

## ✓ What is Python? Mention two of its key features.

Python is the high level interpreted programming language invented by Guido Van Rossum in 1991.

### Features-

1. Easy to learn and read
  2. Extensive rich library
  3. Free and open source
  4. Dynamically typed
- 

## How do you write a single-line comment in Python?

Single line comments in python are written inside the '#'.  

---

## How do you write a multi-line comment in Python?

Multiline comments in python are written inside `"""..."""` (single inverted commas)  

---

## What is the difference between print() and return?

print() is a function which takes objects as argument return does not use objects  

---

## How do you get user input in Python?

In Python user input is taken by using input() function  

---

## How do you check the version of Python installed?

To check the version of python, first we open command prompt then type `python --version` and press enter. It will show the version of Python installed in your PC  

---

## Is Python case-sensitive? Give an example.

Yes Python is case sensitive language. For example if we type `PRINT()` it will throw an error  

---

## How do you run a Python script from the command line?

To run a python from the command line we have to type a command

```
python file_name
```

---

## What are keywords in Python? How can you list them?

Keywords are the reserved words in python that you cannot use as variable names To list them

```
import keyword  
print(keyword.kwlist)
```

```
import keyword  
print(keyword.kwlist)
```

---

## How do you declare a variable in Python?

In python variable declaration is done simply by using = to assign a value to it example

```
name = "Shreya Maurya"  
age = 22
```

---

## What's the difference between = and ==?

in python, = is used for assigning the value and == is used for comparing values

---

## How do you swap two variables without using a third variable?

Swapping two values:

```
a=10  
b=20  
a= a+b #30  
b= a-b #10  
a= a-b #20  
print(a,b)
```

---

## How do you write a one-line if statement?

```
if condition: statement
```

## What's the difference between None and 0 in Python?

In python none returns a NULL value while 0 is a numeric value

---

## What is indentation in Python and why is it important?

Indentation in Python refers to the leading whitespace (spaces or tabs) at the beginning of a line of code. Statements with the same level of indentation are considered to be part of the same block, such as within if statements, for or while loops, function definitions, and class definitions. Incorrect indentation will lead to an IndentationError.

---

## What are Python's built-in data types?

Datatype defines the type of variable we are using in our code  
Numeric Types: Integers (int)  
Floating-point numbers (float) Complex numbers (complex) Boolean Type  
Sequence Types: Strings (str) Lists (list) Tuples (tuple) Range (range) Sets (set) Mapping Type: Dictionaries (dict)

---

## How do you check the type of a variable?

In Python, the type of a variable can be checked using the built-in type() function.

---

## What's the difference between a list and a tuple?

Lists are mutable: This means their elements can be changed, added, or removed after the list has been created.

Tuples are immutable: Once a tuple is created, its elements cannot be modified, added, or removed.

## How do you create a dictionary in Python?

1. Using Curly Braces {}
  2. Using the dict() Constructor
-

## What's the difference between `append()` and `extend()` for lists?

`append(element)`:

This method adds a single element to the end of the list

`extend(iterable)`:

This method adds all elements from an iterable (like another list, tuple, string, or set) to the end of the list.

## How do you remove an item from a list?

From a list an item can be removed using `remove()` function

## How do you reverse a list in Python?

```
my_list = [1, 2, 3, 4, 5]
my_list.reverse()
print(my_list)
```

## How do you sort a list in ascending order?

```
numbers = [5, 2, 9, 1]
numbers.sort()
```

## What is the difference between shallow copy and deep copy?

Shallow copy -

Copies the object but references nested objects. Changes in nested objects affect both copies.

Deep copy -

Copies the object and all nested objects recursively. Changes don't affect the original.

## How do you convert a string to lowercase?

```
text = "HELLO"  
lowercase = text.lower()
```

## How do you check if a string starts with a particular word?

```
text = "Python is great"  
text.startswith("Python")    # True
```

---

## What's the difference between is and ==?

== - Compares values.

is - Compares object identity (whether they point to the same object in memory)

## How do you merge two dictionaries in Python 3.9+?

```
dict1 = {"a": 1}  
dict2 = {"b": 2}  
merged = dict1 | dict2
```

---

## How do you find the length of a dictionary?

Length of the dictionary is measured using len() function

```
data = {"a": 1, "b": 2}  
len(data)    # 2
```

---

## How do you create a set?

1. using set() function
  2. using curly braces
- 

## What's the difference between set() and {} in Python?

set() - creates an empty set.

{ } - creates an empty dictionary.

## How do you find the union of two sets?

To find union of two sets we use

```
a.union(b)
```

---

## How do you find the intersection of two sets?

To find intersection of two sets we use

```
a.intersection(b)
```

---

## What's the difference between remove() and discard() in sets?

remove(x) - removes element x, raises KeyError if not present.

discard(x) - removes element x, does nothing if not present.

## How do you convert a list into a tuple?

list is converted into tuple using tuple() function

```
lst = [1, 2, 3]  
tup = tuple(lst)
```

---

## **\*\* Control Flow \*\***

## How does the if-elif-else structure work in Python?

if-elif-else allows conditional branching

```
x = 10
if x > 0:
    print("Positive")
elif x == 0:
    print("Zero")
else:
    print("Negative")
```

---

## What is the difference between for and while loops?

for loop -

Iterates over a sequence (list, range, string, etc.).

while loop -

Repeats as long as a condition is true.

## How do you loop through a dictionary's keys and values?

```
my_dict = {"a": 1, "b": 2}
for key, value in my_dict.items():
    print(key, value)
```

## How do you break out of a loop?

To break out of a loop using 'break' keyword.

## How do you skip the current iteration in a loop?

To skip the current iteration in loop we use 'continue' keyword.

## What is an infinite loop? Give an example.

A loop that never ends because its condition is always true.

```
while True:  
    print("infinite")
```

## How do you use the range() function?

1. range(stop)
  2. range(start, stop)
  3. range(start, stop, step)
- 

## What's the difference between range(5) and range(1,5)?

range(5) -

0,1,2,3,4 range(1,5) -

1,2,3,4

---

## How do you iterate over both index and value in a list?

To iterate over both index and value is done by using enumerate().

```
fruits = ["apple", "banana"]  
for index, value in enumerate(fruits):  
    print(index, value)
```

## How does the else clause work with loops in Python?

In python, else clause runs if the loop completes normally (without break)

## How do you use a nested loop?

A nested loop is a loop inside another loop



```
for i in range(2):  
    for j in range(3):  
        print(i, j)
```

---

## How do you loop through multiple lists simultaneously?

Looping through multiple lists is done by using zip() function

## How do you reverse iterate over a list?

```
nums = [1, 2, 3]  
for x in reversed(nums):  
    print(x)
```

---

## What is a list comprehension? Give an example.

Comprehension is a concise way to create lists.

```
squares = [x*x for x in range(5)]
```

---

## How do you use a conditional inside a list comprehension?

```
even = [x for x in range(10) if x % 2 == 0]
```

---

## \*\* Functions \*\*

## How do you define a function in Python?

In python, a function is defined using the def keyword.

```
def function_name(parameters):  
    return value
```

---

## **What is the difference between positional and keyword arguments?**

Positional arguments -

Matched by order.

Keyword arguments -

Matched by name.

## **What are default parameter values in functions?**

Parameters can have default values if not provided by caller.

## **How do you pass a variable number of arguments to a function?**

**\*args** (for variable positional arguments):

The asterisk (\*) before a parameter name in a function definition indicates that the parameter will collect all excess positional arguments into a tuple.

**kwargs** (for variable keyword arguments):

The double asterisk (\*\*) before a parameter name in a function definition indicates that the parameter will collect all excess keyword arguments into a dictionary.

## **How do you return multiple values from a function?**

We can return multiple values from a function as a tuple (or list/dict)

## **What is a lambda function? Give an example.**

A lambda function is a small anonymous function. A lambda function can take any number of arguments, but can only have one expression.

```
square = lambda x: x*x  
print(square(5))
```

## What's the difference between local and global variables?

### Local -

A local variable is declared inside a function, accessible only there.

### Global -

A global variable is declared outside functions, accessible everywhere.

## How do you modify a global variable inside a function?

By using the global keyword.

```
x = 10  
def change():  
    global x  
    x = 20  
change()  
print(x)    # 20
```

---

## What is recursion in Python? Give an example.

Recursion is a technique in which a function calls itself repeatedly until it reaches the base condition

```
def factorial(n):  
    if n == 1:  
        return 1  
    return n * factorial(n-1)  
  
print(factorial(5))    # 120
```

## What is a docstring in Python functions?

A docstring, short for "documentation string," in Python is a string literal that appears as the first statement in a module, function, class, or method definition. It is the String literal inside a function describing its purpose. example:

```
"""This function returns the sum of two numbers."""
```

## How do you use type hints in functions?

```
def add(a: int, b: int) -> int:  
    return a + b
```

## What are function annotations?

Function annotations in Python are a way to add metadata (like type hints) to function parameters and return values. They are optional and do not affect the execution of the code

## What is the purpose of the pass statement?

The pass statement in Python acts as a placeholder. It's used when a statement is syntactically required, but you don't want to execute any code there yet. Essentially, it's a "do nothing" operation.

**\*\* Object-Oriented Programming (OOP) \*\***

## How do you define a class in Python?

```
class Person:  
    def __init__(self, name):  
        self.name = name
```

## How do you create an object from a class?

```
p = Person("Alice")
```

## What is init in Python?

**init** is a special method (constructor) that initializes object attributes when an object is created.

## What is the difference between instance variables and class variables?

Instance variable -

Belongs to each object (defined with self).

Class variable

Shared by all objects globally (defined directly in class).

## What is inheritance in Python?

Inheritance is the property of OOPs where child class inherits the property of parent class

## How do you call a parent class constructor?

Parent class constructor is called by using `super()` keyword

## What's the difference between method overloading and overriding?

Overloading -

Multiple methods with same name but different parameters (not directly supported in Python; achieved with default args or \*args).

## Overriding -

Child class redefines a method from parent class.

## What is multiple inheritance?

When a class inherits from more than one parent.

## What's the role of super() in Python?

super() keyword is use to call the parent class.

## What are magic methods in Python? Give two examples.

Special methods surrounded by \_\_.

### **init** -

constructor

### **str** -

string representation example:

**len, add**, etc.

## What does str do in a class?

The **str** method in a Python class defines the informal, human-readable string representation of an object of that class.

## What is polymorphism in Python?

Different classes implementing the same method but behaving differently.

## What is encapsulation in Python?

Bundling data (attributes) and methods inside a class; restricting direct access using private attributes (`_var`, `__var`).

## What are @staticmethod?

A static method is a method that belongs to a class rather than an instance of the class.

## How do you make an attribute private in Python?

To make an attribute private in python we use `'__'` in starting of the attribute

## **\*\* Modules, Packages & File Handling \*\***

## How do you import a module in Python?

In python, a module is imported by using `import` keyword

## What's the difference between `import module` and `from module import`?

`import module`:

This statement imports the entire module into the current namespace.

`from module import name`:

This statement imports specific functions, classes, or variables directly into the current namespace.

## How do you find the location of an imported module?

```
import math
print(math.__file__)
```

---

## How do you create your own module in Python?

1. Create a .py file with functions/classes.

```
# mymodule.py
def greet(name):
    return f"Hello {name}"
```

2. Import it in another script

```
import mymodule
print(mymodule.greet("Shreya"))
```

Start coding or [generate](#) with AI.