



PYTHON - IMPORTANT TIPS



Date : _____

Page No. : _____

- These tips will help you write better code and avoid common mistakes.
- Follow these while writing programs for exams and projects.



1 Indentation is Important



- Python uses indentation to define blocks.
- Use 4 spaces for each indent.
- Be consistent.

Example:

```
if num > 0:
    print("Positive") ✓

if num > 0:
    print("Positive") ✗ (Indentation Error)
```

2 Case Sensitive

Aa

- Python is case sensitive.
- Variable "name" and "Name" are different.

Example:

```
Name = "Sunil"
name = "nehra"
print(Name) # Sunil
print(name) # nehra
```

3 Use Meaningful Names



- Use meaningful variable and function names.
- It makes code easy to read.

Example:

```
# Bad
x = 5

# Good
number_of_students = 5
```

4 Comment Your Code



- Comments explain what your code does.
- Use comments to make code easier to understand.

Example:

```
# This program adds two numbers
a = 10      # first number
b = 20      # second number
s = a + b   # add
print(s)    # output sum
```

5 Use Proper Input



- Always take input with a prompt.
- It makes your program user-friendly.

Example:

```
name = input("Enter your name: ")
age = int(input("Enter your age: "))
```

6 Type Casting is Important



- Convert input to the required data type.
- Use int(), float(), str() etc.

Example:

```
n = int(input("Enter a number: "))
p = float(input("Enter price: "))
s = str(100)
```

7 Check for Errors



- Read error messages carefully.
- They help you find and fix the problem.

Example:

```
print(10/0)
# ZeroDivisionError: division by zero

print(x)
# NameError: name 'x' is not defined
```

8 Use Built-in Functions



- Python has many built-in functions.
- Use them to save time and code.

Example:

```
len("hello")      # 5
max(10, 20, 30)   # 30
sorted([3, 1, 2]) # [1, 2, 3]
```

9 Keep It Simple



- Write simple and easy to understand code.
- Avoid very long lines.
- Break the logic into small steps.

Example:

```
# Bad
print((a+b)*(c-d)/(e+f))

# Good
x = a + b
y = c - d
z = e + f
result = x * y / z
print(result)
```

10 Use Loops Wisely



- Use for loop when you know the range.
- Use while loop when condition is needed.

Example:

```
# for loop
for i in range(1, 6):
    print(i)

# while loop
i = 1
while i <= 5:
    print(i)
    i += 1
```



11 Test With Different Inputs



- Always test your program with different inputs.
- Helps to find logical errors.

Example:

```
Try inputs:
n = 5
n = -3
n = 0
n = 100
```

12 Save and Run Regularly



- Save your work frequently.
- Run your program regularly to check the output.

Example:

```
# Save: File -> Save As
# Run: press F5 or Run button
```

Do's

- ✓ Do understand the problem first.
- ✓ Do plan your logic before coding.
- ✓ Do use meaningful variable names.
- ✓ Do use comments.
- ✓ Do test your code.
- ✓ Do keep practicing!



Don'ts

- ✗ Don't copy code without understanding.
- ✗ Don't ignore error messages.
- ✗ Don't hard code values.
- ✗ Don't make your code very complex.
- ✗ Don't forget to check edge cases.



Practice More at
www.codestepacademy.in
for FREE Notes & Worksheets



Scan to Visit
Website





PYTHON PROGRAMS - EXAMPLES

Date : _____

Page No. : _____

- These programs are simple and useful for practice and exams.
- Read the code carefully and try running them yourself.



1 Print "Hello, World!"

A simple program to print text.

```
print("Hello, World!")
```

Output:

Hello, World!

2 Addition of Two Numbers

Takes two numbers and prints their sum.

```
a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
s = a + b
print("Sum =", s)
```

Sample Input:

Enter first number: 12
Enter second number: 25

Output:

Sum = 37

3 Find Area of Circle

Calculates area of a circle.

```
import math
r = float(input("Enter radius: "))
area = math.pi * r * r
print("Area =", area)
```

Sample Input:

Enter radius: 7

Output:

Area = 153.938...

4 Check Even or Odd

Checks whether a number is even or odd.

```
n = int(input("Enter a number: "))
if n % 2 == 0:
    print("Even")
else:
    print("Odd")
```

Sample Input:

Enter a number: 9

Output:

Odd

5 Factorial of a Number

Finds factorial of a positive number.

```
n = int(input("Enter a number: "))
fact = 1
for i in range(1, n+1):
    fact *= i
print("Factorial =", fact)
```

Sample Input:

Enter a number: 5

Output:

Factorial = 120

6 Generate Fibonacci Series

Prints first n terms of Fibonacci series.

```
n = int(input("How many terms? "))
a, b = 0, 1
print(a, end=' ')
for i in range(1, n):
    print(b, end=' ')
    a, b = b, a + b
```

Sample Input:

How many terms? 7

Output:

0 1 1 2 3 5 8

7 Table of a Number

Prints multiplication table of a number.

```
n = int(input("Enter a number: "))
for i in range(1, 11):
    print(n, "x", i, "=", n*i)
```

Sample Input:

Enter a number: 5

Output:

5 x 1 = 5
5 x 2 = 10
.
5 x 10 = 50

8 Sum of Natural Numbers

Finds sum of first n natural numbers.

```
n = int(input("Enter n: "))
s = n * (n + 1) // 2
print("Sum =", s)
```

Sample Input:

Enter n: 10

Output:

Sum = 55

9 Check Prime Number

Checks whether a number is prime.

```
n = int(input("Enter a number: "))
if n > 1:
    for i in range(2, int(n**0.5)+1):
        if n % i == 0:
            print("Not Prime")
            break
    else:
        print("Prime")
else:
    print("Not Prime")
```

Sample Input:

Enter a number: 17

Output:

Prime

10 Reverse a Number

Reverses the digits of a number.

```
n = int(input("Enter a number: "))
rev = 0
while n > 0:
    rev = rev * 10 + n % 10
    n = n // 10
print("Reversed number =", rev)
```

Sample Input:

Enter a number: 1234

Output:

Reversed number = 4321

11 Count Vowels in a String

Counts number of vowels in a string.

```
s = input("Enter a string: ").lower()
vowels = "aeiou"
count = 0
for ch in s:
    if ch in vowels:
        count += 1
print("Vowels =", count)
```

Sample Input:

Enter a string: Hello World

Output:

Vowels = 3

12 Largest of Three Numbers

Finds the largest among three numbers.

```
a = float(input("Enter first number: "))
b = float(input("Enter second number: "))
c = float(input("Enter third number: "))
if a >= b and a >= c:
    print("Largest =", a)
elif b >= c:
    print("Largest =", b)
else:
    print("Largest =", c)
```

Sample Input:

Enter first number: 12
Enter second number: 45
Enter third number: 23

Output:

Largest = 45



Tips for Writing Programs

- ✓ Understand the problem clearly.
- ✓ Use meaningful variable names.
- ✓ Indent your code properly.
- ✓ Test your program with different inputs.
- ✓ Practice regularly!



More Practice Ideas

- ★ Find GCD or LCM of two numbers
- ★ Convert Celsius to Fahrenheit
- ★ Check Armstrong number
- ★ Find second largest number in a list
- ★ Shuffle elements of a list randomly

Practice More at

www.codestepacademy.in

for FREE Notes & Worksheets



Scan to Visit

Website →



PYTHON - IMPORTANT QUESTIONS FOR VIVA

- These questions are commonly asked in practical / viva exams.
- Answers are short, simple and to the point - Learn them well!

Date : _____

Page No. : _____

Q1: What is Python?

Ans: Python is a high-level, interpreted, general-purpose programming language.

Q2: Who developed Python and in which year was it first released?

Ans: Guido van Rossum developed Python and it was first released in 1991.

Q3: Is Python case sensitive?

Ans: Yes, Python is case sensitive.

Q4: What are the modes in Python?

Ans: Python has two modes - Interactive Mode and Script Mode.

Q5: What is the difference between Interactive Mode and Script Mode?

Ans: In Interactive Mode, we type commands one by one in the shell.
In Script Mode, we write the program in a file and run it.

Q6: How do you print output in Python?

Ans: We use the print() function.
Example: print("Hello")

Q7: What is a variable?

Ans: A variable is a name given to a memory location used to store data.

Q8: How do you assign a value to a variable?

Ans: We use the assignment operator (=).
Example: x = 10

Q9: What are the basic data types in Python?

Ans: int, float, str, bool.

Q10: What is the output of type(5.0)?

Ans: <class 'float'>

Q11: What is a comment in Python?

Ans: A comment is a line ignored by the interpreter.
Single line comment starts with #.

Q12: What is the use of input() function?

Ans: input() is used to accept data from the user.
Example: name = input("Enter your name: ")

Q13: How do you take integer input from the user?

Ans: Use int(input()).
Example: n = int(input("Enter a number: "))

Q14: What is the difference between / and // operators?

Ans: / gives floating point division (quotient).
// gives floor division (integer quotient).

Q15: What is the % operator used for?

Ans: It is used to get the remainder (modulus) after division.

Q16: What is the output of 5 % 2?

Ans: 1

Q17: What is a conditional statement?

Ans: A conditional statement is used to make decisions based on certain conditions. (if, if-else, if-elif-else)

Q18: What is the difference between break and continue?

Ans: break terminates the loop.
continue skips the current iteration and continues next.

Q19: What is a loop in Python?

Ans: A loop is used to repeat a block of code.
(for loop, while loop)

Q20: What does range() function do?

Ans: range() generates a sequence of numbers.
Example: range(5) → 0, 1, 2, 3, 4

Q21: What is a string in Python?

Ans: A string is a sequence of characters enclosed in single, double or triple quotes.

Q22: What is the use of len() function?

Ans: len() returns the length of a string, list, tuple, etc.

Q23: What is the difference between list and tuple?

Ans: List is mutable (can be changed), Tuple is immutable (cannot be changed).

Q24: How do you add an element to a list?

Ans: Use append() or insert() method.
Example: lst.append(10)

Q25: How do you remove an element from a list?

Ans: Use remove() or pop() method.
Example: lst.remove(10) or lst.pop()

Q26: What is a dictionary in Python?

Ans: A dictionary stores data in key-value pairs.

Q27: How do you access a value in a dictionary?

Ans: Using its key, e.g., d["name"]

Q28: What will happen if we access a key that does not exist?

Ans: It will give a KeyError.
To avoid this, use get() method.

Q29: What is a module in Python?

Ans: A module is a file containing Python definitions and statements. It helps in code reusability.

Q30: How do you import a module in Python?

Ans: Using import <module> or from <module> import <name>
Example: import math
from math import sqrt



TIPS FOR VIVA

- ✓ Answer confidently and in your own words.
- ✓ Speak clearly and to the point.
- ✓ If you don't know, be honest.
- ✓ Show your practical files neatly.
- ✓ Practice programs regularly.



QUICK FACTS

- ✓ Python is easy to learn.
- ✓ It is open source.
- ✓ It has a large standard library.
- ✓ It supports multiple platforms.
- ✓ It is widely used in industry.



Practice More at
www.codestepacademy.in
for FREE Notes & Worksheets

Scan to Visit
Website

