Que.1-Introduction to exceptions and how to handle them using try,except and finally.

Ans. In programming exceptions are unexpected events that disrupt the normal flow of a program’s execution.

It’s can arise from various sources:

* Runtime Errors : incorrect input , invalid operations (e.g., division by zero),memory issues.
* Logical Errors : incorrect algorithm , invalid assumptions.
* External Factors : Network problems , hardware failures.
* The try , except and finally blocks:

Try :~ this block encloses the code that might potentially raise an exception.

Except:~ this block is executed only if an exception occurs within the try block . you can specify the type of exception to handle.

Finally:~ this block is executed regardless of whether an exception occurred or not.it’s often used for cleanup tasks.

* Ex.,

Try:

Result=10/0

Except ZeroDivisionError:

Print(“Error:Division by zero!”)

Except Exception as e:

Print(f”an unexpected error occurred:{e}”)

Finally:

Print(“This block always executes.”)

Que.2-Understanding multiple exceptions and custom exceptions.

Ans. **Multiple Exception:~**

User can have multiple exception blocks to handle different types of exceptions specifically.

**Ex.,**

Try:

Result=10/0

My\_list=[1,2,3]

Print(my\_list[5])

Except ZeroDivisionError:

Print(“Error:Division by zero!”)

Except IndexError:

Print(“Error:List index out of range.”)

Except Exception as e:

Print(f”An unexpected error occurred:{e}”)

**Custom Exception:~**

User can define it’s own exception classes by extending the built in exception class. This allow you to create specific exception types for your aapplication’s unique error conditions.

**Ex.,**

Class AgeTooLowError(Exception):

Pass

Class AgeTooHighError(Exception):

Pass

Def check\_age(age):

If age<18:

Raise AgeTooLowError(“age is too low.must be at least 18 years old.”)

Elif age>60:

Raise AgeTooHighError(“age is too high.must be at most 60 years old.”)

Else:

Print(“age is within the acceptable range.”)

Try:

Age=int(input(“enter your age:”))

Check\_age(age)

Except AgeTooLowError as e:

Print(e)

Except AgeTooHighError as e:

Print(e)