

---

# Python Programming in 2025

## > Foundational Basics

---

**Total: 30** | **NAME:** \_\_\_\_\_ | **Date:** \_\_\_\_\_ | **Time: 45**

### INSTRUCTIONS:

- This exam covers **Chapter 01: Foundational Basics** of Python.
  - Total duration: **45 minutes**, total marks: **50**.
  - Attempt **all sections**. Each question carries its own mark.
  - Write **clean and readable code** with proper indentation.
  - You should use pen for writing, and a separate sheet if needed.
  - For programming questions, focus on **logic, syntax**, and **correct output**.
  - **Use of calculators or mobile phones is not allowed.**
  - **No unfair means** will be tolerated; all work must be your own.
  - Be honest, stay calm, and code with confidence.
-

---

## Q1: MCQS

1x10

1. Who **developed** python?
  - A. Dennis Ritchie
  - b. Brendan Erich
  - c. Guido van Rossum
  - d. James gosling
2. What does python use to **execute** code?
  - A. Compiler
  - b. Interpreter
  - c. Assembler
  - d. Translator
3. Which of the following is a **valid** python variable name?
  - A. 2value
  - b. Value\_1
  - c. for
  - d. My-name
4. What kind of values does the **+** operator work on in python?
  - A. Only numbers
  - b. Only strings
  - c. Both numbers and strings
  - d. None of these
5. What is the output **of 5 % 2**?
  - A. 2
  - b. 0

- 
- c. 1
  - d. 2.5

6. Which of the following is a **logical** operator?

- A. +
- b. \*
- c. Or
- d. ==

7. **If-else** statements are used for:

- a. Loops
- b. Memory allocation
- c. Decision making
- d. File reading

8. In `for i in range (5)`, the **range (5)** part is:

- a. A variable
- b. An iterable
- c. A condition
- d. A function call

9. The **variable names passed inside** a function definition are called:

- a. Parameters
- b. Calls
- c. Arguments
- d. Outputs

10. What does **'x' mode** do in `OPEN ('file.txt', 'x')`?

- A. Opens file in append mode
- b. Creates a new file & writes
- c. Reads a file
- d. Deletes a file

---

## Q2: SHARE YOUR CONCEPTS 2x4

1. What is the difference between `input()` and `print()` functions in Python?
2. Explain the logical operator ``and`` with an example.
3. Differentiate between **bugs** and **exceptions**. Which of these can be **handled** in Python, and **how**?
4. What is the difference between an **interpreter** and a **compiler**?

-----  
=====

## Q3: DEBUGGING 2.5 x 2

Answer Format:

LINE NUMBER: TYPE OF ERROR

OTHER LINE: ERROR TYPE

DEBUGGED CODE

### 1. Detect:

```
Name = input ("Enter your name", "Enter  
your second name"):  
  
print("fHello, {name}!") # Print name  
given in input
```

---

## 2. Detect:

```
def myFunc(file='', mode='', content)
    file = open(file, mode)
    if file.readable():
        return file.read()
    elif file.writable():
        return file.write(conten)
    else:
        print("I Don't know what to do")
        return ''
answer = myfunc("myfile.txt", 'r',
'Updated content')
print(answer)
```

-----  
=====

---

## Q4: PREDICT OUTPUT

$2x^2 + 3$

### 1. PREDICT:

```
total = 0
for i in range(1, 5):
    total += i
print(total)
```

### 2. PREDICT:

```
def test(x):
    if x % 3 == 0:
        return x // 3
    else:
        return x + 3
print(test(9))
print(test(8))
```

---

### 3. PREDICT:

```
for i in range(1, 4):  
    for j in range(i):  
        print(i, end=' ')  
    print()
```