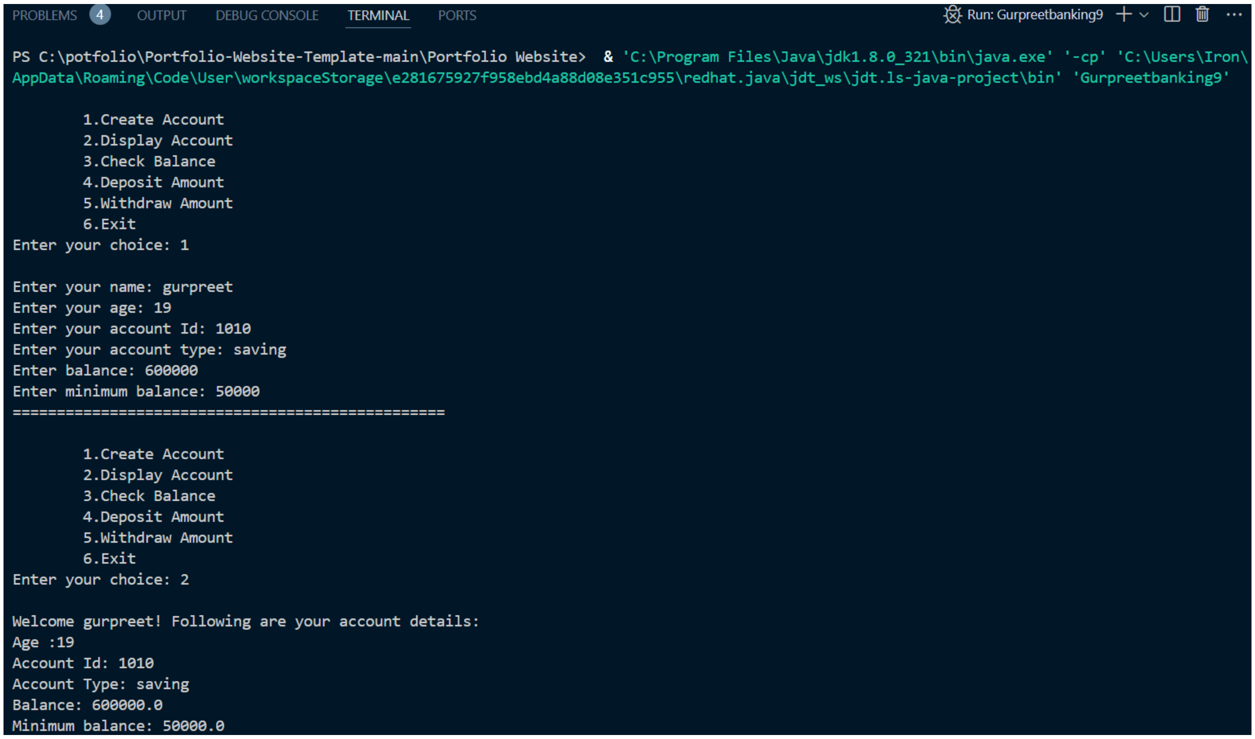
```
//package assignment;
import java.util.Scanner;
//CUSTOMER CLASS
class Customer {
        private String customerName;   //declaration of customerName
        private int customerAge;   //declaration of customerAge
        public void setCustomerName(String customerName){
                this.customerName=customerName;  //setting value of customerName
                }

        public String getCustomerName(){
                return customerName;   //returning value of customerName
                }

        public void setCustomerAge(int customerAge){
                this.customerAge=customerAge; //setting value of customerAge
                }

        public int getCustomerAge(){
                return customerAge;  //returning value of 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;   //setting value of balance
                }

         double getBalance(){
                return balance;  //returning value of balance
                }

         void setAccountId(int accountId){
                this.accountId=accountId;   //setting value of balance
                }

         int getAccountId(){
                return accountId; //returning value of accountId
                }

        void setAccountType(String accountType){
                this.accountType=accountType;  //setting value of balance
                }

        String getAccountType(){
                return accountType;  //returning value of accountType
                }

        void setCustomerObject(Customer custobj){
                this.custobj=custobj; //setting value of balance
                }

        Customer getCustomerObject(){
                return custobj;    //returning value of custobj
                }

        public abstract boolean withdraw(double amount); //abstract method withdraw

}


//SAVING ACCOUNT CLASS
class SavingsAccount extends Account{
        //inheriting Account class in SavingAccount
        private double minimumBalance;      //declaration of minimumBalance
        public void setMinimumBalance(double minimumBalance){
                this.minimumBalance=minimumBalance;    //setting minimumBalance
                }

        public double getMinimumBalance(){
                return minimumBalance;  //returning 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;  //balance minus amount
                        return true;      //returning true
                        }
                else
```

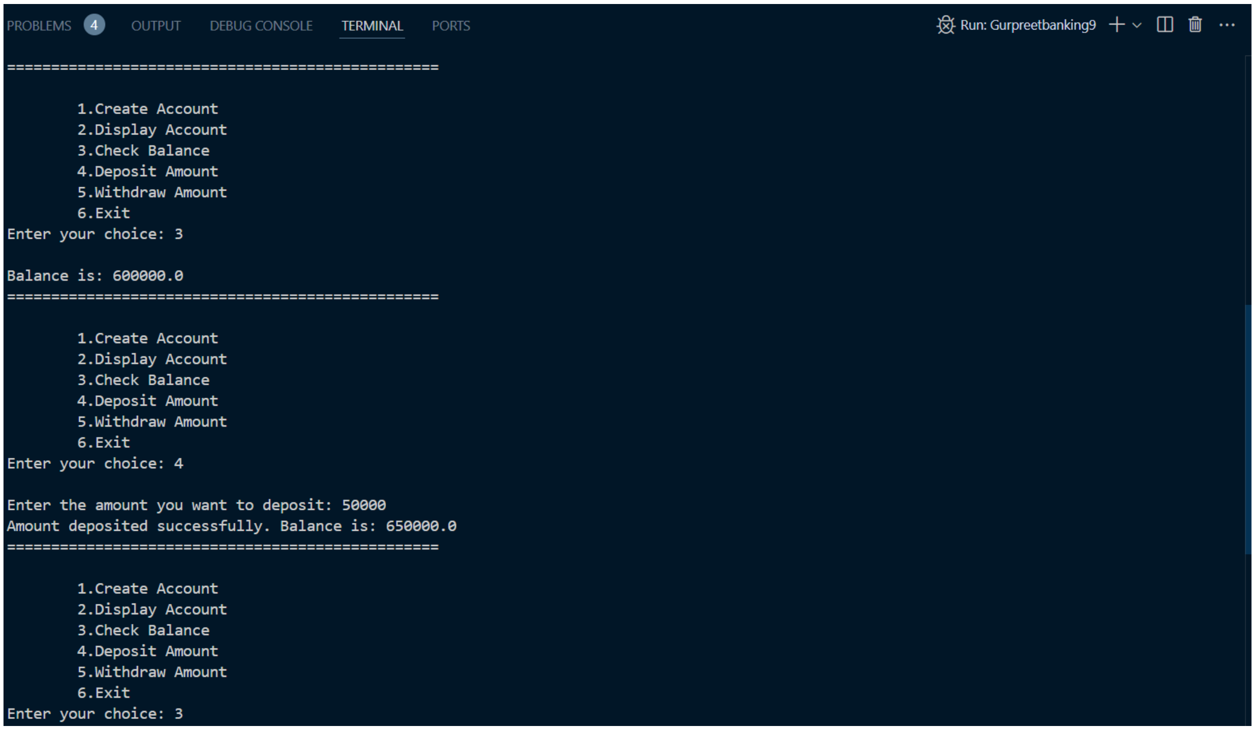
```
            return false; //returning false
        }
    }


//BANK CLASS
class Bank {
    public static Scanner sc=new Scanner(System.in);  //creating object of scanner class
    public SavingsAccount a=new SavingsAccount();    // creating object of SavingAccount class
    public Customer c=new Customer();        //creating object of Customer class

    public SavingsAccount createAccount(){  //method to create an Account

        System.out.print("Enter your name: ");    //printing on console
        String customername=sc.nextLine();    //taking customername as input from user
        c.setCustomerName(customername);    //calling setCustomerName method

        System.out.print("Enter your age: ");     //printing on console
        int customerage=sc.nextInt();         //taking customerage as input from user
        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: ");   //printing on console
        int accountid=sc.nextInt(); //taking accountid as input from user
        a.setAccountId(accountid); //calling setAccountId method

        System.out.print("Enter your account type: ");   //printing on console
        String accounttype=sc.next();  //taking accounttype as input from user
        a.setAccountType(accounttype); //calling setAccountType method

        System.out.print("Enter balance: ");  //printing on console
        double balance=sc.nextDouble();//taking balance as input from user
        a.setBalance(balance);//calling setBalance method

        System.out.print("Enter minimum balance: ");  //printing on console
        double minbalance=sc.nextDouble(); //taking minbalance as input from user
        a.setMinimumBalance(minbalance); //calling setMinimumBalance method


        return a;      //returning saving account
        }


    void getWithdrawAmount(){     //method to withdraw amount
        System.out.print("Enter the amount you want to withdraw: ");  //printing on console
        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){ //calling withdraw method
                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 Maulibanking9{
          public static void main(String[] args){
                    Scanner sc=new Scanner(System.in);  //creating object of scanner class
                    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();    //calling createAccount method

System.out.println("==================================================");
                                        break;
                              case 2:
                                        bm.displayAccountInformation();  //calling
displayAccountInformation method

System.out.println("==================================================");
                                        break;
                              case 3:
                                        bm.checkBalance();   //calling checkBalance method

System.out.println("==================================================");
                                        break;
                              case 4:
                                        System.out.print("Enter the amount you want to deposit: ");
                                        double amount=sc.nextDouble();
                                        bm.depositAmount(amount);       //calling depositAmount method

System.out.println("================================================");
                                        break;
                              case 5:
                                        bm.getWithdrawAmount();         //calling getWithdrawAmount
method

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);
                    }
 }
```

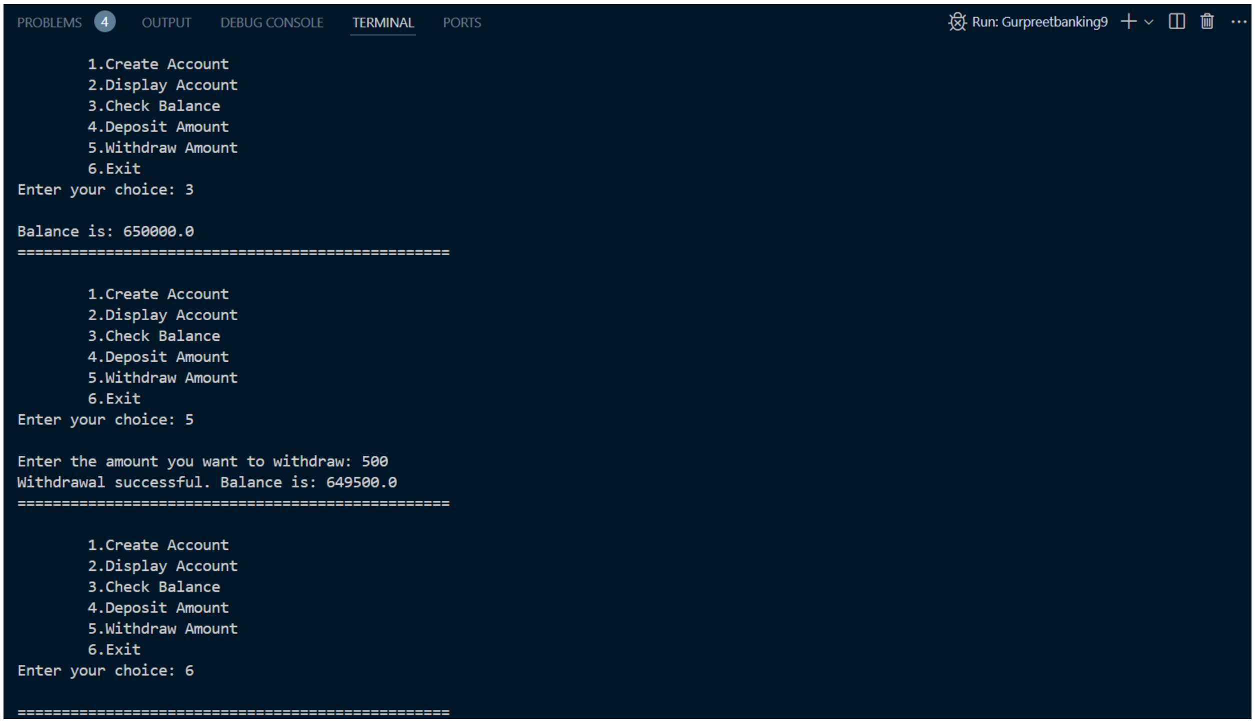
OUTPUT:



```
PS C:\potfolio\Portfolio-Website-Template-main\Portfolio Website>  & 'C:\Program Files\Java\jdk1.8.0_321\bin\java.exe' '-cp' 'C:\Users\Iron\
AppData\Roaming\Code\User\workspaceStorage\e281675927f958ebd4a88d08e351c955\redhat.java\jdt_ws\jdt.ls-java-project\bin' 'Gurpreetbanking9'

        1.Create Account
        2.Display Account
        3.Check Balance
        4.Deposit Amount
        5.Withdraw Amount
        6.Exit
Enter your choice: 1

Enter your name: gurpreet
Enter your age: 19
Enter your account Id: 1010
Enter your account type: saving
Enter balance: 600000
Enter minimum balance: 50000
=================================================

        1.Create Account
        2.Display Account
        3.Check Balance
        4.Deposit Amount
        5.Withdraw Amount
        6.Exit
Enter your choice: 2

Welcome gurpreet! Following are your account details:
Age :19
Account Id: 1010
Account Type: saving
Balance: 600000.0
Minimum balance: 50000.0
```



```
=================================================

        1.Create Account
        2.Display Account
        3.Check Balance
        4.Deposit Amount
        5.Withdraw Amount
        6.Exit
Enter your choice: 3

Balance is: 600000.0
=================================================

        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: 50000
Amount deposited successfully. Balance is: 650000.0
=================================================

        1.Create Account
        2.Display Account
        3.Check Balance
        4.Deposit Amount
        5.Withdraw Amount
        6.Exit
Enter your choice: 3
```



```
        1.Create Account
        2.Display Account
        3.Check Balance
        4.Deposit Amount
        5.Withdraw Amount
        6.Exit
Enter your choice: 3

Balance is: 650000.0
=================================================

        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: 500
Withdrawal successful. Balance is: 649500.0
=================================================

        1.Create Account
        2.Display Account
        3.Check Balance
        4.Deposit Amount
        5.Withdraw Amount
        6.Exit
Enter your choice: 6

=================================================
```