```
/**
Name: Atharva Arun Pandharikar
Class: SE-II    Batch: B   Roll No: 20

Program Name:
 Using concepts of Object Oriented programming develop solution for Banking solution
contains following operations such as
•Create an account
•Deposit money
•Withdraw money
•Honor daily withdrawal limit
•Check the balance
•Display Account information.

Experiment No: 09
*/
import java.util.Scanner;
class Customer {
private String customerName;
private int customerAge;
public void setCustomerName(String customerName){
this.customerName=customerName;
 }
public String getCustomerName(){
return customerName;
 }
public void setCustomerAge(int customerAge){
this.customerAge=customerAge; }
public int getCustomerAge(){
return customerAge;
}
}
abstract class Account { //creating abstract class account
protected double balance; //declaration of balance
protected int accountId; //declaration of accountId
protected String accountType; //declaration of accountType
protected Customer custobj; //declaration of customer obj
void setBalance(double balance){
this.balance=balance;
}
double getBalance(){
return balance;
}
void setAccountId(int accountId){
this.accountId=accountId;
}
int getAccountId(){
return accountId;
}
void setAccountType(String accountType){
this.accountType=accountType;
```

```
}
String getAccountType(){
return accountType; //returning value of accountType
}
void setCustomerObject(Customer custobj){
this.custobj=custobj;
}
Customer getCustomerObject(){
return custobj;
}
public abstract boolean withdraw(double amount);
}
//SAVING ACCOUNT CLASS
class SavingsAccount extends Account{
//inheriting Account class in SavingAccount
private double minimumBalance;
public void setMinimumBalance(double minimumBalance){
this.minimumBalance=minimumBalance; //setting minimumBalance
}
public double getMinimumBalance(){
return minimumBalance;
}
public boolean withdraw(double amount){
//method to return true or false
if((balance-amount)>minimumBalance){
//check whether withdraw amount is greater than minimumBalance
balance-=amount;
return true;
}
else
return false; //returning false
}
}
//BANK CLASS
class Bank {
public static Scanner sc=new Scanner(System.in);
public SavingsAccount a=new SavingsAccount();
public Customer c=new Customer();
public SavingsAccount createAccount(){

System.out.print("Enter your name: ");
String customername=sc.nextLine();
c.setCustomerName(customername);
System.out.print("Enter your age: ");
int customerage=sc.nextInt();
if(customerage<18){//check whether the age is less than 18
do{
System.out.print("Minimum age should be 18 to create an account.\nPlease enter valid age: ");
customerage=sc.nextInt();
}while(customerage<18); //if age is less than 18
}
```

```
c.setCustomerAge(customerage); //calling setCustomerName method
a.setCustomerObject(c);//calling setCustomerName method
System.out.print("Enter your account Id: ");
int accountid=sc.nextInt();
a.setAccountId(accountid);
System.out.print("Enter your account type: ");
String accounttype=sc.next();
a.setAccountType(accounttype);
System.out.print("Enter balance: ");
double balance=sc.nextDouble();//taking balance as input from user
a.setBalance(balance);//calling setBalance method
System.out.print("Enter minimum balance: ");
double minbalance=sc.nextDouble();
a.setMinimumBalance(minbalance);

return a;
}
void getWithdrawAmount(){
System.out.print("Enter the amount you want to withdraw: ");
double amount=sc.nextDouble(); //taking amount as input from user
if(amount>20000){ //check whether amount is greater than 20000
System.out.println("Withdrawal failed. Maximum limit of withdrawal in one transaction is
Rs.20000.");
}
else{ //if amount is less than 20000
if(a.withdraw(amount)==true){
System.out.println("Withdrawal successful. Balance is: "+a.getBalance());
}
else
System.out.println("Sorry!!! Not enough balance"); //if not enough balance
}
}
public void depositAmount(double amount){ //method to deposit Amount
double bal=a.getBalance()+amount; //previous balance + amount
a.setBalance(bal); //call setBalance method
System.out.println("Amount deposited successfully. Balance is: "+a.getBalance());
}
public void checkBalance(){ //method to check Balance
System.out.println("Balance is: "+a.getBalance());//calling getbalance method
}
public void displayAccountInformation(){ //method to display Account Information
System.out.println("Welcome "+c.getCustomerName()+"! Following are your account details:");
//display name of customer
System.out.println("Age :"+c.getCustomerAge()); //display Age of customer
System.out.println("Account Id: "+a.getAccountId()); //display Account Id of customer
System.out.println("Account Type: "+a.getAccountType()); //display Account Type of customer
System.out.println("Balance: "+a.getBalance()); //display Balance of customer
System.out.println("Minimum balance: "+a.getMinimumBalance()); //display Minimum balance of
customer
}
}
```

```
//MAIN CLASS
public class BankOperations{
public static void main(String[] args){
Scanner sc=new Scanner(System.in);
SavingsAccount a; //cresting object of SavingsAccount class
Bank bm=new Bank(); //cresting object of Bank class
do{
//menu driven program
System.out.println("\n\t1.Create Account\n\t2.Display Account\n\t3.Check Balance"
+ "\n\t4.Deposit Amount\n\t5.Withdraw Amount\n\t6.Exit");
System.out.print("Enter your choice: "); //printing on console
int choice=sc.nextInt(); //taking input from user
System.out.println("");
switch(choice) //switch case
{
case 1:
a=bm.createAccount();
System.out.println("=======================================================");
break;
case 2:
bm.displayAccountInformation();
System.out.println("=======================================================");
break;
case 3:
bm.checkBalance();
System.out.println("=======================================================");
break;
case 4:
System.out.print("Enter the amount you want to deposit: ");
double amount=sc.nextDouble();
bm.depositAmount(amount);
System.out.println("=======================================================");
break;
case 5:
bm.getWithdrawAmount();
System.out.println("=======================================================");
break;
case 6:
System.out.println("=======================================================");
return ; //stop execution of program
default:
System.out.println("INVALID INPUT !!");//printing invalid input
System.out.println("=======================================================");
break;
}

}while(true);
}
}
```

OUTPUT:-

```
1.Create Account
2.Display Account
3.Check Balance
4.Deposit Amount
5.Withdraw Amount
6.Exit
Enter your choice: 1
Enter your name: XYZ
Enter your age: 20
Enter your account Id: 12345
Enter your account type: Saving
Enter balance: 500
Enter minimum balance: 100
=================================================
1.Create Account
2.Display Account
3.Check Balance
4.Deposit Amount
5.Withdraw Amount
6.Exit
Enter your choice: 2
Welcome XYZ! Following are your account details:
Age :20
Account Id: 12345
Account Type: Saving
Balance: 500.0
Minimum balance: 100.0
=================================================
1.Create Account
2.Display Account
3.Check Balance
4.Deposit Amount
5.Withdraw Amount
6.Exit
Enter your choice: 3
Balance is: 500.0
=================================================
1.Create Account
2.Display Account
3.Check Balance
4.Deposit Amount
5.Withdraw Amount
6.Exit
Enter your choice: 4
Enter the amount you want to deposit: 1000
Amount deposited successfully. Balance is: 1500.0
=================================================
1.Create Account
2.Display Account
3.Check Balance
```

```
4.Deposit Amount
5.Withdraw Amount
6.Exit
Enter your choice: 5
Enter the amount you want to withdraw: 200
Withdrawal successful. Balance is: 1300.0
==================================================
1.Create Account
2.Display Account
3.Check Balance
4.Deposit Amount
5.Withdraw Amount
6.Exit
Enter your choice: 6
==================================================
```