**5 ) Exception Handling**

* **Introduction to exceptions and how to handle them using try, except, and finally.**

**What is an Exception?**

An exception is an error that occurs during the execution of a program. When Python encounters an error, it interrupts the normal flow of the program and raises an exception. If exceptions are not handled properly, the program will crash.

**Common Examples of Exceptions:**

**ZeroDivisionError** - Division by zero

**ValueError** - Wrong value type

**FileNotFoundError** - File operation on a non-existent file

**IndexError** - Index out of range in a list

**TypeError** - Operation on incompatible data types

**Exception Handling in Python**

Python provides a method to catch and handle exceptions using the try, except, and finally blocks.

1. **Try Block**

The code that may cause an exception is placed inside the try block.

1. **except Block**

If an exception occurs in the try block, it is caught by the except block, where you can define how to handle the error.

1. **finally Block**

The finally block contains code that is always executed, whether an exception occurs or not. It's typically used for cleanup actions, like closing a file.

Syntax:

try:

# Code that might raise an exception

except SomeException:

# Code to handle the exception

finally:

# Code that runs no matter what

* **Understanding multiple exceptions and custom exceptions.**

1. **Multiple Exceptions**

**What is Multiple Exceptions:**

When a program runs, it may encounter more than one type of error. Python allows us to handle different errors separately using multiple except blocks, or together using a tuple of exceptions.

**Why Handle Multiple Exceptions?**

Different errors may require different handling. For example:

* ValueError: If the user enters a string instead of a number.
* ZeroDivisionError: If the user enters 0 and you try to divide by it.
* FileNotFoundError: If a file is missing.

1. **Custom Exceptions**

**What are Custom Exceptions?**

Python has many built-in exceptions, but sometimes we need to define our own errors for special situations in our programs. These are called custom exceptions.

* If the input is a negative number, our custom NegativeNumberError is raised.
* This allows us to define our own rules and error messages.