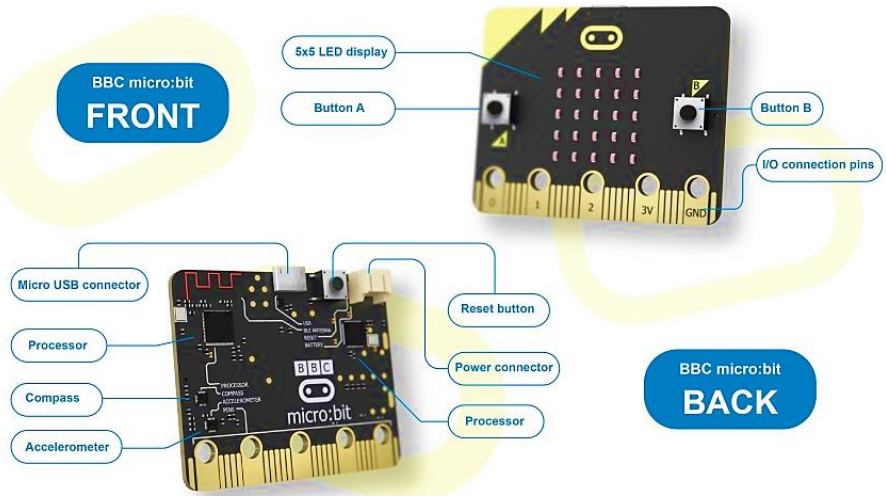


රාස්බෙරි බෝඩ් එක මත Quard Core ප්‍රොසෙසර් එක, Micro USB බාහිර බල සැපයුමක්, USB කනෙක්ටර් හතරක්, LAN පෝට් එකක්, HDMI කනෙක්ටර් එකක්, Micro SD ස්ලොට් එකක්, I/O Pin, ඇනලොග් ඉන්පුට් පින් සහ තවත් උපාංග තිබෙණු පෙනේවි.

4) මයික්‍රෝබිට් (Micro Bit)

ළමුන්ට ඉගෙන ගැනීම සහ ඉගැන්වීම සඳහාම නිර්මාණය කර ඇති උපාංගයක් ලෙස මයික්‍රෝබිට් උපාංගය හඳුන්වා දෙන්නට පුළුවන. මෙය BBC ආයතනයෙන් එක්සත් රාජධානියේ ළමුන්ගේ පරිගණක අධ්‍යාපනය වෙනුවෙන් හඳුන්වා දෙනු ලැබූවකි. වර්තමානය වෙන විට ඉගැන්වීමේ කටයුතු සඳහා බහුලව භාවිතා කරයි.

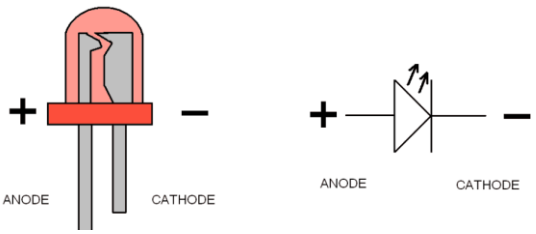
මයික්‍රෝ බිට් බෝඩ් එකෙහි අන්තර්ගතය



මෙහි ඉදිරිපස එල්ඊඩී බල්බ් 25 ක දර්ශකයක් තිබේ. මයික්‍රෝබිට් බෝඩ් එකෙහි දර්ශන තිරය මෙයයි. මීට අමතරව දෙපසින් A සහ B යනුවෙන් බටන් දෙකක් දක ගන්නට පුළුවන. මෙම බටන් මඟින් විවිධ වැඩසටහන් ක්‍රියාත්මක කරවන්නට පුළුවන. මයික්‍රෝබිට් බෝඩ් එකෙහි පසුපස පරිපථයේ විවිධ කොටස් දක ගන්නට පුළුවන. ඒවා අතර ප්‍රධාන වෙන්නේ මයික්‍රෝප්‍රොසෙසරය (Microprocessor), බ්ලූටූත් ඇන්ටනාව (Bluetooth antenna), ඇක්සලමීටරය සහ කොම්පාස් එක (Accelerometer and compass) ය. එමෙන්ම බ්ලොක්ලි (blockly) මත පදනම් වූ මයික්‍රොසොෆ්ට් බ්ලොක් එඩිටරය උපයෝගී කර ගෙන පහසුවෙන්ම ක්‍රමලේඛනය කළ හැකි වීමත් මෙහි ඇති විශේෂත්වයකි. එහිදී මයික්‍රොසොෆ්ට් බ්ලොක් එඩිටරය තුළ වූ මයික්‍රෝබිට් සිමියුලේටරය (Micro Bit Simulator) භාවිතා කොට දෘඩාංගය නැතිව වුව වැඩසටහන් අත්හදා බැලීමට හැකි වීම විශේෂත්වයකි. https://makecode.microbit.org යන වෙබ් ලිපිනය හරහා මයික්‍රොබිට් බ්ලොක් එඩිටරය වෙත පිවිසෙන්නට පුළුවන.

IOT **සඳහා නිතර අවශ්‍ය වෙන ඉලෙක්ට්‍රොනික උපාංග**

LED බල්බ - එල්ඊඩී බල්බයක් යනු ආලෝක විමෝචන ඩයෝඩයකි.



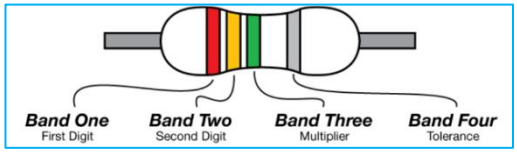
රෙසිස්ටර් - ප්‍රතිරෝධක හෙවත් රෙසිස්ටර් මඟින් විදුලිය ගලා යෑමට බාදාවක් ඇති කර වෝල්ටීයතාවය අඩු කරයි.





මූලික වශයෙන් රෙසිස්ටර් බාහිර ප්‍රමාණ කීපයකින් දැක ගන්නට පුළුවන. එම ප්‍රමාණවලින් රෙසිස්ටරයක වොට් ගණන දැක්වේ. කුඩාම රෙසිස්ටරයේ ප්‍රමාණය මිලිමීටර් 4 කි. එහි වොට් ප්‍රමාණය අටෙන් එකකි. මීළඟට ඇත්තේ මිලිමීටර් 6.5 ක ප්‍රමාණයේ රෙසිස්ටර් එකයි. එය වොට් කාලේ (¼) රෙසිස්ටර් එකයි. IOT සඳහා බහුලවම යොදා ගන්නේ මෙම ප්‍රමාණයේ රෙසිස්ටරයයි.

මෙම වොට් ගණනට අමතරව රෙසිස්ටර් එකක වැදගත්ම දෙය වන්නේ එහි ඕම්ස් අගයයි. ඕනෑම රෙසිස්ටර් එකක අගය දක්වා තිබෙන්නේ ඒ මත ඇති වර්ණවත් වළලු මඟින්ය. මූලික වශයෙන් වර්ණවත් වළලු හතරක් ඇති රෙසිස්ටර් බහුලව යොදා ගනු ලබයි. මෙම සෑම වර්ණ වළල්ලකටම අගයන් තිබේ. එම අගයන් ලැබෙන්නේ මෙසේය.



කළු = 0 / දුඹුරු = 1 / රතු = 2 / තැඹිලි = 3 / කහ = 4 / කොළ = 5 / නිල් = 6 / දම් = 7 / අළු = 8 / සුදු = 9 / රන් පැහැය = 5% ක වෙනස පෙන්වයි / රිදී පැහැය = 10% ක වෙනස පෙන්වයි.

රෙසිස්ටර් එකක් දෙස බැලීමේදී රන් හා රිදී පැහැය හැර අනෙක් වර්ණ තුනක් දැකගත හැකිය. අවසානයේ ඇත්තේ රන් හෝ රිදී පැහැති වර්ණයයි.



ඒ අනුව රෙසිස්ටරයක දුඹුරු, කළු හා රතු යන වර්ණයන් ඇතැයි සිතමු. එවිට දුඹුරු වර්ණයට අදාළ 1 යන අගයත්, කළු වර්ණයට අදාළ 0 යන අගයත් අප එක පෙළට ලියාගත යුතුය. අවසන් වර්ණය වන රතු වර්ණයේ අංකය වෙනුවට එම අංකයෙන් දැක්වෙන ඉලක්කමට අදාළ බින්දු ප්‍රමාණයක් දැක්විය යුතුය. ඒ අනුව එහිදී රතු වර්ණයට අදාළව බින්දු දෙකක් (00) යොදයි. එවිට අදාළ රෙසිස්ටරයේ අගය වෙන්නේ 1000 කි. මෙය ඕම් 1000 ක් හෝ කිලෝ ඕම් 1 ක් ලෙස දක්වනු ලැබේ.

සංවේදක (Sensor) සහ මොඩියුල (Module)

සරලව සංවේදකයක් හෙවත් සෙන්සර් එකක් යනු යම් භෞතිකමය වෙනස් වීමක් හඳුනාගත හැකි හෝ එසේ හඳුනාගෙන එයට ප්‍රතිචාර දැක්වීමට සමත් උපාංගයකි. මොඩියුලයක් ලෙස හැඳින්වෙන්නේ සෙන්සර් එකක් හෝ කීපයක් සමඟ තවත් පරිපථ කොටස් සම්බන්ධ කර ඇති උපාංගයකි. ඇතැම් මොඩියුලවල සෙන්සර් නැත. නිතර භාවිතා වෙන සෙන්සර් සහ මොඩියුල කීපයක් පිළිබඳ විස්තර මෙසේය.

LDR (Light Dependent Resistor - ආලෝක සංවේදී ප්‍රතිරෝධක)



අඳුර වැටෙන විට දැල්වෙන විදුලි බුබුලු මෙන්ම ආලෝකය ලැබීමත් සමඟ සක්‍රීය වන විදුලි උපාංග යනාදිය තැනීමේදී සංවේදක වශයෙන් යොදා ගන්නේ LDR සංවේදකය.

චුම්බක ස්විචය (Magnet switch)



දොරක් හෝ ජනේලයක් හෝ විවෘත කර ඇති බව මෙන්ම වසා ඇති බව හඳුනාගන්නට මෙම සංවේදකය භාවිතා කරයි. එසේ හඳුනා ගැනීමෙන් අනතුරුව අවශ්‍යතාවය මත සංඥා නිකුත් කිරීම වැනි දේ කළ හැකිය.

උෂ්ණත්ව සංවේදක (Temperature Sensor)



පරිසරයේ ඇති උෂ්ණත්වය හඳුනාගෙන ඒ අනුව ක්‍රියාකාරීත්වය වෙනස් විය යුතු උපාංග තැනීමට මෙය භාවිතා කරයි. කෙසේ වෙතත් මයික්‍රෝබිට් උපාංගයේ උෂ්ණත්ව සංවේදකයක් අන්තර්ගත බැවින් බාහිරින් උෂ්ණත්ව සංවේදකයක් සම්බන්ධ කරන්නේ නැතිව උපාංග ක්‍රියාත්මක කරවන්නට පුළුවන.

ක්ෂුද්‍ර සර්වෝ මෝටර් මොඩියුලය (Micro Servo Motor Module)



මෙය කුඩා දෘතිරෝද බොහෝමයක් භාවිතයෙන් ක්‍රියාත්මක වෙන මෝටරයකි. මෙහි ක්‍රියාකාරීත්වය MDS එකක් හරහා පාලනය කරන්නට පුළුවන. එමඟින් කුඩා උපාංග චලනය කිරීම වැනි දේ පහසුවෙන් කළ හැකිය.

වැසි හඳුනා ගැනීමේ මොඩියුලය (Rain Module)



මෙමඟින් වැසි වැටෙන බව හඳුනාගන්නට පුළුවන. එසේ හඳුනාගත් පසු වැසි වැටේ නම් යම් ක්‍රියාවක් වීමට සැලැස්වීම හෝ වැසි වැටෙන්නේ නැත්නම් යම් ක්‍රියාවක් වීම හෝ කළ හැකිය.

ජලමට්ටම හඳුනාගැනීමේ සංවේදකය (Water Level Sensor Module)



මෙය භාවිතා කරන්නේ ටැංකියක ජල මට්ටම හඳුනා ගැනීමටය. එසේ හඳුනා ගැනීමෙන් පසුව වෙනත් උපාංගයක් ක්‍රියාත්මක කරවීම හෝ දැනට ක්‍රියාත්මක වෙමින් පවතින මෝටරයක් වැනි උපාංගයක් නතර කිරීම හෝ කළ හැකිය.

චලන සංවේදක මොඩියුලය (Motion Sensor Module)



මිනිසෙකු හෝ සතෙක් හෝ චලනය වන විට ඒ බව හඳුනාගෙන යම් ක්‍රියාකාරකමක් විය යුතු පරිපථ සැකසීම සඳහා මෙම මොඩියුලය භාවිතා කළ යුතුය. ඒ අනුව මේ හරහා ලැබෙන ආදානයට අනුව යමෙක් මේ ඉදිරියේ චලනය වෙන විට යම් කාර්යයක් වීමට සැලැස්විය හැක.

අතිද්වනි මොඩියුලය (Ultrasonic Module)



වවුලෙකු පියාසර කරන විට අතිද්වනි තරංග නිකුත් කර ග්‍රහනය කර ගන්නා ආකාරයෙන් මෙම අතිද්වනි මොඩියුලය ක්‍රියාත්මක වන විටද ඉන් අතිද්වනි තරංග නිකුත් කර නැවත ඒවා ග්‍රහනය කර ගනී. ඒ අනුව මේ හරහා මීට ඉදිරියෙන් තිබෙන යම් උපාංගයක් හඳුනා ගන්නට පුළුවන. චලනය වන රොබෝ උපාංග ක්‍රියාත්මක වීමේදී ඉදිරියෙන් ඇති බාධක හඳුනාගැනීමට මෙම මොඩියුලය උවමනා වේ.

රිලේ මොඩියුලය (Relay module)



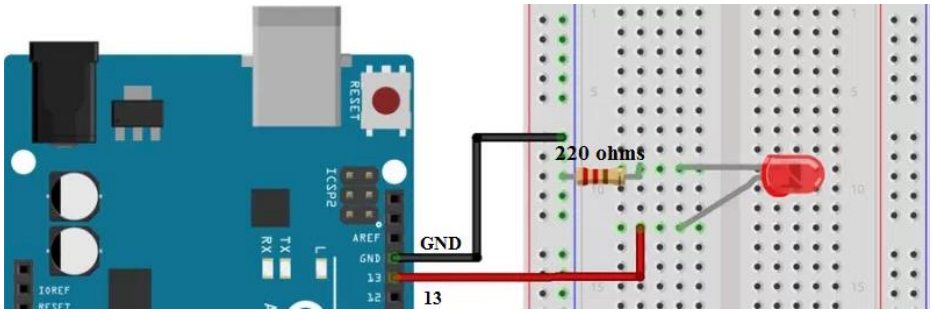
පරිපථයක් හරහා නිකුත් කරන්නේ කුඩා අංකිත සංඥාවකි. එවැනි අංකිත සංඥාවක් හඳුනාගෙන බාහිර විදුලි උපාංගයක් ක්‍රියා කරවන්නට නම් රිලේ මොඩියුල් එක යොදා ගත යුතුය. මෙම මොඩියුල් එක වෙත නිවසට ලැබෙන ප්‍රධාන විදුලිය සහ පරිපථය හරහා යොමු කරන සංඥාවන් යන දෙකම සම්බන්ධ කරනු ලැබේ.

මේවාට අමතරව විවිධ සංවේදක සහ මොඩියුල අවශ්‍යතාවය මත භාවිතා කරන්නට පුළුවන.

සරල යෙදුම් ක්‍රමලේඛ

බාහිරින් ඇති එක් එල්ඊඩී බල්බයක් ක්‍රියාත්මක කරවීම

මේ සඳහා ආඩුරිනෝ ඇටවුම පහත ආකාරයෙන් සකසා ගත යුතුය. මෙහි ඇති රෙසිස්ටරය 220 Ω වේ.

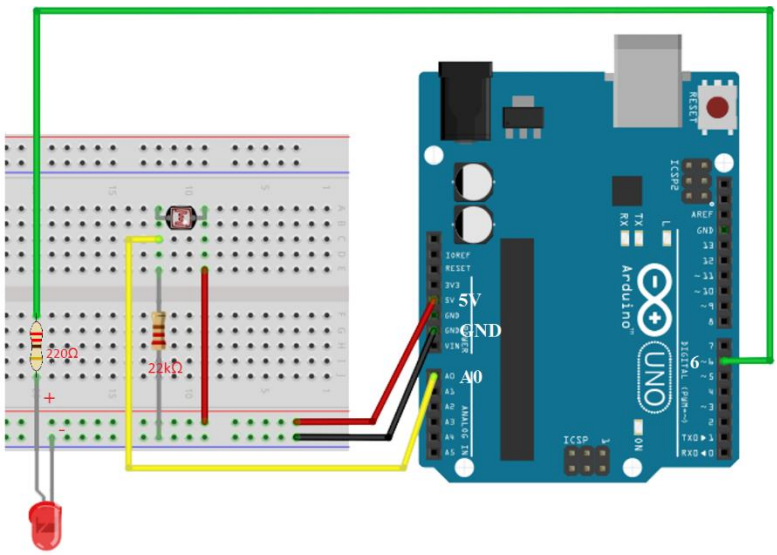


File > Examples > 01.Basics > Blink දැන් ඊට අදාළ කෝඩ් විවෘත වේ. මෙහිදී කෝඩ් තුළ පහළින් delay(1000) ලෙස අගයන් දෙකක් දැක ගන්නට පුළුවන. ඉන් මුලින් ඇති අගය delay(2000) ලෙස වෙනස් කළ යුතුය. දැන් ඉහළින් ඇති ටූල් බාර් එකෙන් දකුණු පසට ඇති ඊතල ලකුණ සහිත බටන් එක ක්ලික් කළ යුතුය. එවිට මෙම ප්‍රෝග්‍රෑම් එක ආඩුරිනෝ බෝඩ් එක මත ඇති මයික්‍රෝකන්ට්‍රෝල් එක වෙත අප්ලෝඩ් වෙයි. එම අප්ලෝඩ් වීමෙන් පසුව බෝඩ් එක මත ඇති කහ පැහැති එල්ඊඩී බල්බය තත්පර දෙකක් දැල්වී, තත්පරයක් නිවී තිබෙන ආකාරය දැක ගන්නට පුළුවන.

```
void loop() {
  digitalWrite(13, HIGH);
  delay(2000);
  digitalWrite(13, LOW);
  delay(1000);
}
```

LDR උපාංගය ආඩුඊනෝ සමඟ භාවිතා කිරීම

මේ සඳහා ආඩුඊනෝ බෝඩ් එක, බ්‍රෙඩ්බෝඩ් එක, LDR එකක්, 22kΩ, 220Ω ප්‍රතිරෝධක සහ LED බල්බයක් උවමනා කෙරේ.

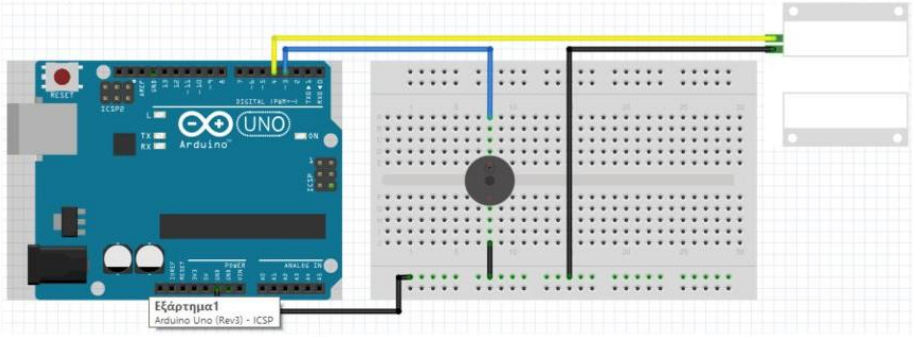


රැහැන් සම්බන්ධ කර ගැනීමෙන් පසුව ආඩුඊනෝ බෝඩ් පරිගණකය හා සම්බන්ධ කර ගත යුතුය. ඉන්පසුව පරිගණකයේදී ආඩුඊනෝ වැඩසටහන විවෘත කර පහත දැක්වෙන කේත ටයිප් කළ යුතුය.

```
int sensorReading;
void setup(){
pinMode(6,OUTPUT);
}
void loop(){
sensorReading=analogRead(0);
if (sensorReading<700){
digitalWrite(6,HIGH);
}
else digitalWrite(6,LOW);
delay(1000);}
```

චුම්භකිත ස්විචය මඟින් දොරක් විවෘත කර හෝ වසා ඇති බව හඳුනා ගැනීම

චුම්භකිත ස්විචයක් මඟින් පහසුවෙන්ම දොරක් විවෘත කර ඇති බව හෝ වසා ඇති බව හඳුනාගෙන ක්‍රියාත්මක වෙන පරිපථයක් සැකසිය හැකිය. මෙම කාර්යය ආඩුඊනෝ බෝඩ් එක හරහා කරන්නේ මෙසේය.



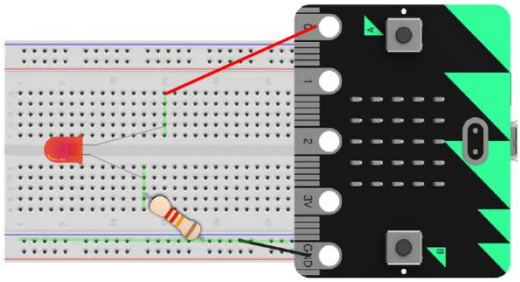
මෙහිදී සෘණ අන්තය ආඩුඊනෝ බෝඩ් එකෙහි GND පින් එකක් වෙත සම්බන්ධ කළ යුතුය. එහි අනෙක් අන්තය බසරයට සහ චුම්භකිත ස්විචයේ එක් අන්තයක් වෙත සම්බන්ධ කළ යුතුය. චුම්භකිත ස්විචයේ අනෙක් අන්තය ආඩුඊනෝ බෝඩ් එකෙහි 4 වැනි පින් එකටද, බසරයේ අනෙක් අන්තය ආඩුඊනෝ බෝඩ් එකෙහි 3 වැනි පින් එකටද සම්බන්ධ කළ යුතුය.

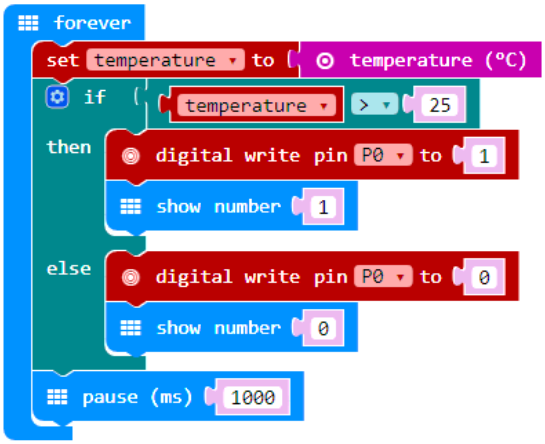
```
1. const int buzzer = 3;
2. const int sensor = 4;
3. int state;
4. void setup()
5. {pinMode(sensor, INPUT_PULLUP);}
6. void loop()
7. {state = digitalRead(sensor);
8. if (state == HIGH){
9. tone(buzzer, 400);}
10. else{noTone(buzzer);}
11. delay(200);}
```

මෙම කේත සටහනේ 8 වැනි පේළියේදී චුම්භකිත ස්විචය මඟින් ඉහළ සංඥාවක් ලැබෙද බලයි. එසේ නම් 9 වැනි පේළියෙන් බසරය නාද කරයි. එසේ නැත්නම් else කොටස ක්‍රියාත්මක වී බසරය නාද වීම නතර කරයි.

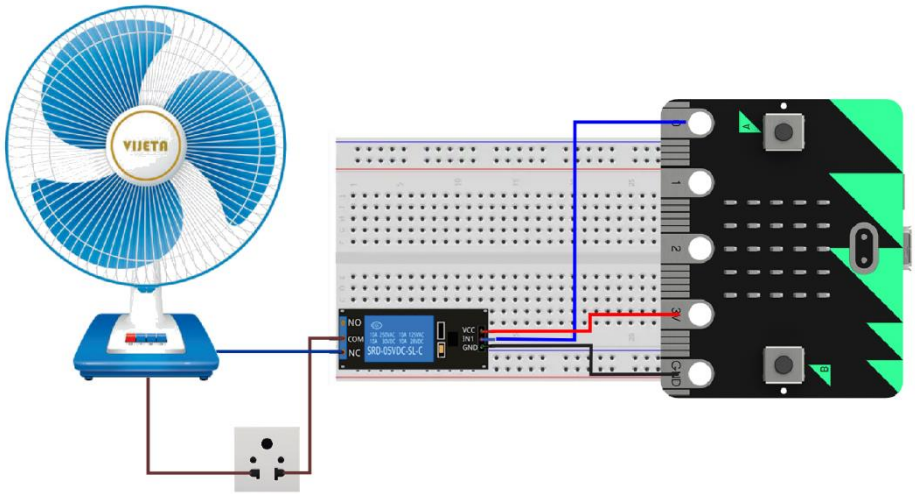
උෂ්ණත්වය වැඩි වන විට පංකාවක් ක්‍රියාත්මක වීමට සැලැස්වීම

මෙය මූලිකව තේරුම් ගැනීම සඳහා Micro Bit වලින් ඇටවුම පහත ආකාරයෙන් නිර්මාණය කර ගත යුතුය. කාමරයේ උෂ්ණත්වය හඳුනාගෙන එම උෂ්ණත්වය සෙල්සියස් අංශක 25 ට වැඩි වූ විට එල්ඊඩී බල්බය දැල්වීම සහ දර්ශකය මත 1 පෙන්වීම මෙහිදී වෙයි. උෂ්ණත්වය අඩු නම් එල්ඊඩී බල්බය නිවා දැමීමත්, දර්ශකය මත 0 පෙන්වීමත් සිදු වෙයි.





මෙම පරිපථයට රිලේ මොඩියුලයක් සම්බන්ධ කර LED බල්බය දැල්වීම වෙනුවට විදුලි පංකාවක් ක්‍රියාත්මක කරවීම හෝ නිවසේ ඕනෑම විදුලි උපාංගයක් සම්බන්ධ කර ක්‍රියාත්මක කරවීම කළ හැක. ඒ සඳහා පහත ආකාරයෙන් ඇටවුම සකස් කරගත යුතුය. මෙහිදී භාවිතා කළ යුත්තේ 3V රිලේ එකකි.

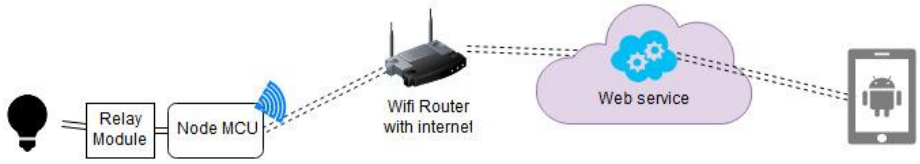


මෙහිදී කේත සටහනේ වෙනසක් වන්නේ නැත. LED බල්බය වෙත යොමු කළ සංඥාව රිලේ මොඩියුලයට යොමු කිරීමට පරිපථය සකස් කර ඇති බැවින් එම සංඥාව හරහා පංකාව ක්‍රියාත්මක වේ.

දුරස්ථ යතුරක් (remote switch) ගොඩනැඟීමට ඇති සරල IOT යෙදුම

මේ සඳහා IOT උපාංගයට අමතරව වයිෆයි රවුටරයක් සුහුරු දුරකථනයක් (Smart Phone) සහ ඇප් එකක් අවශ්‍ය කෙරේ. IOT උපාංගය සෘජුවම සම්බන්ධ වන්නේ වයිෆයි රවුටරය සමඟය. වයිෆයි රවුටරය වෙනුවට වයිෆයි හොට්ස්පොට් යෙදූ දුරකථනයක් වුව භාවිතා කළ හැකිය.

දුරස්ථ යතුරක් ගොඩනැඟීමට වයිෆයි පහසුකම සහිත MDS එකක් අවශ්‍ය වේ. එසේ නැත්නම් MDS එක සමඟ වයිෆයි මොඩියුලයක් සම්බන්ධ කර ගන්නට පුළුවන. මේ කාර්යය වඩාත් පහසුවෙන් Node MCU බෝඩ් එක භාවිතයෙන් කළ හැක. මේ සඳහා අවශ්‍ය වැඩසටහන ඇතුළු කිරීමට Node MCU බෝඩ් එක පරිගණකයට සම්බන්ධ කර ඉහත විස්තර කළ ආකාරයට ඩ්‍රයිවර්ස් ඉන්ස්ටෝල් කිරීම සහ පෝට් එක තේරීම කළ යුතුය.



මෙම සරල යෙදුම සඳහා blynk ලයිබ්‍රරි එක ඉන්ස්ටෝල් කර ගත යුතුය. ඒ සඳහා Arduino IDE එක තුළ ඇති මෙනු බාර් එකෙන් Sketch > Include Library > Manage Libraries හරහා යා යුතුය. මෙවිට විවෘත වෙන Manage Libraries තුළින් blynk ලයිබ්‍රරි එක සොයා ඉන්ස්ටෝල් කර ගත යුතුය.

දුරස්ථ යතුරක් ගොඩනැඟීම සඳහා අවශ්‍ය මූලික කේත ලබා ගැනීමට File > Examples > Blynk > Boards_WiFi > NodeMCU යන්න තේරිය යුතුය. මෙවිට ලැබෙන කේත සටහනේ පහත වෙනස්කම් කළ යුතුය.

1) char auth[] = "ඔබේ ටෝකන් අංකය"

2) char ssid[] = "වයිෆයි රවුටරයේ නම";

3) char pass[] = "රවුටරයේ මුර පදය";

ඔබේ ටෝකන් අංකය ලබාගැනීමට නම් දුරකථනයේ Blynk ඇප් එක ඉන්ස්ටෝල් කර තිබිය යුතුය. එම ඇප් එක ඇන්ඩ්‍රෙයිඩ් මෙහෙයුම් පද්ධතිය සඳහා මෙම QR එක හරහා ලබාගත හැක. මෙහි නිල වෙබ් අඩවිය www.blynk.cc වේ.

දුරස්ථ උපාංගය ක්‍රියාත්මක කරවන්නේද Blynk ඇප් එක හරහාය.

```
NodeMCU §
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
char auth[] = "34e4a43a0bc82f6d7e592";
char ssid[] = "AndroidAP";
char pass[] = "abcd1234";
void setup()
{Serial.begin(9600);
 Blynk.begin(auth, ssid, pass);}
void loop()
{Blynk.run();}
```

නිපුණතාවය 12- ව්‍යාපාර තුළ තොරතුරු සන්නිවේදන තාක්ෂණය

අංකිත ආර්ථිකය

තොරතුරු හා සන්නිවේදන තාක්ෂණය භාවිතය නිසා අංකිත ආර්ථිකයක් වර්තමානයේ ගොඩ නැගී ඇත. අංකිත සන්නිවේදන ජාල,පරිගණක මෘදුකාංග ඉහ අනෙකුත් තොරතුරු තාක්ෂණය අන්තර්ගත තාක්ෂණයන් මත පදනම් වූ ආර්ථිකයකි.

ආර්ථිකයක් තුල සිදුකෙරෙන කාර්යයන්

- භාන්ඩ ප්‍රචාරණය
- මුදල් ගෙවා භාණ්ඩ මිලදී ගැනීම
- භාණ්ඩ ප්‍රවාහණය
- තොග පාලනය
- රක්ෂණ පහසුකම්

ආදි කාර්යයන් පහසු කර ගැනීම සඳහා සුදුසු පරිදි තාරතුරු තාක්ෂණ මෙවලම් යොදා ගෙන නවීකරණය වූ ආර්ථිකයක්, **අංකිත ආර්ථිකයක්** ලෙස හඳුන්වයි.

අංකිත ආර්ථිකයක භාවිතා වන ව්‍යාපාරික ක්‍රමවේද

- ප්‍රති වෙන්දේසිය
- සමූහ මිලදී ගැනීම
- විද්‍යුත් වෙළඳපොළ

වෙන්දේසිය

භාණ්ඩ හා සේවා විකිණිමේ හා මිලදී ගැනීමේ ක්‍රියාවලියේදී ගැණුම්කරුවන් හා වෙළඳුන් අතර භාණ්ඩ හෝ සේවා වල මිල තීරණය කිරීම සඳහා ලන්සු තැබීම වෙන්දේසියක් ලෙස හැදින්වේ.
මෙය ආකාර දෙකකි
1. සාමාන්‍ය වෙන්දේසිය
2. විලෝම වෙන්දේසිය

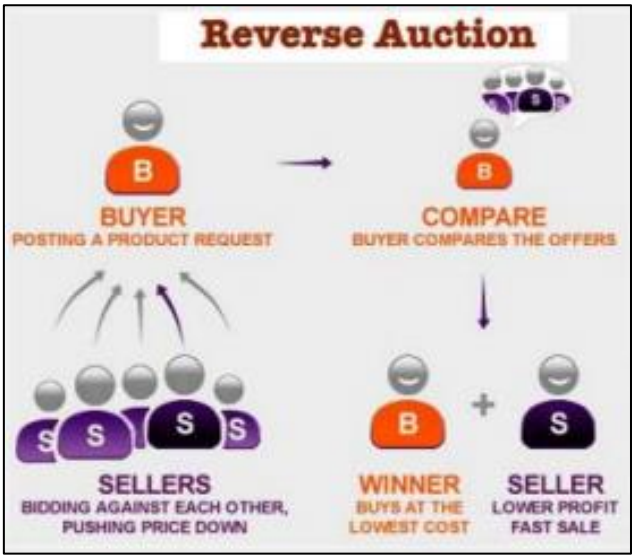
සාමාන්‍ය වෙන්දේසිය

මෙහිදී ගැණුම්කරුවන් තරගකාරීව ලන්සු තැබීමෙන් භාණ්ඩයේ මිල තීරණය කරයි.මෙහිදී කාලයත් සමග භාණ්ඩයේ මිල ඉහළ යයි.වෙන්දේසිය අවසන් වන්නේ වැඩිම ලන්සු තබන්නා වෙත භාණ්ඩය අත්වූ පසුවය.



විලෝම වෙන්දේසිය

ගැණුම්කරුවන් විසින් භාණ්ඩ හෝ සේවා ඉල්ලුම් කරයි.සාමාන්‍ය වෙන්දේසියක දී මෙන් නොව මෙහිදී විකුණුම්කරුවන් තරගකාරීව භාණ්ඩයේ මිල පහත දමමින් මිල තීරණය කරයි.මෙහිදී කාලයත් සමග භාණ්ඩයේ මිල පහල යයි.



අන්තර්ජාලය මගින් ද විලෝම වෙන්දේසිය සඳහා පහසුකම් සලසයි.එම ජනප්‍රිය වෙබ් අඩවි ලෙස,

www.ariba.com / www.sorcity.com / www.hedgehog.com දැක්විය හැකිය.

- අන්තර්ජාලය පාදක කරගත් විලෝම වෙන්දේසිය මගින් ,
  - ✓ මිල අඩු කර ගැනීමට අවස්ථාව ලැබීම
  - ✓ කාලය ඉතිරි කර ගැනීම වැනි පහසුකම් සැලසේ.

කණ්ඩායම් ලෙස මිලදී ගැනීම

භාණ්ඩ මිලදී ගැනීමේ දී යම් භාණ්ඩයකින් වැඩි ප්‍රමාණයක් මිලදී ගන්නා විට එයට වට්ටමක් හිමි වේ. එම නිසා එහි මිල අඩු වේ.

භාණ්ඩ විකුණුම් ප්‍රමාණය වැඩි කර ගැනීම සඳහා යොදා ගන්නා උපක්‍රමයක් ලෙස මෙය හැදින්විය හැකිය.

විද්‍යුත් වෙළඳපොළ

ගැණුම්කරුවන් හා විකුණුම්කරුවන් අතර ගණුදෙණු සිදුවන්නේ වෙළඳපොළ තුලය.මෙහි පාරිභෝගිකයන්, තොග වෙළෙන්දන්, සිල්ලර වෙළෙන්දන් හා නියෝජිතයින් දැකිය හැකිය.මෙය වෙබ් අඩවියක් මුලික කරගෙන සිදු වන්නේ නම් එය විද්‍යුත් වෙළඳපොළ කි.

මෙය කොටස් දෙකකින් සැදුම් ලත්ය.

01. ඉදිරි අන්තය

02. පසු අන්තය

ඉදිරි අන්තය

ගැණුම්කරුවන් හා විකුණුම්කරුවන් මුණගැසෙන්නේ මෙම අන්තයේදීය.භාණ්ඩ හා සේවා පිළිබඳ විස්තර, ඒවායේ මිල ගණන් මෙහි අන්තර්ගතය.

පසු අන්තය

නිෂ්පාදකයාගෙන් හෝ තොග වෙලෙන්දාගෙන් භාණ්ඩ මිලදී ගෙන ඒවා ගබඩා කර ගැණුම්කරුවන් ඒවා මිලදී ගැනීම දක්වා වූ ක්‍රියාවලිය මෙයට අයත් වේ.

වෙළඳ සංවිධාන ආකාර

01. මාර්ග අපගත ව්‍යාපාර

02. මාර්ග අපගත හා මාර්ග ගත ව්‍යාපාර

03. මාර්ග ගත ව්‍යාපාර

මාර්ග අපගත ව්‍යාපාර

මිලදී ගැනීමේ හා විකිණීමේ ක්‍රියාවලිය සඳහා අන්තර්ජාලය භාවිතා නොකරන ව්‍යාපාර මෙනමින් හැඳින් වේ.

මාර්ග අපගත හා මාර්ග ගත ව්‍යාපාර

මිලදී ගැනීමේ හා විකිණීමේ ක්‍රියාවලිය සඳහා අන්තර්ජාලය භාවිතා කරනවා සේම අන්තර්ජාලය භාවිතා නොකර භෞතිකව ව්‍යාපාර ආයතනයක් තුලද සිදු වන වෙළඳාම මෙලෙස හැඳින් වේ.

මාර්ග ගත ව්‍යාපාර

මිලදී ගැනීමේ හා විකිණීමේ ක්‍රියාවලිය සඳහා අන්තර්ජාලය පමණක් භාවිතා කරන ව්‍යාපාර මෙලෙස හැඳින් වේ.

ව්‍යාපාර ක්‍රියාකාරීත්වය කෙරෙහි තොරතුරු තාක්ෂණයේ භූමිකාව

- ✓ ගිණුම්කරණය හා තොරතුරු තාක්ෂණය
- ✓ මානව සම්පත් හා තොරතුරු තාක්ෂණය
- ✓ නිෂ්පාදන ක්‍රියාවලිය හා තොරතුරු තාක්ෂණය
- ✓ අලෙවිකරණය හා තොරතුරු තාක්ෂණය
- ✓ විකුණුම් හා තොරතුරු තාක්ෂණය
- ✓ සැපයුම්දාම කළමණාකරණය හා තොරතුරු තාක්ෂණය
- ✓ ව්‍යාපාර සන්නිවේදනය හා තොරතුරු තාක්ෂණය

විද්‍යුත් වාණිජ්‍ය

අන්තර්ජාලය හරහා භාණ්ඩ හා සේවා විකිණීම, මිලදී ගැනීම හෝ හුවමාරුව මෙලෙස හඳුන්වයි.

විද්‍යුත් ව්‍යාපාර

ව්‍යාපාරයක සියළුම ක්‍රියාකාරකම් සඳහා තොරතුරු හා සන්නිවේදන තාක්ෂණය යොදා ගැනීම මගින් ව්‍යාපාර කටයුතු කරගෙන යාම මෙනමින් හඳුන්වයි.

විද්‍යුත් ව්‍යාපාරික ගණුදෙණු වර්ග

- B2B
- B2C
- C2C
- C2B
- B2E
- G2C

ව්‍යාපාරයෙන් ව්‍යාපාරයට- B2B

ව්‍යාපාර ව්‍යාපාර අතර මාගර්ගතව සිදුවන වානිජමය ගණුදෙණු මෙම ගණයට අයත්ය. නිශ්පාදකයා හා තොග වෙළෙන්දා අතරත් තොග වෙළෙන්දා හා සිල්ලර වෙළෙන්දා අතරත් සිදු වන්නේ මෙවැනි ගණුදෙණුය.

උදා - අන්තර්ජාල වෙබ් අඩවියක් හරහා රථවාහන ආනයන කරන වෙළඳ ආයතනයක් විදේශ රටක රථවාහන සමාගමකින් රථ වාහන ඇනවුම් කිරීම

ව්‍යාපාරයෙන් පාරිභෝගිකයාට - B2C

නිශ්පාදකයා හා පාරිභෝගිකයා අතර හෝ බෙදාහරින්නා හා පාරිභෝගිකයා අතර හෝ සිල්ලර වෙළෙන්දා සහ පාරිභෝගිකයා අතර මාගර්ගතව සිදුවන ගණුදෙණු මෙලෙස හඳුන්වයි.

උදා - අන්තර්ජාල වෙබ් අඩවියක් හරහා විශ්‍රාමික සාමාන්‍ය පුද්ගලයකු විසින් කොටස් වෙළඳපලෙහි ලියාපදිංචි ප්‍රසිද්ධ සමාගමකින් කොටස් මිලදී ගැනීම

පාරිභෝගිකයාගෙන් පාරිභෝගිකයාට - C2C

අන්තර්ජාගය හරහා **පාරිභෝගිකයෙක් තවත් පාරිභෝගිකයෙකුට භාන්ඩ හෝ සේවා විකිණීම මෙලෙස හැදින් වේ.**

උදා - අන්තර්ජාල වෙබ් අඩවියක් හරහා පුද්ගලයකු විසින් භාවිත කරන ලද වාහනයක් තවත් පුද්ගලයකුට අලෙවි කිරීම

පාරිභෝගිකයාගෙන් ව්‍යාපාරයට - C2B

යම්කිසි **පාරිභෝගිකයෙකු ව්‍යාපාරයක් වෙත යම් සේවාවක් ලබා දීම මෙවැනි ගණුදෙණු වලදි සිදුවේ.**

උදා - අන්තර්ජාල වෙබ් අඩවියක් හරහා පුද්ගලයෙකු විසින් භාවිතා කරන ලද වාහනයක් වාහන අලෙවි කරන ආයතනයක් විසින් නැවත විකිණීමේ අරමුණින් මිලදී ගැනීම

ව්‍යාපාරයෙන් රැකියා නියුක්තිකයින්ට - B2E

ව්‍යාපාරය හා එම ආයතනය තුල සිටින සේවකයින් අතර සිදුවන ගණුදෙණු විශේෂයකි.

උදා - යම් ප්‍රසිද්ධ සමාගමක් තම ආයතනය සතු කොටස් අන්තර්ජාල වෙබ් අඩවියක් හරහා තම සේවකයෙකුට අලෙවි කිරීම

රජයෙන් පාරිභෝගිකයාට - G2C

රජයක් සිය පුරවැසියන්ට යම් යම් සේවා සපයයි.එම සේවා මාගර්ගතව සැපයීම මෙහිදි සිදුවේ

උදා - ශ්‍රී ලංකාවේ සමහර පළාත් රථවාහන සඳහා ආදායම් බලපත්‍ර මාර්ගගතව ලබා දීම.

විද්‍යුත් ව්‍යාපාර වල වාසි

- භාන්ඩ විකිණීමෙන් පසු සහය දිම හා මගපෙන්වීම
- ආකර්ෂණීය භාණ්ඩ තෝරා ගැනීම
- තරගකාරී මිල ගණන්
- තෘප්තිමත් බව සහතික වීම
- වේගවත් සේවාවන්
- පහසුවෙන් ලගා විය හැකිය
- ඉක්මනින් මිලදී ගැනීම හා විකිණීම
- දැන්වීම්
- තැගිදීමි , වට්ටම් දීම

විද්‍යුත් ව්‍යාපාර වල අවාසි

- ඇණවුම් කල භාණ්ඩ වල විස්වාසනීයත්වය ගැටළු සහිතවීම
- විකුණන සමාගම් අතර සබඳතාවයක් නොමැතිවීම
- මුදල් ගෙවීමේදී ණයපත් වල විශ්වාසනීය බව පිළිබඳ ගැටළු ඇතිවීම
- ඉතිරි මුදල් ගැන ආනාරක්ෂිත බව
- එවනු ලබන භාණ්ඩ වල අනාරක්ෂිත බව

13 වන ඒකකය - තොරතුරු තාක්ෂණයේ නව නැඹුරු හා අනාගත දිශානති

මෑත කාලීනව තොරතුරු හා සන්නිවේදන තාක්ෂණයේ වෙනස් වීම් හා වර්ධනය වන ක්ෂේත්‍ර ගැන විමසා බැලීමේ දී ශීඝ්‍රයෙන් සංවර්ධනය වන අංශ කිහිපයක් හඳුනා ගත හැකිය. මේවා අතර කෘත්‍රිම බුද්ධිය, සාර්ව ද්‍රව්‍ය අන්තර්ජාලය හා ක්වොන්ටම් පරිගණනය ඉදිරියෙන් සිටියි. අප සාර්ව අන්තර්ජාලය පිලිබඳව පුළුල්ව මෙයට ඉහත ඒකකයක දී කතා කළෙමු.

අප එදිනෙදා භාවිත කරන පරිගණක වලට මිනිසාට මෙන් තේරුම් ගැනීමේ හා හැඟීම් ප්‍රකාශ කිරීමේ හැකියාව නොමැත. මෙම පරතරය අඩුකර ගනිමින් මිනිසා මෙන්ම ඉහත හැකියා දැක්විය හැකි පරිගණක නිර්මාණය කිරීමට වර්තමානයේ දී පරීක්ෂකයන් උනන්දුවක් දක්වනු ලැබේ. මෙලෙස නිර්මාණය කර ගනු ලබන බුද්ධිය කෘත්‍රිම බුද්ධිය ලෙස හැඳින්වේ.

කෘත්‍රිම බුද්ධිය සමඟ යෙදවුම් ගණනාවක් එළි දැක ඇති අතර ගැඹුරු ස්නායුනාල (Deep Natural Networks) දැනුම් ප්‍රස්තාර (knowledge Graphs), මිනිස් මොළය පරිගණක පද්ධති මගින් නිර්මාණය කරන අතුරු මුහුණත් මෙන්ම ගූගල් , ටෙස්ලා හා ඇපල් වැනි සමාගම් පර්යේෂණ වල යෙදෙන රියදුරන් රහිත මෝටර් රථ (Self Driving Vehicles)තාක්ෂණය වැනි ක්ෂේත්‍ර වඩා වැඩි උනන්දුවක් මෙන්ම ආකර්ශනයක් ද පරේෂකයන් තුළ මෙන්ම මේ පිළිබඳව උනන්දු වන පුද්ගලයන් අතර ද ඇති කිරීමට සමත් වී තිබේ.



ටෙස්ලා ආයතනය මගින් නිර්මාණය කරන ලද පූර්ණ ස්වයං ධාවන වැඩසටහන

කෘත්‍රිම බුද්ධිය සහිත පරිගණකයක දැකිය හැකි ලක්ෂණ කිහිපයකි.

- බාහිර පරිසරයෙහි සිදුවන සිද්ධීන් හඳුනා ගැනීමේ හැකියාවෙන් යුතු වීම.
- තීරණ ගැනීම සඳහා අවශ්‍ය වන බුද්ධිය දියුණු කර ගැනීමේ හැකියාවෙන් යුතු වීම.
- අත්දැකීම් ඇසුරින් දැනුම ස්වයං වර්ධනය කර ගැනීමේ හැකියාව
- පරිසරයෙහි සිදුවන ක්‍රියාවලීන්ට අනුව ක්‍රියා කිරීමේ හා අනුගත වීමේ හැකියාවෙන් යුක්ත වීම.

කෘත්‍රිම බුද්ධිය පිළිබඳව කතා කිරීමෙහි දී එහි ස්වභාවය අනුව ප්‍රධාන ආකෘති දෙකකට බෙදා දැක්විය හැකිය.

1- **ශක්තිමත් ආකෘතිය -** කෘත්‍රිම බුද්ධි පද්ධතියකට සිතා කටයුතු කිරීමේ හැකියාවක් නැතහොත් ස්වයං ඉගෙනුමක් සමඟින් ඇති පද්ධතීන් මේ යටතට ගැනිය හැකි ය.
2- **දුර්වල ආකෘතිය -** කෘත්‍රිම බුද්ධි පද්ධතියකට එයට ලබා දෙන ලද දේමත පමණක් සිතා කටයුතු කළ හැකි වීම මින් අදහස් වේ. මේ වන විට ද භාවිත වන බහුතරයක්ම කෘත්‍රිම බුද්ධි වැඩසටහන් මෙම දෙවැනි වර්ගයට අයත් වේ.

මෙම පටු බුද්ධිමත් භාවය අවබෝධ කර ගැනීම සඳහා මිනිසුන් විවිධ තාක්ෂණයන් නිර්මාණය කර ගෙන තිබේ.

(වගුවට අදාල දත්ත උපුටා ගැනීම 13 ශ්‍රේණිය ගුරුඅත්පොතට අදාල සම්පත් පොත ඇසුරෙනි)

තාක්ෂණය	**තාක්ෂණයට අදාල විස්තරය**
සෙවීම් කාර්යයන්	ලෝක සිතියම සටහන් කර ඇති රටවල් අතරින් රටක් හෝ ස්ථානයක් සොයා ගැනීම.
ප්‍රවීණ පද්ධති	ක්‍රමලේඛනයක අඩංගු පාලන ව්‍යුහ හා නීති භාවිත කරමින් දැනුම එක් රැස් කරගැනීම හා ඒවා සාමාන්‍ය ජනයාට අවශ්‍ය කාර්යයන් වල දී ලබා දීම
ස්වාභාවික භාෂා සැකසුම්	විවිධ මානව භාෂා හඳුනා ගැනීම හා තේරුම් ගැනීමේ කාර්යය
කථනය හඳුනා ගැනීම	- උදාහරණයක් ලෙස ඇමේසන් ඇලෙක්සා වැනි Digital Assistant වැනි උපාංග වල මිනිසා විසින් ලබා දෙන උපදෙස් වලට අදාලව තොරතුරු සොයා දීම හා උපාංග හැසිරවීමට අදාල වගන්ති හඳුනා ගැනීම සිදු වේ.
පරිගණක දැක්ම සහ දර්ශනය	වස්තූන් වෙන්කර හඳුනා ගැනීම මෙන්ම කැමරාවක් මගින් ලබා ගත් අන්තර්ගතය පරිවර්තනය සඳහා උපකාරී කිරීම
යන්ත්‍ර ඉගෙනුම	දත්ත එක් රැස් කොට එම දත්ත රටා වල සබඳතා හඳුනා ගනිමින් ඒ ඇසුරින් පුරෝකථන මෙන්ම තොරතුරු ලබා ගැනීම. උදාහරණයක් ලෙස පුද්ගලයන් භාවිත කරන අතෙහි බඳින ස්මාට් ඔරලෝසුවක් හරහා අදාල පුද්ගලයාට අදාල සෞඛ්‍ය තත්ත්වය මැන බලන දත්ත රැස්කොට ඔහුගේ ඇතිවිය හැකි අනරාක්ෂිත අවස්ථාවක් ගැන ක්ෂණිකව තොරතුරු ලබා දීම.
ස්නායුක ජාල පද්ධති(Neural Network)	මිනිස් මොලයේ ස්වභාවය හා එහි නියුරෝධ පද්ධතියෙහි සැකසුම අධ්‍යයනය කරමින් සකසන ලද කෘත්‍රිම බුද්ධිය මත පදනම් වූ පද්ධතියකි.මිනිස් මොලය විසින් ගැටලුවක් විසඳීමේ දී එය විශ්ලේශනය කරන ආකාරය හා ඊට විසඳුම් ලබා ගන්නා ආකාරය මෙවැනි පද්ධතියක් ගොඩනැගීමේ දී අධ්‍යයනය කරනු ලැබේ.

Question to Watch

කෘත්‍රිම බුද්ධිය සම්බන්ධයෙන් පහත දැක්වෙන කුමන වගන්තිය සත්‍ය වේ ද?
(1) විශේෂඥ පද්ධතියක අනුමාන යන්ත්‍රය (Inference Engine) මානව තීරණ ගැනීමේ ක්‍රියාවලිය මුළුමනින් ම අනුකරණය කරයි.
(2) විශේෂඥ පද්ධතියක දැනුම් සමුදායක් තුළ (Knowledge Base) විශේෂඥයකුගේ ප්‍රාථමික දැනුම පමණක් අඩංගු වේ.
(3) විශේෂඥ පද්ධති සැමවිට ම ජානමය ඇල්ගොරිතම (Genetic Algorithms) මගින් ක්‍රියාත්මක කෙරේ.
(4) ජානමය ඇල්ගොරිතම මගින් දී ඇති ගැටලුවක් සඳහා එක් පිළිතුරක් පමණක් ලබා දේ.
(5) ජානමය ඇල්ගොරිතම දී ඇති ප්‍රශ්නයක විසඳුම පුනර්කරණ (iterations) කිහිපයක දී පරිණාමනය කරයි.

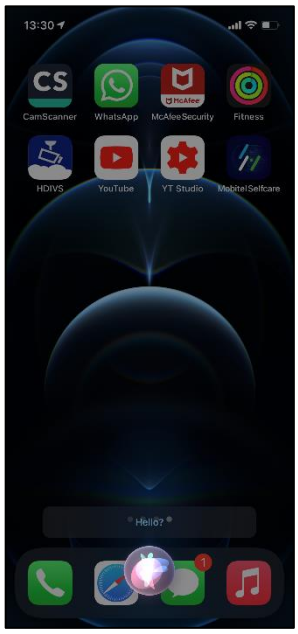
මෘදුකාංග කාරක නැතහොත් මෘදුකාංග නියෝජිත (Software Agents) යනුවෙන් අදහස් වන්නේ, කුමක්ද?

මෘදුකාංග නියෝජිතයින් සාම්ප්‍රදායික පරිගණක වැඩසටහන් වලින් ඔබ්බට ගමන් කළ පරිණාමීය පියවරක් නියෝජනය කරයි. මෘදුකාංග නියෝජිතයින්ට තමන් විසින්ම ක්‍රියාත්මක කර ක්‍රියාත්මක වීමේ හා අනුගත වීමේ හැකියාව ඇත. මෙනිසා මෙවැනි මෘදුකාංගයකට මිනිස් පරිශීලකයෙකුගේ අදහස් හෝ අන්තර්ක්‍රියා අවශ්‍ය නොවේ. මෘදුකාංග නියෝජිතයින්ට යෙදුම් සහ මාර්ගගත බුද්ධිමත් නියෝජිතයන් ඇතුළු වෙනත් වැඩසටහන් හෝ නියෝජිතයන් ආරම්භ කිරීමට, අධීක්ෂණය කිරීමට සහ අවසන් කිරීමට හැකිය.

වර්තමානයෙහි දී බොහෝ කාර්යයන් සඳහා මෘදුකාංග නියෝජිතයන් භාවිත වනු දැකිය හැකිය.

වින්ඩෝස් පාරිසරිකයෙහි ක්‍රියාත්මක කළ හැකි Cortana සහ iOS හා මැක් මෙහෙයුම් පද්ධතියෙහි ක්‍රියාත්මක වන Apple Siri මෙවැනි මෘදුකාංග නියෝජිතයන් දෙකකට උදාහරණ ලෙස සැලකිය හැකිය. ප්‍රධාන වශයෙන් මෘදුකාංග නියෝජිත ලෙස සැලකෙන ඉහත ඩිජිටල් ඇසිස්ටන් වැඩසටහන් දෙක මිනිස් කටහඬ හඳුනා ගනිමින් එම විධාන වලට අනුව කටයුතු කරයි.





මේ හැරුණු විට මෘදුකාංග නියෝජිත භාවිත වන අවස්ථා කිහිපයක් පහත පරිදි දැක්විය හැකිය.
ස්වයංක්‍රීයව ඉලක්ක ගත අන්තර්ජාල සෙවීම් සිදු කිරීම සඳහා,
පැමිණෙන විද්‍යුත් තැපෑල පරීක්ෂා කර වඩා වැදගත් විද්‍යුත් තැපැල් ලිපි හඳුනා ගනිමින් ඒවා අතරින් ප්‍රමුඛතාවය ලබා දිය යුතු ඒවා තෝරා ගැනීම සඳහා .
විද්‍යුත් පෝරම ස්වයංක්‍රීයව පිරවීම වැනි කාර්යයන් සඳහා.

මාර්ගගත රැකියා සෙවීම් සිදු කිරීම සඳහා,
සමාජ ජාල පැතිකඩ සමමුහුර්ත කිරීම වැනි කාර්යයන් සඳහා.
ඊ-වාණිජ්‍යය තුළ වඩා වාසිදායක ගනුදෙනු හඳුනා ගැනීම සඳහා හා වඩා වාසිදායක ගනුදෙනු කිරීම සඳහා.

පොදුවේ ඕනෑම මෘදුකාංග නියෝජිතයෙකු තුළ දැකිය හැකි ලක්ෂණ කිහිපයක් හඳුනා ගත හැකි ය.

නම්‍යශීලීබව - වෙනස් වන පරිසර තත්ත්වයන් වල දී ඒ අනුව හැඩ ගැසීමේ හැකියාව මේවාට ඇත.

ස්වයං පාලන හැකියාවෙන් යුතු වීම - පරිශීලකයෙකු නොමැතිව ක්‍රියා කිරීම මෙන්ව තීරණ ගැනීම හා ප්‍රතිචාර දැක්වීමේ හැකියාව තිබේ.

සන්නිවේදන හැකියාවෙන් යුතු වීම- වෙනත් කාරක අතර මෙන්ම පද්ධතීන් අතර ද සන්නිවේදන මෙම නියෝජිත වලට තිබේ.

අර්ථවත් ආකාරයෙන් හැසිරීමේ හැකියාවෙන් යුතු වීම- මෘදුකාංග නියෝජිතයෙක් පරිශීලකයෙකු වෙනුවට පද්ධතියක් හැසිරවීමට භාවිත වන ඒකකයක් බැවින් එයට අවස්ථාවට ක්‍රියාකිරීමේ හැකියාවක් තිබේ.

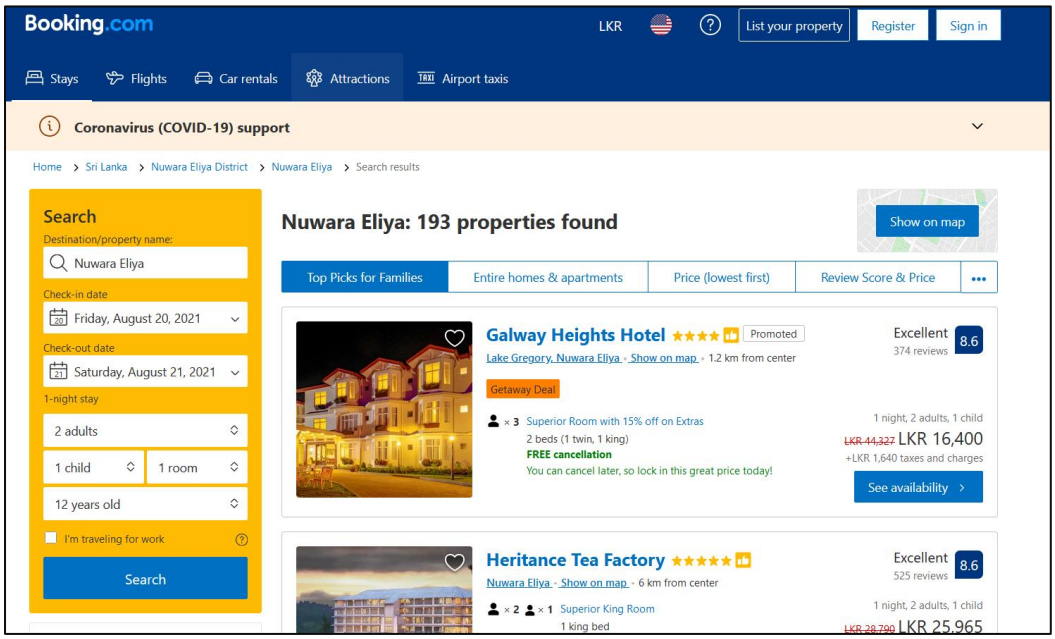
බහුකාරක පද්ධතියක් යනු කුමක්ද?

කාරක ගණනාවකින් සමන්විත වූ ඒවා එකිනෙක ක්‍රියාකරන පද්ධතියක් බහුකාරක පද්ධතියක් ලෙස හැඳින්විය හැකි ය.

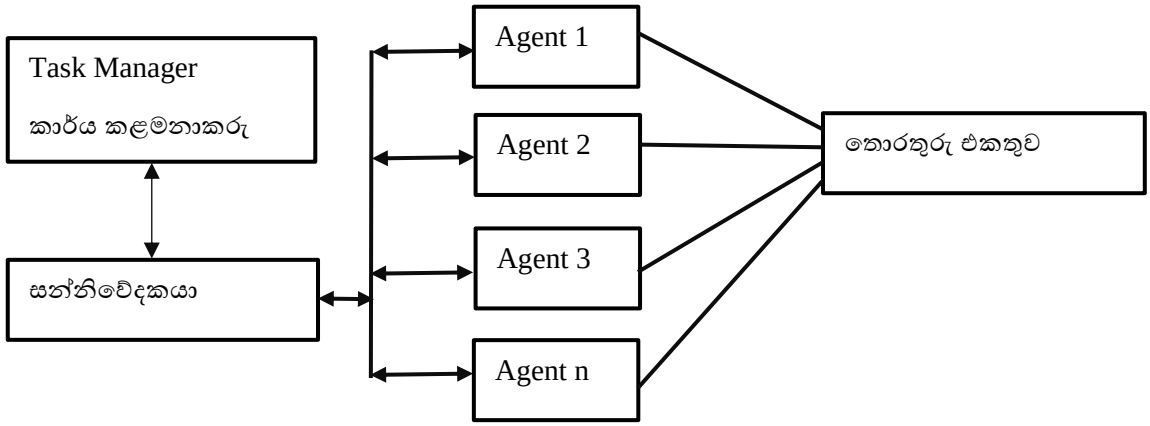
මෙම කාරණය වඩාත් පැහැදිලි කර ගැනීමට අපි එදිනෙදා සරල උදාහරණයක් ගනිමු. මිනිස් සිරුරෙහි ජීවය පවත්වා ගෙන යාම සඳහා එකිනෙකට සම්බන්ධිත පද්ධති රාශියක් සිරුර තුළ දැකිය හැකිය. ඒවා අතර මනා සම්බන්ධතාව හා නිසි ක්‍රියාකාරීත්වය නිසා ජීව පද්ධතිය නිසි ලෙස පවත්වා ගත හැක. බහුකාරක පද්ධතියක් යනුවෙන් අදහස් වන්නේ මෙලෙස නිසි සම්බන්ධතාවයක් සහිත කාරක ගණනාවක එකතුවකි.

අන්තර්ජාලය පාදකව ක්‍රියාත්මක වන booking.com වැනි හෝටල් වෙන්කර ගැනීමේ පද්ධතියක් (online booking systems) සලකන්න. හෝටලයක් වෙන්කර ගැනීමේ කාර්යයෙහි දී කාර්යයන්

ගණනාවක් සිදු කිරීමට වේ. අදාල දිනයන් වලට තමාගේ අවශ්‍යතාව අනුව හෝටලයක් සොයා ගැනීම, එහි කාමර ගණන තේරීම, මිල ගණන් සැසඳීම , ඇති ස්ථානය සිතියම් හරහා විමසා බැලීම මෙන්ම මුදල් ගෙවීම වැනි කාර්යයන් රාශියක් එක් පද්ධතියක් තුළ දැකිය හැකිය. මේවා අතර මනා සම්බන්ධයක් තිබිය යුතු අතර මෙවැනි පද්ධතියක මෘදුකාංග කාරක රාශියක් එකිනෙකට සම්බන්ධිතව ක්‍රියාත්මක වේ.



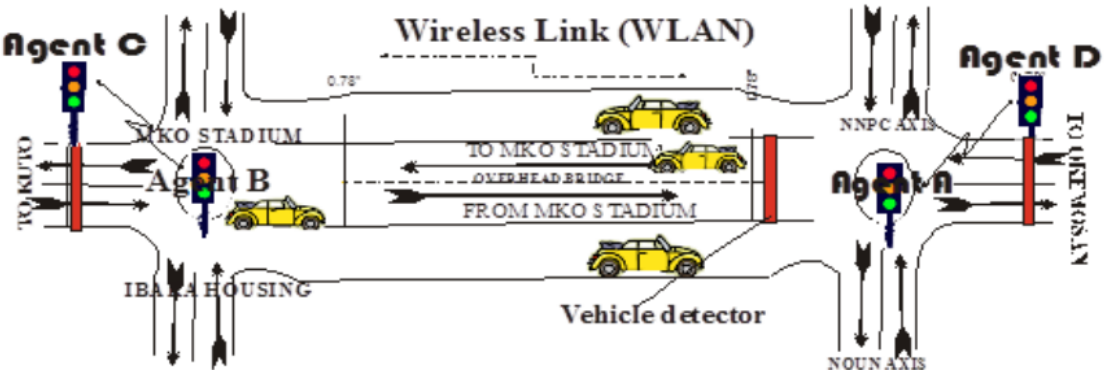
පහත දැක්වෙන්නේ බහුකාරක පද්ධතියක සැකසුමක් දැක්වෙන සටහනකි.



ස්වභාවය අනුව කාරක තාක්ෂණයන් කිහිපයක් දැකිය හැකිය.

- **තොරතුරු කාරක** - සංකීර්ණ පද්ධති වල තොරතුරු එක් රැස් කිරීම, ගවේශනය, සැසැඳීම මෙන්ම නිර්දේශයන් ලබා දීම සඳහා තොරතුරු කාරක භාවිත වේ.
- **පුරෝකථන කාර්යයන් සඳහා භාවිත වන කාරක-** තීරණ ගැනීම සඳහා අවශ්‍ය වන සාධක ලබා දෙමින් අනාවැකි පළකිරීමේ කාර්යය සඳහා උදව් වේ.
- **ජාලකළමනාකරණ කාරක-** අන්තර්ජාලය තුළ දී දත්ත හුවමාරුව හා පාලනය සඳහා ජාලකළමනාකරණ කාරක භාවිත කළ හැකි ය.
- **මහාමාර්ග වල තදබද පාලනය සඳහා භාවිත වන කාරක-** අධික තදබදයක් පවතින ජනාකීර්ණ නගර වල එම තදබදය පාලනය සඳහා කාරක තාක්ෂණයන් භාවිත කරනු ලැබේ.

Design of an Agent-Based Traffic Control System



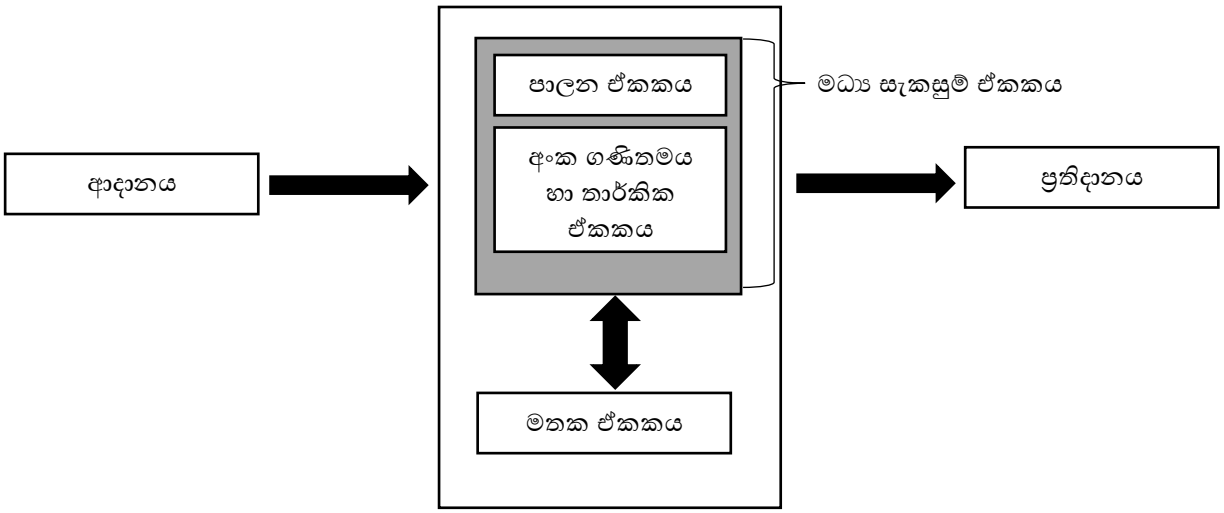
වාහන තදබදය පාලනය සඳහා සකසන ලද බහුකාරක නැතහොත් මෘදුකාංග ඒජන්ත පද්ධතියක සැලසුමක් - https://www.researchgate.net/publication/337150041_Design_of_an_Agent-Based_Traffic_Control_System

Web Link

වැඩිදුර කියවීම සඳහා මූලාශ්‍ර ලෙස භාවිත කරන්න - *https://whatis.techtarget.com/definition/software-agent*

මොනවද මේ වොන් නියුමාන් ආකෘතිය මත පදනම් වූ පරිගණක ?

පරිගණකයට උපදෙස් සැපයීම සඳහා අවශ්‍ය උපදෙස් පරිගණකය තුළම ගබඩා කොට අවශ්‍ය විටදි ඒවා භාවිතයට ගත හැකි බව පෙන්නුම් කරන ආචිත ක්‍රමලේඛ (Stored program)සංකල්පය මුලින්ම යෝජනා කරන ලද්දේ ජෝන් වොන් නියුමාන් (John Von Neumann) විසිනි. ඔහු විසින් සැලසුම් කරන ලද පරිගණකයේ පාලන ඒකකය (Control unit) අංක ගණිතමය හා තාර්කික ඒකකය(ALU) / මතක ඒකකය (Memory unit)/ රෙජිස්ටර (Register) හා ආදාන ප්‍රතිදාන ඒකක (Input output system) වලින් සමන්විත වේ. මෙම ආකෘතිය මත පදනම් වූ පරිගණ වොන් නියුමාන් ආකෘතිය මත පදනම් වූ පරිගණක නම් වේ. අප භාවිත කරන පරිගණක නිර්මාණය වී ඇත්තේ මෙම ආකෘතිය මතය.



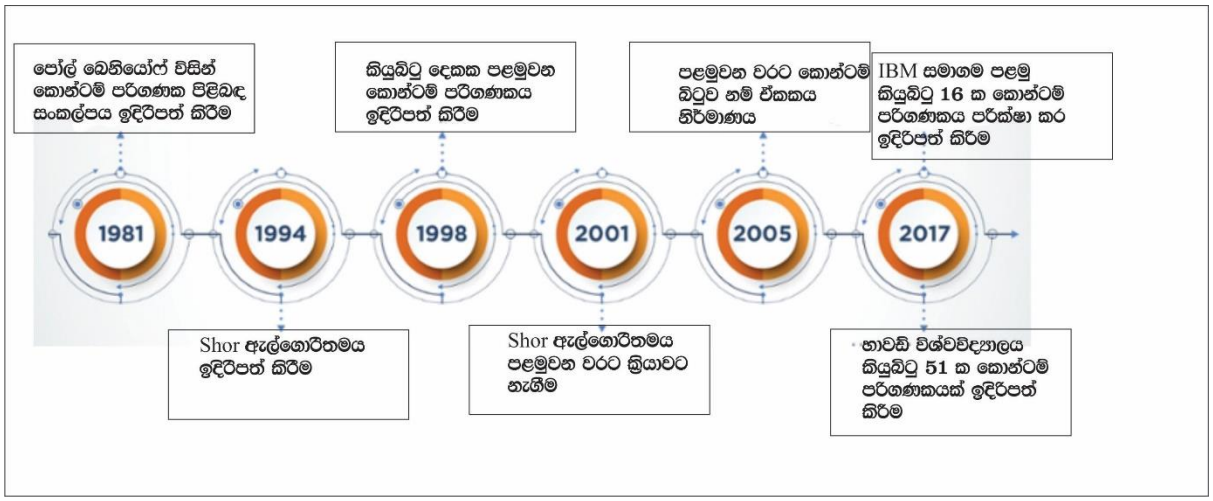
පරිගණක සම්පත් වේගයෙන් වර්ධනය වීම සිදුවන බැවින් වොන් නියුමාන් ආකෘතිය මත පදනම් වන පරිගණක මත පමණක් පදනම් වීම දැන් අපහසු කරුණක් ව තිබේ. මුරේ රීතිය (Moore's Law) අනුව

ක්ෂුද්‍ර සකසන වල වේගය සෑම මාස 18 කට වරක්ම දෙගුණවේ. මේ අනුව සිලිකන් පතුරක් භාවිතයෙන් නිර්මාණය කරනු ලබන ක්ෂුද්‍ර සකසන වේගවත් කිරීමට යාමේ දී ඒවායේ අධික තාපයක් ඇති වීම මෙන්ම තවදුරටත් කුඩා කිරීමේ සීමා සහිත බව නිසා නව තාක්ෂණ පිළිබඳව සිතා බැලීමට විද්‍යාඥයන්ට සිදුවී ඇත.

කොන්ටම් පරිගණක (Quantum Computing) හා එහි භාවිතය

අප එදිනෙදා භාවිත කරන ඩිජිටල් පරිගණක වල භාවිත වන්නේ අංකිත සංඛ්‍යාක පද්ධතිය වන අතර මේ අනුව 1 හෝ 0 මගින් විද්‍යුත් ස්පන්ධන නිරූපණය කරනු ලැබේ. වරකට 1 හෝ 0 පමණක් නිරූපණය කළ හැකි පරිගණක පද්ධති වල දියුණු කිරීමේ යම් සීමාවක් දැකිය හැකිය. මෙයට එක් විසඳුමක් ලෙස කොන්ටම් පරිගණක නම් සංකල්පය ඉදිරිපත් වී තිබේ. මෙහිදී පරමාණුක යොදා ගනිමින් පරිගණකයෙහි මතකය හා සකසනය නිර්මාණය කර ගෙන තිබේ. කොන්ටම් පරිගණක වල 1 හා 0 එකවර නිරූපණය කළ හැකි බැවින් සංකීර්ණ කාර්යයන් වල දී එම කාර්යයන් කාර්යක්ෂමව කර ගැනීම සඳහා මෙම පරිගණක යොදා ගත හැකිව තිබේ. සාමාන්‍ය ඩිජිටල් පරිගණක වල බිටු වක් (1 හෝ 0 මගින් නිරූපණය වේ) මගින් දත්ත පරිගණයෙහි තැන්පත් වුවද එකවර 1 හා 0 යන දෙකම නිරූපණය කළ හැකි කොන්ටම් පරිගණක වල භාවිත වන්නේ කියුබිටය නම් අගයයි.

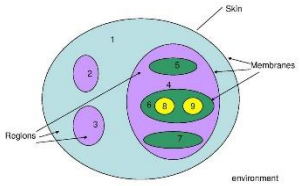
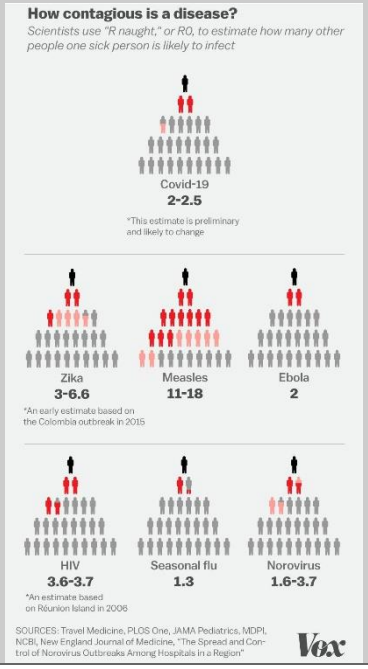
කොන්ටම් පරිගණකයේ ගමන්මඟ



සොබාදහමෙන් පෙළඹවෙන පරිගණකකරණය (Natural Inspired Computing)

සොබාදහමෙන් පෙළඹවෙන පරිගණනය නැතහොත් ප්‍රකෘති අනුප්‍රේරිත පරිගණනය යනු විවිධ පාරිසරික තත්ත්වයන් තුළ ඇති වන සංකීර්ණ ගැටලු විසඳීම සඳහා ස්වාභාවිකව සිදුවන සංසිද්ධි හැසිරෙන ආකාරය නිරීක්ෂණය කිරීම තුළින් නව පරිගණක ක්‍රමවේදයන් දියුණු කිරීමට උත්සාහ කරන තාක්ෂණයකි. මෙම පරිගණනය ස්වාභාවික පරිගණන තාක්ෂණයක් ලෙස ද හැඳින්වේ.

උදාහරණයක් ලෙස කුහුඹුවන්ගේ ජනපද, මී මැසි හා කුරුල්ලන් රංචු වැනි ජීව විද්‍යාත්මක කණ්ඩායම් උත්තේජකවලට ප්‍රතික්‍රියා කරන්නේ කෙසේද, තොරතුරු සැකසීම සහ තීරණ ගන්නේ කෙසේද යන්න පිළිබඳව අධ්‍යයන කරමින් ඒවා ඇසුරින් එදිනෙදා ගැටලු වලට විසඳුම් සෙවීම මින් සිදුවේ.මෙම ප්‍රකෘති අනුප්‍රේරිත පරිගණනය තුළ විවිධ ශිල්ප ක්‍රම මෙන්ම ප්‍රවේනික ඇල්ගොරිතම (ප්‍රධාන ඇල්ගොරිතම තුනක් මෙහි දී භාවිත වන අතර ඒවා Physics-Based Algorithms, Biology-Based Algorithms සහ chemistry-based algorithms ලෙස වර්ග කළ හැකිය) හා පද්ධති (Neural Network) ද භාවිත වේ. දැන් අපි මෙම පරිගණන ක්‍රමය තුළ භාවිත වන ශිල්ප ක්‍රම කිහිපයක් සැකෙවින් හඳුනා ගනිමු.

ශිල්ප ක්‍රමය	විස්තරය
සමූහ බුද්ධිය(Swarm intelligence)	කුහුඹු ජනපද , මී වද හා කුරුළු රංචු වැනි සමූහ වල ක්‍රියාකාරීත්වය හඳුනා ගනිමින් ඒ ඇසුරින් පද්ධති නිර්මාණය කර ගැනීම මෙහි දී සිදු වේ. මෙවැනි ඒකකයක් තුළ තනිතනිව එම සතුන් කටයුතු කරන ආකාරය මෙන්ම සමූහයක් ලෙස ක්‍රියාකරන ආකාරය ද අධ්‍යයනය කරනු ලැබේ. මේ අනුව තනි තනි ඒකක එකතු කොට සමූහ බුද්ධි රටාවන් ගොඩ නැගීම මගින් සංකීර්ණ පද්ධති ගොඩනැගීමට ආදර්ශ ලබා ගැනීම සිදුවේ.
පටල පරිගණනය (Membrane computing)	 (MC) ලෙස හැඳින්විය හැකිය.	ජීව විද්‍යාත්මක සෛල, විශේෂයෙන් සෛලීය පටල අධ්‍යයනය කිරීමෙන් නව පරිගණක ආකෘති සොයා ගැනීමට පරිගණක විද්‍යාව තුළ යොදා ගැනෙන ශිල්ප ක්‍රමය පටල පරිගණකකරණය
සන්නිවේදන පද්ධති හා නියමාවලි(Communication Systems and Protocols)		කොවිඩ් වැනි වෛරසයන් පැතිරෙන ආකාරය සලකන්න. එක් පුද්ගලයෙක් ගෙන් තවත් පුද්ගලයන් රාශියකට එය ව්‍යාප්ත වන අතර එම පුද්ගලයන්ගෙන් තවත් පුද්ගලයන් රාශියකට ලෙස ශීඝ්‍ර ව්‍යාප්තියක් මේවා පෙන්වයි. එලෙස ව්‍යාප්ත වන රටා අධ්‍යයනය කොට එම ආකාරයෙන් නවීන සන්නිවේද පද්ධති හා නියමාලි දියුණු කිරීමට අදහස් ලබා ගැනීම සඳහා මෙම ක්‍රමවේය භාවිත කෙරේ.

සොබාදහමෙන් ආභාසවෙන පරිගණකකරණය (nature inspired computing) පිළිබඳ පහත දැක්වෙන වගන්ති සලකන්න:

A – සොබාදහමෙන් ආභාසවෙන පරිගණකකරණයේ දී සංකීර්ණ ගැටළු විසඳීම උදෙසා පරිගණක ආකෘතියක් සැලසුම් කර සංවර්ධනය කිරීමට, ස්වාභාවික ප්‍රපංචයක්/සංසිද්ධියක් (phenomena/scenario) නිරීක්ෂණය කර යොදා ගනු ලැබේ.

B – පරිගණනීයව අභියෝගාත්මක ගැටළු අවබෝධ කර ගැනීමේ සහ විසඳීමේ හැකියාව වැඩි දියුණු කිරීම සඳහා කෘත්‍රිම බුද්ධිය (Artificial Intelligence) මගින් සොබාදහමින් ආභාසවෙන පරිගණකකරණය භාවිත කළ හැක.

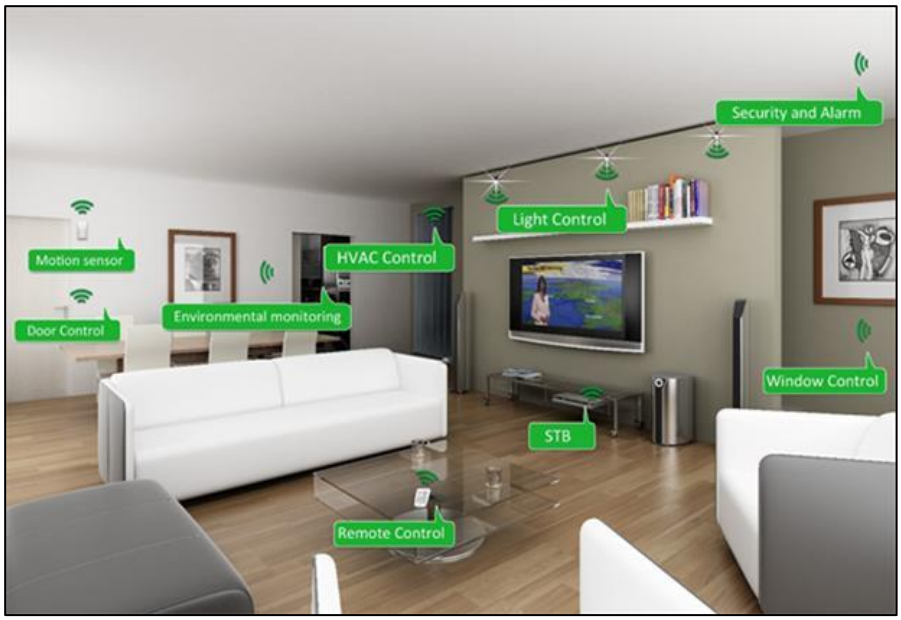
C – සොබාදහමින් ආභාසවෙන පරිගණකකරණය යටතෙහි සංවර්ධනය කරන ලද පරිගණක ආකෘති සහ ඇල්ගොරිතම, වැනි වනාන්තර, සාගර සහ වන සත්ත්ව අභයභූමි වැනි ස්වාභාවික පරිසර සඳහා ආභාසක් භාවිත කළ හැක.

ඉහත සඳහන් කවර වගන්තියක්/වගන්ති **සාවද්‍ය** වේ ද?

(1) A පමණි (2) B පමණි (3) C පමණි
(4) A සහ C පමණි (5) A, B සහ C සියල්ල ම

සාර්වර්ත්‍රික පරිගණකය ලෙස හැඳින්වෙන්නේ කුමක්ද?

වර්තමානයේ දී පරිගණක සෑම තැනකම දැකිය හැකි අතර අප ජංගම පරිගණක භාවිතයට ද බෙහෙවින්ම හුරුවී සිටිමු. ජංගම දුරකථනයක් හෝ ටැබ්ලට් පරිගණකයක් භාවිත කරමින් එදිනෙදා මගතොටේ දී හෝ වාහනයක ගමන් ගන්නා අතර දී අන්තර්ජාලය පරිශීලනය කිරීමට ද අප පෙළඹී සිටිමු. එහෙත් සාර්වර්තික පරිගණකය (Ubiquitous Computing හෝ "ubicomp") යනු එලෙස ජංගමව පරිගණක භාවිත කිරීම ලෙස කෙනෙකු වරදවා වටහා ගත හැකිය. සාර්වර්තික පරිගණනය යනු සෑම ස්ථානයකම , සෑම අවස්ථාවක දීම පරිගණක තාක්ෂණික උපාංග භාවිත කිරීමට අවස්ථාව සැලසීමයි. නිවසක් තුළ හෝ ඉන්පිටත සෑම තැනකම පරිගණක ගත උපාංග භාවිතයෙන් සේවා ලබා ගැනීමට මෙහි දී පරිශීලකයන් හට අවස්ථාව හිමිවේ.



ඉහත රූපය අනුව (මූලාශ්‍රය - https://www.monitis.com/blog/top-7-things-you-need-to-know-about-ubiquitous-computing/) නිවසක කාමරයක් තුළ විවිධ පරිගණකගත උපාංග භාවිත වන ආකාරය දක්වා ඇත. සාර්වර්ත්‍රික පරිගණනයේ දී සිදුවන්නේ මෙලෙස ඕනෑම ස්ථානයක සිට පරිගණකගත උපාංග භාවිතයෙන් කටයුතු පහසුකර ගැනීමට අවස්ථාව සැලසීම ලෙස දැක්විය හැකිය.