Python Assignment 3

What is File function in python? What is keywords to create

and write file.

In Python, file handling is done using the built-in open() function. The open() function opens a file and returns a file object, which can be used to read or write data to the file.

Syntax

file\_object = open('filename', 'mode')

Explain Exception handling? What is an Error in Python?

Exception handling in Python is done using try, except, else, and finally blocks. The try block contains code that might throw an exception, the except block handles the exception, the else block runs if no exceptions occur, and the finally block executes regardless of whether an exception was raised.

try:

# Code that might raise an exception

result = 10 / 0

except ZeroDivisionError as e:

print("Cannot divide by zero:", e)

else:

print("No exceptions occurred.")

finally:

print("This will run no matter what.")

Name Some built-in exception classes?

•ValueError

•IndexError

•TypeError

•KeyError

•ZeroDivisionError

When will the else part of try-except-else be executed?

The else part is executed if no exceptions are raised in the try block.

What happens when „1‟== 1 is executed?

The expression '1' == 1 will return False because '1' is a string and 1 is an integer. They are of different types, so they are not equal.

How Do You Handle Exceptions With Try/Except/Finally In

Python? Explain with coding snippets.

You handle exceptions using the try, except, and finally blocks.

try:

x = int(input("Enter a number: "))

y = 10 / x

except ZeroDivisionError as e:

print("You cannot divide by zero!", e)

except ValueError as e:

print("You must enter an integer!", e)

else:

print(f"Result is {y}")

finally:

print("This will always execute, whether an exception occurred or not.")

What are oops concepts? Is multiple inheritance supported in python

Object-Oriented Programming (OOP) concepts include:

• **Class**: A blueprint for creating objects.

• **Object**: An instance of a class.

• **Inheritance**: A way to form new classes using classes that have already been defined.

• **Polymorphism**: The ability to use a common interface for multiple forms (data types).

• **Encapsulation**: Keeping the internal state of an object hidden from the outside world.

• **Abstraction**: Hiding the complex reality while exposing only the necessary parts.

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.

class MyClass:

def \_\_init\_\_(self, name):

self.name = name # `self` refers to the current instance

def greet(self):

return f"Hello, {self.name}!"

# Usage

obj = MyClass("Mohammed")

print(obj.greet())

**self:** A reference to the current instance of the class. It is used to access variables that belong to the class.

Explain Inheritance in Python with an example? What is init? Or

What Is A Constructor In Python?

Inheritance allows a new class (child class) to inherit attributes and methods from an existing class (parent class).

**\_\_init\_\_:** This is a constructor method in Python classes that is automatically called when an object is created.

What is Instantiation in terms of OOP terminology?

**Instantiation** is the process of creating an object (instance) from a class.

What is used to check whether an object o is an instance of class

A?

You can check if an object is an instance of a class using the isinstance() function.

example

isinstance(obj, MyClass)

What relationship is appropriate for Course and Faculty?

**Course and Faculty**: The relationship could be a “has-a” relationship (Composition or Aggregation). A Faculty “has-a” Course or “teaches” a Course.

What relationship is appropriate for Student and Person?

**Student and Person**: The relationship is an “is-a” relationship (Inheritance). A Student “is-a” Person.