1. **What is a File function in Python? What is the keyword to create and write a file?**
   * In Python, file functions allow operations like reading, writing, and appending data to files.
   * The open() function is used to create and write files. The keyword "w" (write mode) or "a" (append mode) is used for writing. Example:
   * file = open("example.txt", "w") # Opens or creates a file in write mode
   * file.write("Hello, Python!") # Writes to the file
   * file.close() # Closes the file
2. **Explain Exception handling? What is an Error in Python?**
   * Exception handling is a mechanism to handle runtime errors and prevent program crashes. It is done using try-except blocks.
   * An error in Python refers to issues in code that cause it to fail. These include **SyntaxErrors** (incorrect syntax) and **Exceptions** (runtime errors like ZeroDivisionError, ValueError, etc.).
3. **How many except statements can a try-except block have? Name some built-in exception classes.**
   * A try-except block can have multiple except statements to handle different exceptions separately.
   * Some built-in exception classes:
     + ZeroDivisionError
     + ValueError
     + TypeError
     + IndexError
     + KeyError
4. **When will the else part of try-except-else be executed?**
   * The else block executes only if no exceptions occur inside the try block.
5. **Can one block of except statements handle multiple exceptions?**
   * Yes, multiple exceptions can be handled in a single except block by passing them as a tuple.
   * try:
   * x = int("abc")
   * except (ValueError, TypeError):
   * print("An error occurred!")
6. **When is the finally block executed?**
   * The finally block is always executed, regardless of whether an exception occurs or not.
7. **What happens when "1" == 1 is executed?**
   * It returns False because "1" is a string, while 1 is an integer, and Python does not automatically convert them for comparison.
8. **How Do You Handle Exceptions With Try/Except/Finally In Python? Explain with coding snippets.**
   * Example of exception handling with try-except-finally:
   * try:
   * result = 10 / 0 # This will raise ZeroDivisionError
   * except ZeroDivisionError:
   * print("Cannot divide by zero!")
   * finally:
   * print("Execution completed.") # This will always run
9. **What are OOP concepts? Is multiple inheritance supported in Python?**
   * OOP (Object-Oriented Programming) concepts in Python include:
     + **Encapsulation**
     + **Abstraction**
     + **Inheritance**
     + **Polymorphism**
   * Yes, Python supports multiple inheritance.
10. **How to Define a Class in Python? What Is Self? Give An Example Of A Python Class.**
    * A class is defined using the class keyword.
    * self is a reference to the instance of the class, used to access attributes and methods.
    * Example:
    * class Person:
    * def \_\_init\_\_(self, name):
    * self.name = name
    * def greet(self):
    * print("Hello, my name is", self.name)
    * p = Person("Alice")
    * p.greet()
11. **Explain Inheritance in Python with an example? What is \_\_init\_\_? Or What Is A Constructor In Python?**
    * Inheritance allows a class to inherit properties and behavior from another class.
    * \_\_init\_\_ is a constructor method that initializes object attributes.
    * Example:
    * class Animal:
    * def \_\_init\_\_(self, name):
    * self.name = name
    * def make\_sound(self):
    * print("Animal sound")
    * class Dog(Animal): # Inheriting from Animal
    * def make\_sound(self):
    * print("Bark!")
    * d = Dog("Buddy")
    * d.make\_sound() # Output: Bark!
12. **What is Instantiation in terms of OOP terminology?**
    * Instantiation is the process of creating an object (instance) from a class.
13. **What is used to check whether an object o is an instance of class A?**
    * The isinstance(o, A) function is used to check if o is an instance of class A.