Unit 1

1 Mark Questions with Answers

Introduction to Python

1. **Q:** Who developed Python?
   A: Guido van Rossum.

2. **Q:** In which year was Python released?
   A: 1991.

3. **Q:** Is Python compiled or interpreted?
   A: Interpreted.

4. **Q:** Write one feature of Python.
   A: Simple and easy-to-learn syntax.

5. **Q:** Mention one popular application area of Python.
   A: Web development.

Applications of Python

6. **Q:** Which library is used for Machine Learning in Python?
   A: Scikit-learn / TensorFlow.

7. **Q:** Name one Python framework for web development.
   A: Django or Flask.

8. **Q:** Which library is used for data analysis?
   A: Pandas.

9. **Q:** Name one GUI toolkit in Python.
   A: Tkinter.

10. **Q:** Which Python library is used for scientific computing?
    A: NumPy.

Installation & IDEs

11. **Q:** From which website can you download Python?
    A: www.python.org.

12. **Q:** What does IDE stand for?
    A: Integrated Development Environment.

13. **Q:** Write any two Python IDEs.
A: PyCharm, Jupyter Notebook.

14. **Q:** Which IDE comes by default with Python?
A: IDLE.

15. **Q:** Command to check Python version?
A: `python --version`.

Python Syntax, Indentation & Comments

16. **Q:** What is used to define code blocks in Python?
A: Indentation.

17. **Q:** Which symbol is used for single-line comments?
A: `#`.

18. **Q:** How do you write multi-line comments?
A: Using triple quotes `'''` or `"""`.

19. **Q:** Write a simple if statement in Python.
A:

```python
if True:
    print("Hello")
```

20. **Q:** What error occurs if indentation is missing?
A: IndentationError.

Variables, Data Types & Type Casting

21. **Q:** How do you create a variable in Python?
A: By assigning value directly, e.g., `x = 10`.

22. **Q:** Can a variable hold values of different types in Python?
A: Yes.

23. **Q:** Data type of `3.14` is?
A: float.

24. **Q:** Data type of `"Hello"` is?
A: string.

25. **Q:** What is the output of `float(5)`?
A: 5.0.

Operators

26. **Q:** Which operator is used for exponentiation?
   A: `**`.

27. **Q:** Difference between `==` and `is`?
   A: `==` checks values, `is` checks identity.

28. **Q:** Write one logical operator.
   A: `and`.

29. **Q:** Which operator checks membership?
   A: `in`.

30. **Q:** What does `&` do in Python?
   A: Bitwise AND.

Input/Output Functions

31. **Q:** Which function is used to take input from user?
   A: `input()`.

32. **Q:** Write a print statement.
   A: `print("Hello World")`.

33. **Q:** How to print multiple values in one line?
   A: Using commas, e.g., `print("A", "B")`.

34. **Q:** Which function converts input to integer?
   A: `int()`.

35. **Q:** What is the default separator in `print()`?
   A: Space.

Control Structures

36. **Q:** Write syntax of `if` statement.
   A:

```python
if condition:
    statement
```

37. **Q:** Which statement is used to check multiple conditions?
   A: if-elif-else.

38. **Q:** Example of nested if?
    A:

```python
if x > 0:
    if x % 2 == 0:
        print("Positive Even")
```

39. **Q:** Which statement executes if condition is false?
    A: else.

40. **Q:** Why is elif used?
    A: To check multiple conditions when `if` fails.

Looping

41. **Q:** Which loop is used to iterate over a sequence?
    A: for loop.

42. **Q:** Print numbers from 1 to 5 using for loop.
    A:

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

43. **Q:** Which keyword is used to exit a loop?
    A: break.

44. **Q:** Which keyword skips current iteration?
    A: continue.

45. **Q:** What does `pass` do?
    A: Does nothing (placeholder).

46. **Q:** Can loops be nested in Python?
    A: Yes.

47. **Q:** Print even numbers from 2 to 10 using while loop.
    A:

```
i = 2
while i <= 10:
    print(i)
    i += 2
```

48. **Q:** What happens if condition in while loop never becomes false?
   A: Infinite loop.

49. **Q:** Which statement is used to terminate loop immediately?
   A: break.

50. **Q:** What happens if we use else without if?
   A: SyntaxError.

2-Mark Questions with Answers

1. **List any two important features of Python.**
   - Easy to learn and use (simple syntax).
   - Interpreted and dynamically typed.
2. **Give two examples of real-world applications of Python.**
   - Web development (Django, Flask).
   - Data science and machine learning (NumPy, pandas, scikit-learn).
3. **Differentiate between compiler and interpreter in Python installation.**
   - Compiler translates the whole program at once, while **Python uses an interpreter** that executes line by line.
4. **What is the role of an IDE in Python programming?**
   - IDE provides an environment to **write, run, debug, and test Python code** easily (e.g., PyCharm, VS Code).
5. **Write two rules for Python syntax and indentation.**
   - Indentation is mandatory for blocks.
   - Statements do not end with semicolons by default.
6. **Why are comments used in Python? Give an example.**
   - Comments make code readable and explain its purpose.
   - Example: `# This is a single-line comment`
7. **Differentiate between mutable and immutable variables in Python with examples.**
   - Mutable: Can be changed (e.g., `list = [1,2,3]`).
   - Immutable: Cannot be changed (e.g., `tuple = (1,2,3)`).
8. **Write examples of two numeric data types in Python.**
   - Integer → `x = 10`   Float → `y = 3.14`

9. **What is type casting? Give one example of implicit and explicit type casting.**
    - Converting one data type to another.
    - Implicit: `x = 5; y = 2.5; z = x + y` → result is float.
    - Explicit: `int(3.14)` → `3`

10. **Write the difference between relational and logical operators in Python.**

- Relational: Compare values (e.g., `>`, `<`, `==`).
- Logical: Combine conditions (e.g., `and`, `or`, `not`).

11. **What is the use of the membership operator in Python? Give an example.**

- Checks if a value exists in a sequence.
- Example: `"a" in "apple"` → True.

12. **Write the difference between the identity operator `is` and equality operator `==`.**

- `==` checks values → `10 == 10` → True.
- `is` checks memory reference → `[1,2] is [1,2]` → False.

13. **Explain the role of bitwise operators with an example.**

- Perform operations on bits.
- Example: `5 & 3 = 1` (binary AND).

14. **What is the difference between input() and print() functions in Python?**

- `input()` → reads user input.
- `print()` → displays output.

15. **Write a program snippet using if-else to check if a number is even or odd.**

```
n = 4
if n % 2 == 0:
    print("Even")
else:
    print("Odd")
```

16. **What is the difference between if-elif-else and nested if statements?**

- `if-elif-else`: Multiple conditions in sequence.
- Nested if: One `if` inside another.

17. **Write a short program using a for loop to print the first 5 natural numbers.**

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

18. Write a while loop program to print numbers from 1 to 5.

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

19. **What is the use of the break statement in Python? Give an example.**

- Used to exit a loop immediately.

```
for i in range(10):
   if i == 5:
      break
   print(i)
```

20. Differentiate between continue and pass statements with examples.

- **`continue`: Skips current iteration.**

```
for i in range(5):

   if i == 2:

      continue

   print(i)
```

- **`pass`: Does nothing (placeholder).**

```
for i in range(5):

   if i == 2:

      pass

   print(i)
```

5-Mark Questions with Answers

1. Explain the main features of Python with examples.

Answer:

- **Simple & Easy** → Python has readable syntax.
- **Interpreted** → Executes line by line.
- **Portable** → Same code runs on Windows/Linux/Mac.
- **Dynamically Typed** → No need to declare data type.
- **Extensive Libraries** → Supports NumPy, Pandas, etc.

2. Write a short note on Python applications in real life.

Answer:
Python is widely used in:

- **Web Development –** Django, Flask.
- **Data Science & AI –** TensorFlow, scikit-learn.
- **Game Development –** Pygame.
- **Automation/Scripting –** Writing automation scripts.
- **Networking & Cybersecurity –** Packet analysis and security testing.

3. Compare compiler and interpreter. Why is Python called an interpreted language?

Answer:

- **Compiler** translates the whole program into machine code before execution.
- **Interpreter** translates line by line at runtime.
- **Python uses an interpreter** → It executes one line at a time, making debugging easier but execution slower compared to compiled languages like C++.

4. Explain Python variables and data types with examples.

Answer:

- **Variable:** A named storage for values.

  ```
  x = 10  # integer
  y = "Hello"  # string
  ```

- **Data Types in Python:**

i. Numeric (int, float, complex)

ii. Sequence (list, tuple, string)

iii. Set & Dictionary

iv. Boolean

5. Explain type casting in Python with examples.

Answer:

- **Type Casting** → Converting one data type into another.
- **Implicit Casting:** Python automatically converts types.

  ```
  x = 5
  ```

```
y = 2.5

z = x + y  # result = 7.5 (float)
```

- **Explicit Casting:** Using built-in functions.

```
num = int(3.7)  # result = 3

text = str(123)  # result = "123"
```

6. Explain control structures in Python with examples.

Answer:

- **if statement:**

```
x = 10

if x > 5:

    print("Greater")
```

- **if-else statement:**

```
x = 3

if x % 2 == 0:

    print("Even")

else:

    print("Odd")
```

- **if-elif-else:**

```
marks = 75

if marks >= 90: print("Grade A")

elif marks >= 60: print("Grade B")

else: print("Grade C")
```

7. Explain the use of loops in Python with examples of for and while loops.

Answer:

- **for loop:** Iterates over a sequence.

```
for i in range(1, 6):

    print(i)
```

- **while loop:** Runs until condition is false.

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

- **Nested loops:** Loop inside another loop.

8. Differentiate between break, continue, and pass statements with examples.

Answer:

- **break** → exits loop immediately.

```
for i in range(5):
    if i == 3: break
    print(i)
```

- **continue** → skips current iteration.

```
for i in range(5):
    if i == 2: continue
           print(i)
```

- **pass** → does nothing (placeholder).

```
for i in range(5):
    if i == 2: pass
         print(i)
```

9. Explain Python operators with examples.

Answer:

- **Arithmetic:** `+ - * / % // **`
- **Relational:** `> < >= <= == !=`
- **Logical:** `and, or, not`
- **Assignment:** `=, +=, -=`
- **Membership:** `in, not in`
- **Identity:** `is, is not`
- **Bitwise:** `&, |, ^, ~, <<, >>`

10. Write a program to print a multiplication table of a given number using loops.

```
num = 5

for i in range(1, 11):

    print(num, "x", i, "=", num * i)
```

More 5-Mark Questions:

Operators

11. With examples, illustrate the use of assignment and augmented assignment operators.
12. Explain membership operators (`in`, `not in`) with examples.
13. Differentiate between identity operators (`is`, `is not`) and relational operators with examples.
14. Explain logical operators in Python with examples.
15. Explain Python’s arithmetic operators with examples.
16. Discuss relational operators in Python with suitable programs.

Control Structures (Decision Making)

17. Write a Python program using “if statement” to check whether a number is positive.
18. Explain if-else with a program to find the largest of two numbers.
19. Write a program using if-elif-else to calculate grade of a student based on marks.
20. Write a program using nested conditions to check whether a year is a leap year or not.

Looping

21. Write a program to print the first 10 natural numbers using a **for loop**.
22. Write a program to calculate the factorial of a number using a **while loop**.
23. Write a program to print a multiplication table using **nested loops**.
24. What is the purpose of else clause for a loop? Explain how else works with for and while loops, with examples.
25. Write a program to print all even and odd numbers between 1 and 20 using loops.