

Account Manipulation - Abstract class

Yzee bank needs to automate the bank transactions. There are many accounts, like Savings Account, Current Account, Demat Account and so on.

As start up, they need to automate the Savings Account details.

You are provided with a public class Customer with private attributes :

`int customerId`

`String customerName`

`String emailId`

Appropriate public getters and setters are already written.

Write a public 3 argument constructor with arguments – customerId, customerName and emailId.

Write a public class Account with protected attributes :

`int accountNumber`

`Customer customerObj`

`double balance`

Uncomment the public getters and setters provided in the template.

Write a public 3 argument constructor with arguments – accountNumber, customerObj and balance.

Write a public method in Account class as,

`public boolean withdraw(double amount)` – Make this method as abstract.

Write a public class SavingsAccount with private attribute :

`double minimumBalance`

Uncomment the public getters and setters provided in the template.

Make this class SavingsAccount to inherit the Account class.

Write a public 4 argument constructor with arguments – accountNumber, customerObj, balance and minimumBalance.

Implement the withdraw method as

public boolean withdraw(double amount) - This method should return true if withdraw is successful, else return false.

In this method, check if

balance - amount is greater than the minimum balance.

If yes, perform withdraw. Reduce the withdraw amount from the balance and return true.

If not, return false.

Create a public class Main which has the main method. Check the correctness of the methods written in these classes.

Note : All class, methods needs to be declared as public