**Assignment 3- Polymorphism in Python - Bank System**

In this assignment, we will be creating a bank system using inheritance in Python. The system will have a parent class, Account, and two child classes, SavingsAccount and CheckingAccount, each inheriting attributes and methods from the Account class.

**Assignment: Polymorphism in Python - Bank System**

**Step 1: Create a Parent Class**

The BankAccount class is created as the parent class, with basic attributes and methods that are common to all types of bank accounts.

**Step 2: Create Child Classes with Overridden Methods**

The **CheckingAccount** and **SavingsAccount** classes are created as child classes that inherit from the **BankAccount** class. They override the **withdraw** and **deposit** methods to provide specific functionalities for each type of account.

**Conclusion**

This assignment showcased the implementation of polymorphism in a complex bank system in Python. The BankAccount class served as the parent class, while the CheckingAccount and SavingsAccount classes demonstrated the overriding of methods to provide specific functionality for each account type. Polymorphism enabled the use of a common interface for different types of bank accounts, enhancing code reusability and flexibility.

**Solution**:

class BankAccount:  
 def \_\_init\_\_(self, account\_number, balance):  
 self.account\_number = account\_number  
 self.balance = balance  
  
 def display\_info(self):  
 print(f"Account Number: {self.account\_number}, Balance: {self.balance}")  
 return  
  
 def deposit(self, amount):  
 pass  
  
 def withdraw(self, amount):  
 pass  
  
class CheckingAccount(BankAccount):  
 def \_\_init\_\_(self, account\_number, balance):  
 super().\_\_init\_\_(account\_number, balance)  
  
 def withdraw(self, amount):  
 self.balance -= amount  
  
 def deposit(self, amount):  
 self.balance += amount  
  
 def display\_info(self):  
 print(f"Account Number: {self.account\_number}, New Balance: {self.balance}")  
  
  
class SavingsAccount(BankAccount):  
 def \_\_init\_\_(self, account\_number, balance):  
 super().\_\_init\_\_(account\_number, balance)  
  
 def withdraw(self, amount):  
 self.balance -= amount  
  
 def deposit(self, amount):  
 self.balance += amount  
  
 def display\_info(self):  
 print(f"Account Number: {self.account\_number}, New Balance: {self.balance}")  
  
  
# Demonstrate polymorphism  
account\_list = [  
 CheckingAccount(12345, 10000),  
 SavingsAccount(67890, 20000)  
]  
  
for account in account\_list:  
 account.withdraw(5000)  
 account.display\_info()  
 account.deposit(10000)  
 account.display\_info()