**public** **class** Account

{

**public** Account()

{

**super**();

}

**private** **long** accountnumber;

**private** **int** pin;

**private** **double** balance;

**public** String name;

**public** Account(**long** accountnumber,**int** pin,**double** balance,String name)//Parameterized constructor

{

**super**();

**this**.accountnumber=accountnumber;

**this**.pin=pin;

**this**.balance=balance;

**this**.name=name;

}

//getter and setter method used to access private variables.

**public** **long** getAccountNumber()

{

**return** accountnumber;

}

**public** **void** setAccountNumber(**long** accountnumber)

{

**this**.accountnumber=accountnumber;

}

**public** **int** getPin()

{

**return** pin;

}

**public** **void** setPin(**int** pin)

{

**this**.pin=pin;

}

**public** **double** getBalance()

{

**return** balance;

}

**public** **void** setBalance(**double** balance)

{

**this**.balance=balance;

}

}

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** AccountRepository

{

**private** **static** Scanner *scanner*=**new** Scanner(System.***in***);

**public** **static** List<Account> generateAccount()//List used to store the bank customer details.

{

List<Account> accountlist=**new** ArrayList<>();

accountlist.add(**new** Account(1234567890,1234,10000,"Privarthini.B"));

accountlist.add(**new** Account(1234567891,2345,8500,"MohanaPriya"));

accountlist.add(**new** Account(1234567892,8888,1777,"Kalaiselvi"));

accountlist.add(**new** Account(1234567893,3333,50000,"Praveen"));

accountlist.add(**new** Account(1234567894,7532,2500,"Madhumitha"));

accountlist.add(**new** Account(1234567895,9512,3000,"Divyadarsini"));

**return** accountlist;

}

**public** **static** Account authenticateAccount(**long** accountnumber,**int** pin)//method used to check the account number and pin number

{

List<Account> accounts = *generateAccount*();

**for**(Account user:accounts)

{

**if**(user.getAccountNumber()==accountnumber)

{

**if**(user.getPin()==pin)

{

**return** user;

}

**else** **if**(user.getPin()!=pin)

{

System.***out***.println("Entered Pin number is wrong");

**return** **null**;

}

}

}

**return** **null**;

}

**public** **static** **void** deposit(Account account)//method used to deposit amount

{

System.***out***.println("Enter amount to deposit");

**double** amount=*scanner*.nextDouble();

**double** balance =account.getBalance()+amount;

account.setBalance(balance);

System.***out***.println("Sucessfully Amount Deposited : "+amount);

}

**public** **static** **void** printData(Account account)//method used to display the user details

{

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

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

System.***out***.println("User Name : "+account.name);

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

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

}

**public** **static** **void** withDraw(Account account)//method used for withdraw the money

{

System.***out***.println("Enter amount to withdraw");

**double** amount=*scanner*.nextDouble();

**if**((amount-account.getBalance())>account.getBalance())

{

System.***out***.println("Insufficient Balance");

}

**else** **if**((account.getBalance()-amount)<500)

{

System.***out***.println("Insufficient Withdrawn : Minimum balance should be 500");

}

**else**

{

**double** balance=account.getBalance()-amount;

account.setBalance(balance);

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

}

}

**public** **static** Account newAccount()//to create new account

{

**double** intialbalance=0.0;

Account newuser=**new** Account();

newuser.setAccountNumber(*generateAccountNumber*());

newuser.setPin(*generatePin*());

newuser.setBalance(intialbalance);

System.***out***.println("Enter your Name");

String name=*scanner*.next();

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("Account was Successfully created");

System.***out***.println("\n Your Account Details : \n Account Number:"+newuser.getAccountNumber()+"\n Your Pin Number : "+newuser.getPin());

System.***out***.println(" User Name : "+name);

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

**return** newuser;

}

**public** **static** **int** generatePin()//Generate Pin number for new user

{

**int** pin=(**int**)(Math.*random*()\*1000)+1000;

**return** pin;

}

**public** **static** **long** generateAccountNumber()//Generate Account number for new user

{

**long** intialvalue=1234567895;

**do**

{

System.***out***.println("Do You Want to Create new Account? \n (Y) or (N) ? press (E) to exit");

String input=*scanner*.next();

**switch**(input)

{

**case** "Y":

intialvalue+=1;

**return** intialvalue;

**case** "N":

System.***out***.println("Account cannot be created");

**break**;

**case** "E":

System.***out***.println("Exited");

**default**:

System.***out***.println("Enter valid option");

**break**;

}

}**while**(**true**);

}

}

**import** java.util.Scanner;

**public** **class** BankManagement {

**private** **static** Scanner *scanner*=**new** Scanner(System.***in***);

**private** **static** Account *account*=**null**;

**static** **long** *accountnumber*;

**static** **int** *pinnumber*;

**public** **static** **void** main(String[] args)

{

*validate*(); //main class call the method validate

}

**public** **static** **void** mainMenu()//method to display main menu

{

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("Welcome to Aspire Bank");

System.***out***.println("Hi "+*account*.name);

**do**

{

System.***out***.println("Choose from menu");

System.***out***.println("1.Deposit");

System.***out***.println("2.Withdraw");

System.***out***.println("3.View Balance");

System.***out***.println("4.Exit");

**int** choice=*scanner*.nextInt();

**switch**(choice)

{

**case** 1:

AccountRepository.*deposit*(*account*);

**break**;

**case** 2:

AccountRepository.*withDraw*(*account*);

**break**;

**case** 3:

AccountRepository.*printData*(*account*);

**break**;

**case** 4:

System.***out***.println("Thank you");

*validate*();

**default**:

System.***out***.println("Enter a valid option");

**break**;

}

}**while**(**true**);

}

**public** **static** **void** validate()//validation of account number and pin number using regex.

{

**do**

{

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println("1)Login\n2)New Account");

System.***out***.println("Enter an option");

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

**int** input=*scanner*.nextInt();

**switch**(input)

{

**case** 1:

System.***out***.println("Enter an account number");

*accountnumber*=*scanner*.nextLong();

**boolean** pattern1=Long.*toString*(*accountnumber*).matches("\\d{10,16}");

**if**(pattern1==**false**)

{

System.***out***.println("Account number should contain 10-16 digits");

*validate*();

}

**if**(pattern1==**true**)

{

System.***out***.println("Enter a Pin Number");

*pinnumber*=*scanner*.nextInt();

**boolean** pattern2=Long.*toString*(*pinnumber*).matches("\\d{4}");

**if**(pattern2==**false**)

{

System.***out***.println("Pin number should contain 4 digits");

*validate*();

}

}

*account*=AccountRepository.*authenticateAccount*(*accountnumber*,*pinnumber*);

**if**(*account*!=**null**)

{

*mainMenu*();

}

**else**

{

System.***out***.println("Account Not found");

}

**break**;

**case** 2:

AccountRepository.*newAccount*();

**break**;

**default**:

System.***out***.println("Invalid Input");

}

}**while**(**true**);

}

}