1. **Interview Questions with Answers**

1 What is Python?

Python is a high-level, interpreted programming language known for its simplicity and readability. It emphasizes code readability and allows developers to express concepts in fewer lines of code compared to other languages.

2 What are the key features of Python?

Key features of Python include:

Easy-to-read syntax

Dynamic typing

Automatic memory management (garbage collection)

Rich library ecosystem

Cross-platform compatibility

Support for object-oriented programming

Extensibility through C/C++ integration

3 What are the differences between Python 2 and Python 3?

Python 2 and Python 3 have some differences in syntax and features. Python 3 is the newer version and is not backward-compatible with Python 2. Some key differences include:

4 Print statement vs. print() function

Unicode strings as the default

Integer division operator

Exception handling syntax

Different libraries and modules

5 Explain the GIL (Global Interpreter Lock) in Python.

The GIL is a mechanism in CPython (the reference implementation of Python) that ensures only one thread executes Python bytecode at a time. It prevents multiple native threads from executing Python code simultaneously, limiting parallelism in multi-threaded programs. However, it does not prevent the use of multiple threads for I/O-bound tasks.

6 What are Python decorators?

Decorators are a way to modify the behavior of a function or class without directly changing its source code. They allow you to wrap a function or class with additional functionality. Decorators are written using the @symbol followed by the decorator name and placed above the function or class definition.

7 Explain the difference between a shallow copy and a deep copy.

A shallow copy creates a new object that references the original elements. Changes made to the original object will be reflected in the copied object and vice versa. A deep copy creates a new object with its own copy of the original elements. Changes made to the original object will not affect the copied object, and vice versa.

8 What are the different data types in Python?

Python has several built-in data types, including:

9 Numeric types: int, float, complex

Sequence types: list, tuple, range

Text type: str

Mapping type: dict

Set types: set, frozenset

Boolean type: bool

Binary types: bytes, bytearray, memoryview

10 Explain list comprehensions in Python.

List comprehensions provide a concise way to create lists based on existing lists or other iterable objects. They follow the syntax: [expression for item in iterable if condition]. List comprehensions can include optional conditionals to filter elements and transform them into a new list.

11 What is a generator in Python?

A generator is a function that returns an iterator. It allows you to generate a sequence of values on-the-fly, instead of generating the entire sequence at once. Generators use the yield keyword to produce values one at a time, saving memory and improving performance.

12 How do you handle exceptions in Python?

Exceptions in Python can be handled using try-except blocks. The code that might raise an exception is placed inside the try block. If an exception occurs, it is caught by the appropriate except block based on the type of exception. The except block can handle the exception or perform any necessary cleanup tasks.

13 What are lambda functions in Python?

Lambda functions, also known as anonymous functions, are small, single-line functions without a name. They are defined using the lambda keyword and can take any number of arguments but can only have one expression. Lambda functions are often used as inline functions or for simple operations.

14Explain the difference between == and is in Python.

The == operator compares the values of two objects to check if they are equal. The is operator checks if two objects refer to the same memory location, i.e., they are the same object. In other words, == checks for value equality, while is checks for object identity.

15 What is a namespace in Python?

A namespace is a system that keeps track of names used in a program to avoid naming conflicts. It ensures that each name is unique and can be used to refer to a specific object. Namespaces can be global (module-level) or local (function-level) and help organize code and avoid naming collisions.

16 What is the difference between a module and a package in Python?

A module is a single file containing Python code, whereas a package is a directory that contains multiple Python modules or sub-packages. Packages are used to organize related modules and provide a hierarchical structure for organizing code.

17 How do you handle file I/O in Python?

File I/O in Python is handled using built-in functions and objects. To read from a file, you can use the open() function to create a file object and then read its contents using methods like read(), readline(), or readlines(). To write to a file, you can use methods like write() or writelines().

18 Explain the difference between \_\_str\_\_ and \_\_repr\_\_ in Python.

The \_\_str\_\_ method is called by the str() built-in function and returns a string representation of an object. It should provide a readable representation of the object and is intended for end-users. The \_\_repr\_\_ method returns a string representation that can be used to recreate the object and is intended for developers

19 What is the difference between a shallow copy and a deep copy in Python?

A shallow copy creates a new object that references the original elements. Changes made to the original object will be reflected in the copied object and vice versa. A deep copy creates a new object with its own copy of the original elements. Changes made to the original object will not affect the copied object, and vice versa.

20 What are the advantages of using Python for web development?

Some advantages of using Python for web development include:

Large ecosystem of web frameworks (e.g., Django, Flask) and libraries for various tasks.

Readable syntax and easy-to-learn language.

Rapid development and prototyping.

Good support for integration with databases and other technologies.

Cross-platform compatibility.

**Gutta Yeshwenth**

**DS150423**