TYPE-A:PYTHON FUNDAMENTALS:CH-6

1. What are tokens in Python? How many types of tokens are allowed in Python Examplify your answer.

sol:
Tokens are the smallest units of a Python program.

Python has five types of tokens:

1. Keywords – Example: if, else, for
2. Identifiers – Example: total, num1
3. Literals – Example: 10, "Hello", 3.5
4. Operators – Example: +, -, ==
5. Punctuators – Example: (,), :, ,

2. How are keywords different from identifiers ?

sol:

Keywords	**Identifiers**
Keywords are reserved words in Python.	Identifiers are names given to variables, functions, and objects.
They have special meaning.	They are used to identify program elements.
They cannot be used as names.	They can be used as names.
Examples: if, else, while	Examples: total, num1, sum

3. What are literals in Python ? How many types of literals are allowed in Python ?

sol:
Literals are fixed values used in a Python program.

Python has four types of literals:

1. Numeric literals – Example: 10, 3.5, 2+3j
2. String literals – Example: "Hello", 'Python'
3. Boolean literals – Example: True, False
4. Special literal – Example: None

4. Can nongraphic characters be used in Python ? How ? Give examples to support your answer.

sol:

- Yes, nongraphic characters can be used in Python using escape sequences.
- Escape sequences are special codes used to represent nongraphic characters.

Examples:

\n – New line
\t – Tab space
\ – Backslash

5. How are floating constants represented in Python ? Give examples to support your answer.

sol:
Floating constants in Python are used to represent decimal numbers. They can be written in
decimal form or exponential form.

Examples:

Decimal form:
3.14
25.5
0.75

Exponential form:
2.5e3
1.2e-2
5E4

6. How are string-literals represented and implemented in Python ?

sol:
String literals in Python are represented using single quotes, double quotes, or triple quotes.

Single quotes:
'Hello'

Double quotes:
"Python"

Triple quotes:
'''Welcome to Python'''
"""This is a multi-line string"""

Strings in Python are implemented as sequences of characters and are stored in memory as
objects.

7. Which of these is not a legal numeric type in Python ? (a) int (b) float (c) decimal.

sol:
(c) decimal is not a legal numeric type in Python.

8. Which argument of print( ) would you set for:

(i) changing the default separator (space) ?
(ii) printing the following line in current line ?

sol:

(i) sep
(ii) end

9.What are operators ? What is their function ? Give examples of some unary and binary operators.

sol:

Operators are special symbols used in Python to perform operations on values and variables.

Their function is to carry out calculations, comparisons, and logical operations.

Examples of Unary Operators:

- (Unary plus)
- (Unary minus)
- not

Examples of Binary Operators:

- (Addition)
- (Subtraction)
- (Multiplication)
- / (Division)
- == (Equal to)

10.What is an expression and a statement ?

sol:

An expression is a combination of values, variables, and operators. It always produces a result.

A statement is an instruction given to Python to perform an action. It does not always produce a result.

11.What all components can a Python program contain ?

sol:

A Python program can contain the following components:

- Keywords
- Identifiers
- Literals
- Operators
- Punctuators
- Expressions
- Statements
- Comment
- Functions

12. What do you understand by block/code block/suite in Python ?

sol:

A block, code block, or suite in Python is a group of related statements that are executed together. It is defined by proper indentation and is usually written under statements like if, for, while, and functions.

13. What is the role of indentation in Python ?

sol:

Indentation in Python is used to define blocks of code. It helps Python understand which statements belong together and controls the execution of programs.

14. What are variables ? How are they important for a program ?

sol:

Variables are names used to store data values in a program. They allow programs to save, change, and reuse information.

Variables are important because they make programs flexible and easy to understand.
They
help in performing calculations and controlling program flow.

15. What do you understand by undefined variable in Python ?

sol:

An undefined variable in Python is a variable that is used before it is given a value. When
this
happens, Python gives an error because it does not know what value the variable contains.

16. What is Dynamic Typing feature of Python ?

sol:

Dynamic typing in Python means that a variable does not need a fixed data type. The type
of a
variable is decided at runtime and can change when a new value is assigned.

17. What would the following code do : X = Y = 7 ?

sol:

The statement X = Y = 7 assigns the value 7 to both variables X and Y. After this, X and Y
will
store the same value.

18. What is the error in following code : X, Y = 7 ?

sol:

The statement X, Y = 7 causes an error because Python expects two values for X and Y, but only one value (7) is given.

So, this results in a ValueError due to incorrect unpacking.

19. Following variable definition is creating problem X = 0281, find reasons.

sol:

The variable definition X = 0281 creates a problem because numbers starting with 0 are treated as invalid in Python.

In Python, a number cannot start with 0 unless it is written in a special format. So, 0281 is not a valid number and causes a syntax error.

20. "Comments are useful and easy way to enhance readability and understandability of a program." Elaborate with examples.

sol:

Comments are statements written in a program to explain the code. They are not executed by Python.

Comments make a program easy to read and understand. They help programmers remember the purpose of the code and make it easier for others to understand it.

Comments are useful while finding errors and improving programs.

Examples:

Single-line comment:

```
# This program adds two numbers
a = 10
b = 20
print(a + b)
```

Multi-line comment:

```
"""
This program calculates
the area of a circle
"""
r = 5
area = 3.14 * r * r
print(area)
```