# Department of Computing

**CS-213: Advanced Programming**

**Name: M. Hasnain Naeem (212728)**

**Class: BSCS 7AB**

# Lab Quiz # 01

# Task

Write a program of bank management system to manage the account information using inheritance concept.

Create a class “Bank Account” with the customer\_name, account\_number etc. as member variables. Create the derived classes for two types of accounts i.e. current and saving. The derived classes will update the balance and handle the deposit and withdraw cases. Customers should be able to get updated balance after deposit and withdrawal amounts.

**Answer:**

|  |
| --- |
| Solution |
| Task Code:  Main   |  | | --- | | *public class* Main {  *public static void* main(String argz[]){  *// Demonstrating the bank account classes* CurrentBankAccount currentAccount = *new* CurrentBankAccount("M. Hasnain Naeem", 50000);  SavingBankAccount savingsAccount = *new* SavingBankAccount("Muazzam Ali Kazmi", 1000000);   BankAccount[] accounts = {currentAccount, savingsAccount};   *for*(BankAccount account: accounts){  System.*out*.println(account.toString());  System.*out*.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");  }  } } |   BankAccount   |  | | --- | | *public abstract class* BankAccount {  *private* String customerName;  *private long* balance;   *public* BankAccount(String customerName, *long* balance) {  *this*.customerName = customerName;  *this*.balance = balance;  }   *public void* withdraw(*long* amount) *throws* Exception{  *if*(balance - amount >= 0)  balance -= amount;  *else if*(balance - amount < 0)  *throw new* Exception("Error: cannot withdraw more amount than the balance.");  }   *public void* deposit(*long* amount){  balance += amount;  }   *public* String getCustomerName() {  *return* customerName;  }   *public void* setCustomerName(String customerName) {  *this*.customerName = customerName;  }   *public long* getBalance() {  *return* balance;  }   @Override  *public* String toString() {  *return* "BankAccount{" +  "customerName='" + customerName + '\'' +  ", balance=" + balance +  '}';  } } |   CurrentBankAccount   |  | | --- | | */\* Current account has its own withdraw limit.  \*/  public class* CurrentBankAccount *extends* BankAccount {  *private long* withdrawLimit;  *private static* String *accountType*;   *public* CurrentBankAccount(String customerName, *long* balance) {  *super*(customerName, balance);  *accountType* = "current";  withdrawLimit = 100000;  }   @Override  *public void* withdraw(*long* amount) *throws* Exception{  *if*(amount > withdrawLimit)  *throw new* Exception("Error: cannot withdraw more amount than the withdraw limit.");  *else if*(getBalance() - amount < 0)  *throw new* Exception("Error: cannot withdraw more amount than the balance.");  *else if*(getBalance() - amount >= 0)  withdraw(amount);  }   *public static* String getAccountType(){  *return accountType*;  }   @Override  *public* String toString() {  *return* "CurrentBankAccount{" +  "withdrawLimit=" + withdrawLimit +  "} " + *super*.toString();  } } |   SavingBankAccount   |  | | --- | | */\* Saving account has its own withdraw limit.  \*/  public class* SavingBankAccount *extends* BankAccount {  *private long* withdrawLimit;  *private static* String *accountType*;   *public* SavingBankAccount(String customerName, *long* balance) {  *super*(customerName, balance);  withdrawLimit = 50000;  *accountType* = "Saving";  }   @Override  *public void* withdraw(*long* amount) *throws* Exception{  *if*(amount > withdrawLimit)  *throw new* Exception("Error: cannot withdraw more amount than the withdraw limit.");  *else if*(getBalance() - amount < 0)  *throw new* Exception("Error: cannot withdraw more amount than the balance.");  *else if*(getBalance() - amount >= 0)  withdraw(amount);  }   *public static* String getAccountType(){  *return accountType*;  }   @Override  *public* String toString() {  *return* "SavingBankAccount{" +  "withdrawLimit=" + withdrawLimit +  "} " + *super*.toString();  } } |   Task Output Screenshot: |

### Deliverables

Compile a single word document by filling in the solution part and submit this Word file on LMS.