A blue and white logo

Description automatically generated

**Assignment 2**

Submitted By M Shahwaiz Shahid

Reg no. FA22 BCS 049

Subject OOP

Submitted To Mr. Shafqat

Date 23rd Oct 2023

1. Imagine you are developing a banking application. You have a base class **Account** that includes properties like **account Number**, **account Holder Name**, and **balance**. There are two derived classes, **Savings Account** and **Current Account**, which inherit from Account. In addition to the base properties, SavingsAccount has an extra property called **interest Rate** for calculating interest on the account balance. Users can deposit money into both **Savings Account** and **Current Account**. The **Current Account** allows users to withdraw funds, ensuring they don't exceed their account balance and the overdraft limit. The **Savings Account** calculates and adds interest to the balance over time. Users can view their account information, including the account number, account holder name, current balance, and interest rate, for **Savings Account** or the account number, account holder name, and current balance for **Current Account**.

**Program:**

class Account {

private int accountNumber;

private String accountHolderName;

protected double balance;

public Account(int accountNumber, String accountHolderName, double balance) {

this.accountNumber = accountNumber;

this.accountHolderName = accountHolderName;

this.balance = balance;

}

public int getAccountNumber() {

return accountNumber;

}

public String getAccountHolderName() {

return accountHolderName;

}

public double getBalance() {

return balance;

}

}

class SavingsAccount extends Account {

private double interestRate;

public SavingsAccount(int accountNumber, String accountHolderName, double balance, double interestRate) {

super(accountNumber, accountHolderName, balance);

this.interestRate = interestRate;

}

public double getInterestRate() {

return interestRate;

}

public void calculateInterest() {

balance += balance \* (interestRate / 100);

}

public void viewAccountInfo() {

System.out.println("Account Number: " + getAccountNumber());

System.out.println("Account Holder Name: " + getAccountHolderName());

System.out.println("Balance: " + getBalance());

System.out.println("Interest Rate: " + interestRate + "%");

}

}

class CurrentAccount extends Account {

private double overdraftLimit;

public CurrentAccount(int accountNumber, String accountHolderName, double balance, double overdraftLimit) {

super(accountNumber, accountHolderName, balance);

this.overdraftLimit = overdraftLimit;

}

public double getOverdraftLimit() {

return overdraftLimit;

}

public void withdraw(double amount) {

if (balance - amount >= -overdraftLimit) {

balance -= amount;

System.out.println("Withdrawn: " + amount);

} else {

System.out.println("Insufficient funds.");

}

}

public void viewAccountInfo() {

System.out.println("Account Number: " + getAccountNumber());

System.out.println("Account Holder Name: " + getAccountHolderName());

System.out.println("Balance: " + getBalance());

System.out.println("Overdraft Limit: " + overdraftLimit);

}

}

public class Main {

public static void main(String[] args) {

SavingsAccount savingsAccount = new SavingsAccount(1001, "M. Shahwaiz", 5000, 3.5);

CurrentAccount currentAccount = new CurrentAccount(2001, "Ali", 3000, 1000);

savingsAccount.balance += 2000;

currentAccount.balance += 1500;

currentAccount.withdraw(2000);

savingsAccount.calculateInterest();

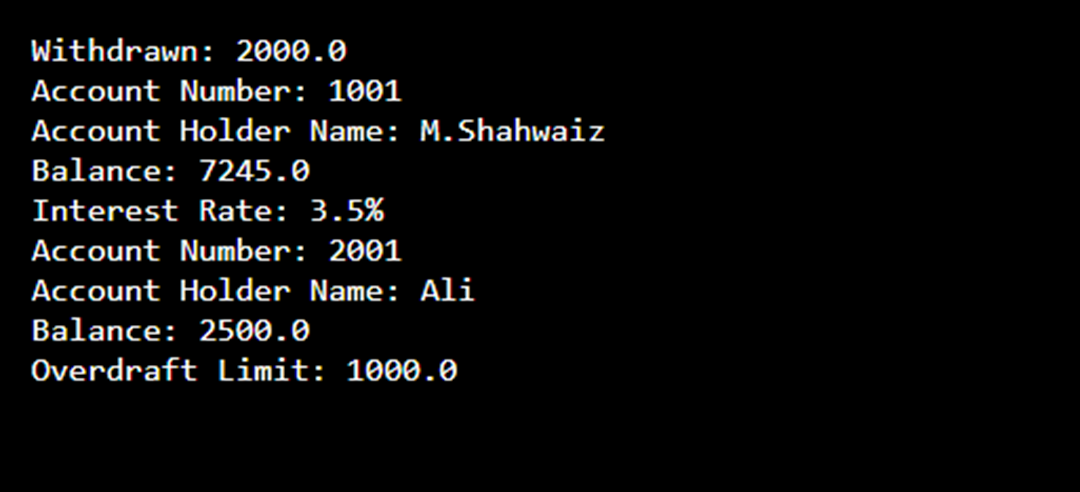
savingsAccount.viewAccountInfo();

currentAccount.viewAccountInfo();

}

}

**Output:**

****

…………………………………………………..

1. Imagine you are designing a zoo management software. You start with a base class called Animal that includes attributes like name and age. Two derived classes, Mammal and Bird, inherit from Animal. Mammals have additional properties like furColor and can perform actions like giveBirthToYoung, while Birds have properties like wingSpan and can perform actions like fly. The zoo software allows zookeepers to record and display information about the animals. For mammals, it includes details about fur color and birth records, and for birds, it shows information about wing span and flight capabilities.

**Program:**

class Animal

{

private String name;

private int age;

public Animal(String name, int age)

{

this.name = name;

this.age = age;

}

public String getName()

{

return name;

}

public int getAge()

{

return age;

}

}

class Mammal extends Animal

{

private String furColor;

public Mammal(String name, int age, String furColor) {

super(name, age);

this.furColor = furColor;

}

public String getFurColor()

{

return furColor;

}

public void giveBirthToYoung()

{

System.out.println(getName() + " is giving birth to young.");

}

public void displayAnimalInfo()

{

System.out.println("Name: " + getName());

System.out.println("Age: " + getAge());

System.out.println("Fur Color: " + furColor);

}

}

class Bird extends Animal

{

private double wingSpan;

public Bird(String name, int age, double wingSpan)

{

super(name, age);

this.wingSpan = wingSpan;

}

public double getWingSpan()

{

return wingSpan;

}

public void fly()

{

System.out.println(getName() + " is flying with a wing span of " + wingSpan + " meters.");

}

public void displayAnimalInfo()

{

System.out.println("Name: " + getName());

System.out.println("Age: " + getAge());

System.out.println("Wing Span: " + wingSpan + " meters");

}

}

public class ZooManagement

{

public static void main(String[] args)

{

Mammal lion = new Mammal("Simba", 5, "Golden");

Bird eagle = new Bird("Baldy", 10, 2.0);

lion.displayAnimalInfo();

lion.giveBirthToYoung();

eagle.displayAnimalInfo();

eagle.fly();

}

}

**Output:**

A screenshot of a computer program

Description automatically generated

……………………………………………………