Home Assignment <9>: Bank Account System with OOPs

Learning Objective:

The objective of this assignment is to learn how to model real-world entities (like a bank account) in Python using classes, attributes, and methods, along with implementing error handling for invalid operations.

Expected Completion Time:

Best Case: 15 minutes
Average Case: 25 minutes

Assignment Details:

Create a Python class named BankAccount to represent and manage basic banking operations.

Requirements:

a) Create a class named `BankAccount`.
b) Inside the class, define the following attributes:

1. `account_holder` → string
2. `balance` → float
3. `account_type` → string (e.g., "Savings" or "Current")

c) Implement the following methods:

1. `deposit(amount)` → increases the balance by the given amount.
2. `withdraw(amount)` → decreases the balance by the given amount if sufficient funds are available, otherwise display `"Insufficient balance"`.
3. `display_balance()` → prints the account holder's name, account type, and current balance.

d) In the main section (`if __name__ == "__main__":`):

- Create at least two `BankAccount` objects with different details.
- Perform deposit and withdrawal operations.
- Display the account details after each operation.

Expected Outcome:

Upon completion of this assignment, you should be able to:

- Create and initialize a Python class with attributes.
- Implement methods to perform actions on class objects.
- Instantiate multiple objects and manage them separately.
- Apply OOP principles to represent and handle real-world data.