Grade 13 A/L ICT · Lesson 9

9.7 Python Basics: Structure, Variables, Data Types, Operators and I/O

Syllabus 9.7. Here we use an imperative language – Python – to encode algorithms. You meet the **structure** of a program, **comments**, **constants and variables**, the **primitive data types**, **arithmetic, relational, logical and bitwise operators**, **operator precedence**, keyboard **input** and screen **output**.

1. Structure, comments, constants and variables

🐍 Constants, variables, types and input

```
RATE = 0.08                      # constant (by convention UPPER CASE)
name = input("Name: ")           # str
qty = int(input("Quantity: "))   # int
price = 250.0                    # float
paid = True                      # bool
total = qty * price * (1 + RATE)
print(name, "pays Rs.", round(total, 2))
print(type(qty).__name__, type(price).__name__, type(paid).__name__)
```

▶ Output

```
Name: Nimal
Quantity: 4
Nimal pays Rs. 1080.0
int float bool
```

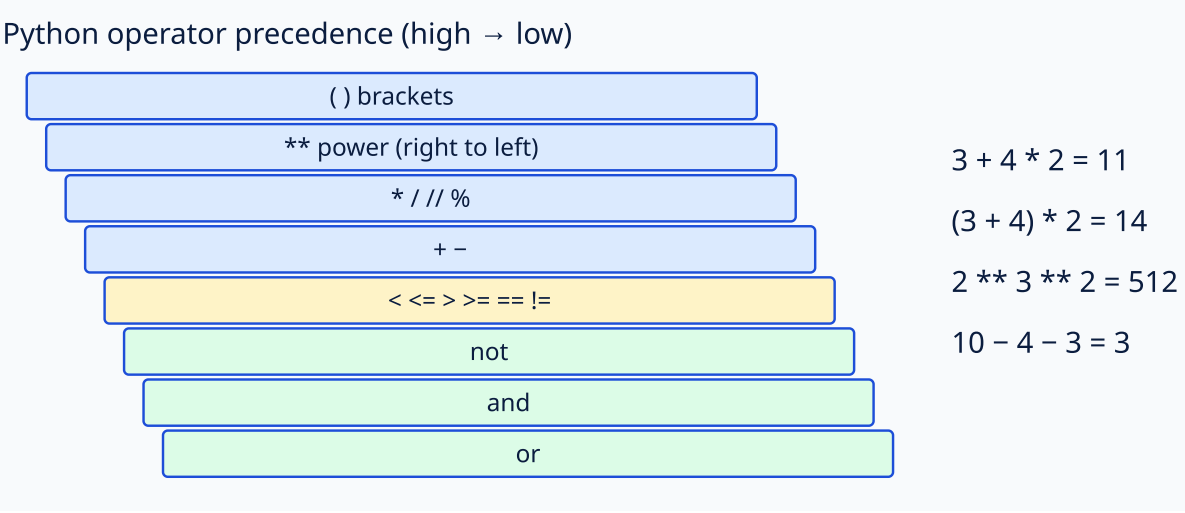
Type	Example	Meaning
int	4	Whole number
float	250.0	Real (decimal) number
str	"Nimal"	Text
bool	True / False	Truth value

A **comment** starts with # and is ignored by the translator. A **variable** is a named memory location whose value can change; a **constant** (written in UPPER CASE by convention) should not change. input() always returns a **str**, so int() or float() converts it.

> 🧮 **Worked example: Nimal buys 4 items at Rs. 250.0 with 8% tax. What is printed?**
> 4 × 250.0 = 1000.0; × 1.08 = **1080.0**, so it prints “Nimal pays Rs. 1080.0”. type() shows int, float, bool.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

2. Operators and precedence



Operator precedence

Category	Operators	Example → result
Arithmetic	+ - * / // % **	7 // 2 → 3, 7 % 2 → 1, 7 / 2 → 3.5
Relational	== != < <= > >=	5 > 3 → True
Logical	and or not	5 > 3 and 2 > 8 → False
Bitwise	& | ^ ~ << >>	5 & 3 → 1, 5 | 3 → 7, 5 ^ 3 → 6, 5 << 1 → 10

🐍 Operators and precedence

```
print(7 // 2, 7 % 2, 7 / 2)     # arithmetic
print(3 + 4 * 2, (3 + 4) * 2)  # precedence
print(2 ** 3 ** 2)             # ** is right to left
print(10 - 4 - 3)              # - is left to right
print(5 > 3 and 2 > 8)         # relational + logical
print(not True or True)
print(5 & 3, 5 | 3, 5 ^ 3, 5 << 1)   # bitwise
```

▶ Output

```
3 1 3.5
11 14
512
3
False
True
1 7 6 10
```

🧱 **Worked example: Explain 2 ** 3 ** 2 and 5 ^ 3.**

groups right to left**: 3** 2 = 9, then 2 **9 =** 512**. 5 = 101 and 3 = 011 in binary; XOR gives 110 =** 6**.

💡 In Python ^ is bitwise XOR, not power – use ** for powers.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

🚀 **Beyond the syllabus (2026 update)** *Not required for the exam – read for interest.* Python 3.12+ in 2026 gives clearer error messages and optional type hints (qty: int) that IDEs check before running.

Key Words

Comment – Note in code starting with # that is not executed.

Variable – Named storage location whose value can change.

Constant – Named value that should not change while running.

Primitive data type – Basic built-in type such as int, float, str or bool.

Integer – Whole number type int.

Float – Number type with a decimal part.

Boolean – Type holding True or False.

Arithmetic operator – Operator such as + or // that calculates numbers.

Relational operator – Operator that compares values and gives True or False.

Logical operator – and, or, not combining conditions.

Bitwise operator – Operator working on individual bits such as & or ^.

Operator precedence – Order in which operators are evaluated.

Sample Q&A

1. What does input() return?

> **Answer:** A string (str).

2. 7 // 2 and 7 % 2?

> **Answer:** 3 and 1.

3. 3 + 4 * 2?

> **Answer:** 11, because * is done before +.

4. Value of 10 - 4 - 3?

> **Answer:** 3 (left to right).

5. Four primitive types?

> **Answer:** int, float, str, bool.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

6. Constant vs variable?

> **Answer:** A constant should not change; a variable can.

MCQ

1. 7 // 2 in Python gives

A. 1
B. 3.5
C. 4
D. 3
E. 2

2. 7 % 2 in Python gives

A. 1
B. 0
C. 2
D. 3.5
E. 3

3. 3 + 4 * 2 gives

A. 14
B. 11
C. 9
D. 10
E. 24

4. 2 3 2 gives

A. 64
B. 12
C. 512
D. 256
E. 36

5. 5 ^ 3 in Python gives

A. 6
B. 2
C. 8
D. 125
E. 15

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

6. not True or True gives

A. 0
B. error
C. False
D. True
E. None

7. The type of the value returned by input() is

A. bool
B. float
C. int
D. list
E. str

8. Nimal’s total with 8% tax on 4 × 250.0 is

A. 1008.0
B. 1000.0
C. 1080.0
D. 1250.0
E. 1800.0

9. Which term means: “Note in code starting with # that is not executed”?

A. Bitwise operator
B. Comment
C. Arithmetic operator
D. Float
E. Constant

10. Which term means: “Named storage location whose value can change”?

A. Operator precedence
B. Relational operator
C. Variable
D. Primitive data type
E. Boolean

11. Which term means: “Named value that should not change while running”?

A. Integer
B. Logical operator
C. Arithmetic operator
D. Constant
E. Comment

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

12. Which term means: “Basic built-in type such as int, float, str or bool”?

A. Bitwise operator
B. Variable
C. Relational operator
D. Float
E. Primitive data type

13. Which term means: “Whole number type int”?

A. Logical operator
B. Integer
C. Operator precedence
D. Boolean
E. Constant

14. Which term means: “Number type with a decimal part”?

A. Bitwise operator
B. Comment
C. Primitive data type
D. Arithmetic operator
E. Float

15. Which term means: “Type holding True or False”?

A. Operator precedence
B. Boolean
C. Relational operator
D. Integer
E. Variable

16. Which term means: “Operator such as + or // that calculates numbers”?

A. Arithmetic operator
B. Constant
C. Logical operator
D. Comment
E. Float

17. Which term means: “Operator that compares values and gives True or False”?

A. Bitwise operator
B. Boolean
C. Primitive data type
D. Relational operator
E. Variable

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

18. Which term means: “and, or, not combining conditions”?

A. Logical operator
B. Constant
C. Integer
D. Operator precedence
E. Arithmetic operator

19. Which term means: “Operator working on individual bits such as & or ^”?

A. Comment
B. Float
C. Bitwise operator
D. Relational operator
E. Primitive data type

20. Which term means: “Order in which operators are evaluated”?

A. Logical operator
B. Integer
C. Variable
D. Boolean
E. Operator precedence

Structured Questions

1. Name four primitive data types with examples. *(4 marks)*

> int 4; float 250.0; str "Nimal"; bool True.

2. Explain variable, constant and comment with code. *(3 marks)*

> qty = 4 variable; RATE = 0.08 constant; # comment ignored.

3. Evaluate 3 + 4 * 2, (3 + 4) * 2 and 2 ** 3 ** 2 and explain. *(3 marks)*

> 11 (* first); 14 (brackets); 512 (** right to left).

4. Give one example each of relational, logical and bitwise operators with results. *(3 marks)*

> 5 > 3 True; 5 > 3 and 2 > 8 False; 5 & 3 = 1.

5. Write a program that inputs quantity and prints the total with 8% tax. *(3 marks)*

> int(input()); total = qty * price * 1.08; print round.

Summary

- Comments start with #.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

- int, float, str and bool are primitive types.
- input() returns a string – convert it.
- ** before * / // % before + −.
- ^ is XOR; ** is power.

MCQ answer key – 9.7

1. **D** – Floor division.
2. **A** – Remainder.
3. **B** – * first.
4. **C** – Right to left.
5. **A** – XOR.
6. **D** – not first.
7. **E** – Always str.
8. **C** – 1000 × 1.08.
9. **B** – Comment – Note in code starting with # that is not executed.
10. **C** – Variable – Named storage location whose value can change.
11. **D** – Constant – Named value that should not change while running.
12. **E** – Primitive data type – Basic built-in type such as int, float, str or bool.
13. **B** – Integer – Whole number type int.
14. **E** – Float – Number type with a decimal part.
15. **B** – Boolean – Type holding True or False.
16. **A** – Arithmetic operator – Operator such as + or // that calculates numbers.
17. **D** – Relational operator – Operator that compares values and gives True or False.
18. **A** – Logical operator – and, or, not combining conditions.
19. **C** – Bitwise operator – Operator working on individual bits such as & or ^.
20. **E** – Operator precedence – Order in which operators are evaluated.

© 2026 schoolict.net (Ruwan Mangala Suraweera). All rights reserved.