Name: Swaranjana Nayak **Reg. No: 19BCE0977**
Date: 17/03/2021 **Slot: L43 + L44**

Assessment 2

Q1. Write a program to demonstrate the knowledge of students in Inheritance.

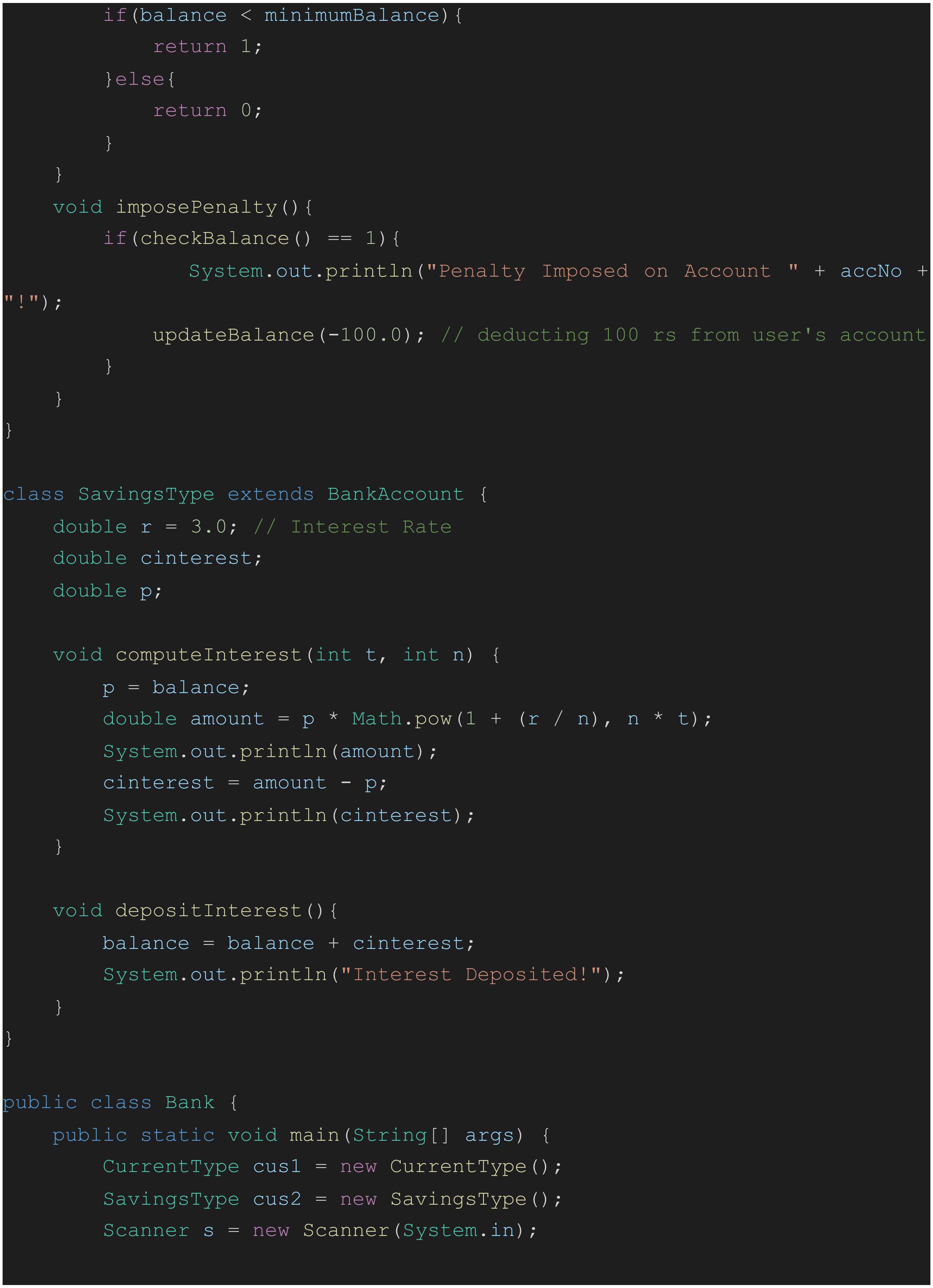
Assume that a bank maintains two kinds of accounts for customers, one called a Savings account and the other a Current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides a cheque book facility but no interest. Current account holders should maintain a minimum balance and if the balance falls below this level, a service charge is imposed.

Create a class account that stores customer name, account number, and type of account. From this derive the classes cur_acct and sav_acct to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks:

a) Accept deposit from a customer and update the balance.
b) Display the balance
c) Compute and deposit interest.
d) Permit withdrawal and update the balance.
e) Check for the minimum balance, impose a penalty, necessary, and update the balance.

Code:

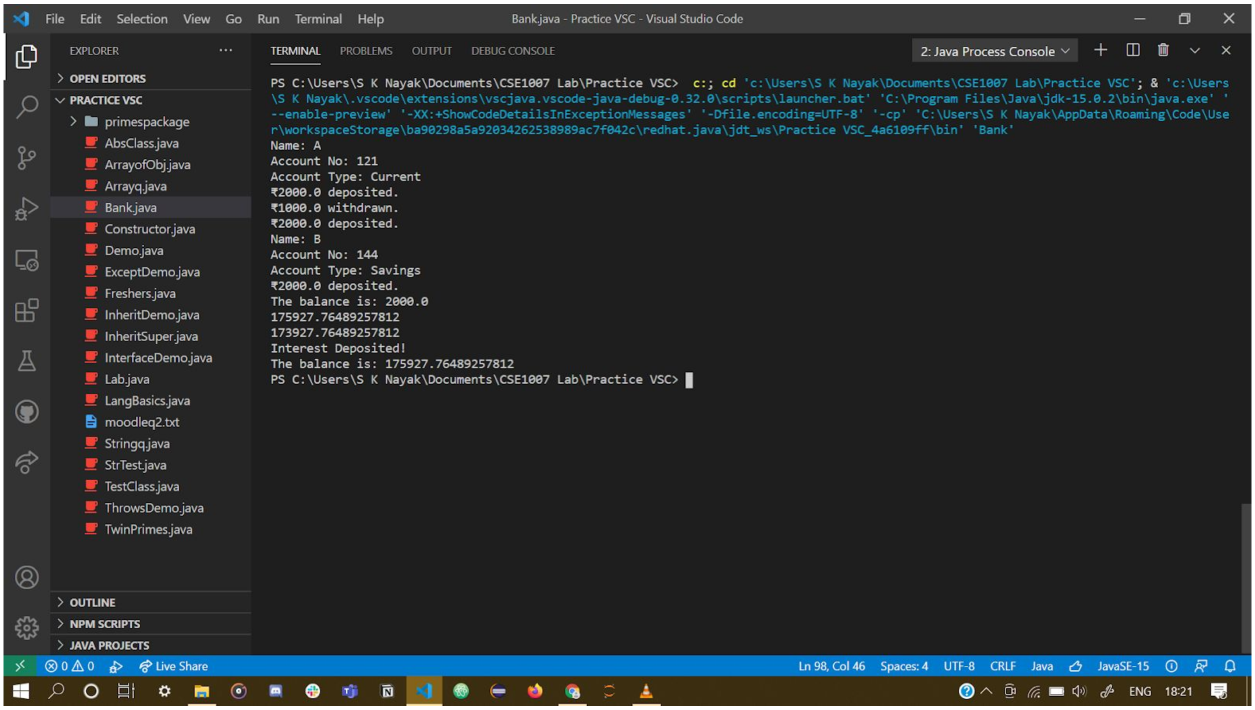
```java
import java.util.Scanner;
import java.lang.Math;
// 19BCE0977
class BankAccount {
    String customerName;
    int accNo;
    String accType;
    public double balance = 0.0;

    void getInput(Scanner s){
        System.out.print("Name: ");
        customerName = s.next();
        System.out.print("Account No: ");
        accNo = s.nextInt();
        System.out.print("Account Type: ");
        accType = s.next();
    }

    void displayAcc(){
        System.out.println(customerName);
        System.out.println(accNo);
        System.out.println(accType);
    }

    void withdraw(double amt){
        balance = balance - amt;
        System.out.println("₹" + amt + " withdrawn.");
    }

    void deposit(double amt){
        balance = balance + amt;
        System.out.println("₹" + amt + " deposited.");
    }

    void displayBalance(){
        System.out.println("The balance is: " + balance);
    }

    void updateBalance(double amt){
        balance = balance + amt;
        System.out.println("Balance Updated.");
    }
}

class CurrentType extends BankAccount {
    double serviceCharge;
    String chequeBook;
    double minimumBalance = 1000.0;

    void chequeBook(){
        System.out.println("Cheque Book associated with " + accNo + " is "
+ chequeBook);
    }
    int checkBalance(){
```



```
        if(balance < minimumBalance){
            return 1;
        }else{
            return 0;
        }
    }
    void imposePenalty(){
        if(checkBalance() == 1){
               System.out.println("Penalty Imposed on Account " + accNo +
"!");
            updateBalance(-100.0); // deducting 100 rs from user's account
        }
    }
}

class SavingsType extends BankAccount {
    double r = 3.0; // Interest Rate
    double cinterest;
    double p;

    void computeInterest(int t, int n) {
        p = balance;
        double amount = p * Math.pow(1 + (r / n), n * t);
        System.out.println(amount);
        cinterest = amount - p;
        System.out.println(cinterest);
    }

    void depositInterest(){
        balance = balance + cinterest;
        System.out.println("Interest Deposited!");
    }
}

public class Bank {
    public static void main(String[] args) {
        CurrentType cus1 = new CurrentType();
        SavingsType cus2 = new SavingsType();
        Scanner s = new Scanner(System.in);
```

```java
        cus1.getInput(s);
        // cus1.displayAcc();
        cus1.deposit(2000.0);
        cus1.withdraw(1000.0);
        cus1.imposePenalty();
        cus1.deposit(2000.0);

        cus2.getInput(s);
        // cus2.displayAcc();
        cus2.deposit(2000.0);
        cus2.displayBalance();
        cus2.computeInterest(2, 4);
        cus2.depositInterest();
        cus2.displayBalance();

        s.close();
    }
}
```

Output:



```
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC>  c:; cd 'c:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC'; & 'c:\Users
\S K Nayak\.vscode\extensions\vscjava.vscode-java-debug-0.32.0\scripts\launcher.bat' 'C:\Program Files\Java\jdk-15.0.2\bin\java.exe' '
--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\S K Nayak\AppData\Roaming\Code\Use
r\workspaceStorage\ba90298a5a92034262538989ac7f042c\redhat.java\jdt_ws\Practice VSC_4a6109ff\bin' 'Bank'
Name: A
Account No: 121
Account Type: Current
₹2000.0 deposited.
₹1000.0 withdrawn.
₹2000.0 deposited.
Name: B
Account No: 144
Account Type: Savings
₹2000.0 deposited.
The balance is: 2000.0
175927.76489257812
173927.76489257812
Interest Deposited!
The balance is: 175927.76489257812
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC> 
```

Q2. Write a program to demonstrate the knowledge of students in working with user-defined packages and sub-packages.

Eg., Within the package named ‘primespackage’, define a class Primes which includes a method checkForPrime() for checking if the given number is prime or not. Define another class named TwinPrimes outside of this package which will display all the pairs of prime numbers whose difference is 2. (Eg, within the range 1 to 10, all possible twin prime numbers are (3, 5), (5,7)). The TwinPrimes class should make use of the checkForPrime() method in the Primes class.

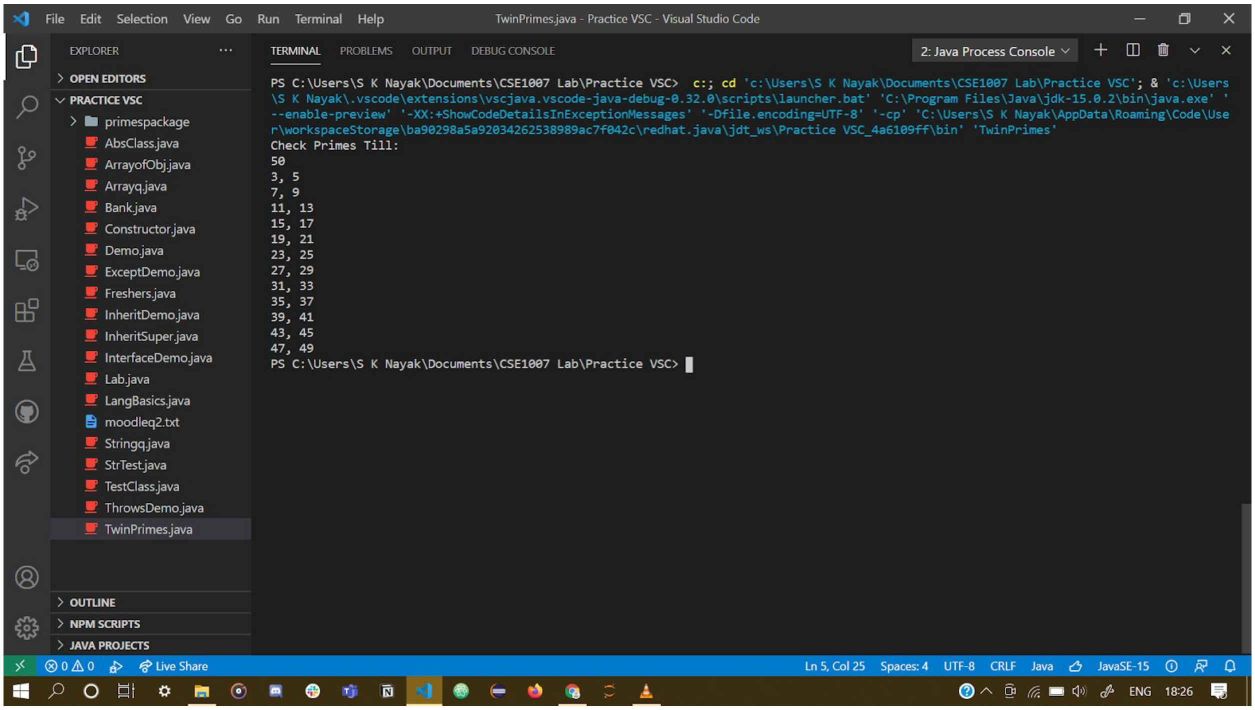
<u>Code:</u>
<u>primespackage/Primes.java</u>

```java
package primespackage;

public class Primes {
    public static boolean checkForPrime(int a){

        if(a < 4)
            return a > 1;

        for(int i = 2; i < a/2; i += 2) {
            if (a % i == 0)
                return false;
        }

        return true;
    }
}
```

<u>TwinPrimes.java</u>

```java
import java.util.*;
import primespackage.Primes;


public class TwinPrimes{
    public static void main(String[] args){

        Scanner in = new Scanner(System.in);
        System.out.println("Check Primes Till:");
        int n=in.nextInt();

        if (n<5){
            in.close();
            return;
        }

        for(int i = 3; i+1 < n; i += 4){
            if(Primes.checkForPrime(i) && Primes.checkForPrime(i+2))
                System.out.println(i + ", " + (i+2));

                           if(i+2  <  n  &&  Primes.checkForPrime(i+1)  &&
Primes.checkForPrime(i+3))
                System.out.println((i+1)+", "+(i+3));
        }

        in.close();
    }
}
```

Output:



```
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC>  c:; cd 'c:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC'; & 'c:\Users
\S K Nayak\.vscode\extensions\vscjava.vscode-java-debug-0.32.0\scripts\launcher.bat' 'C:\Program Files\Java\jdk-15.0.2\bin\java.exe' '
--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\S K Nayak\AppData\Roaming\Code\Use
r\workspaceStorage\ba90298a5a92034262538989ac7f042c\redhat.java\jdt_ws\Practice VSC_4a6109ff\bin' 'TwinPrimes'
Check Primes Till:
50
3, 5
7, 9
11, 13
15, 17
19, 21
23, 25
27, 29
31, 33
35, 37
39, 41
43, 45
47, 49
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC> 
```

Q3. Write a program to demonstrate the knowledge of students in Java Exception handling.

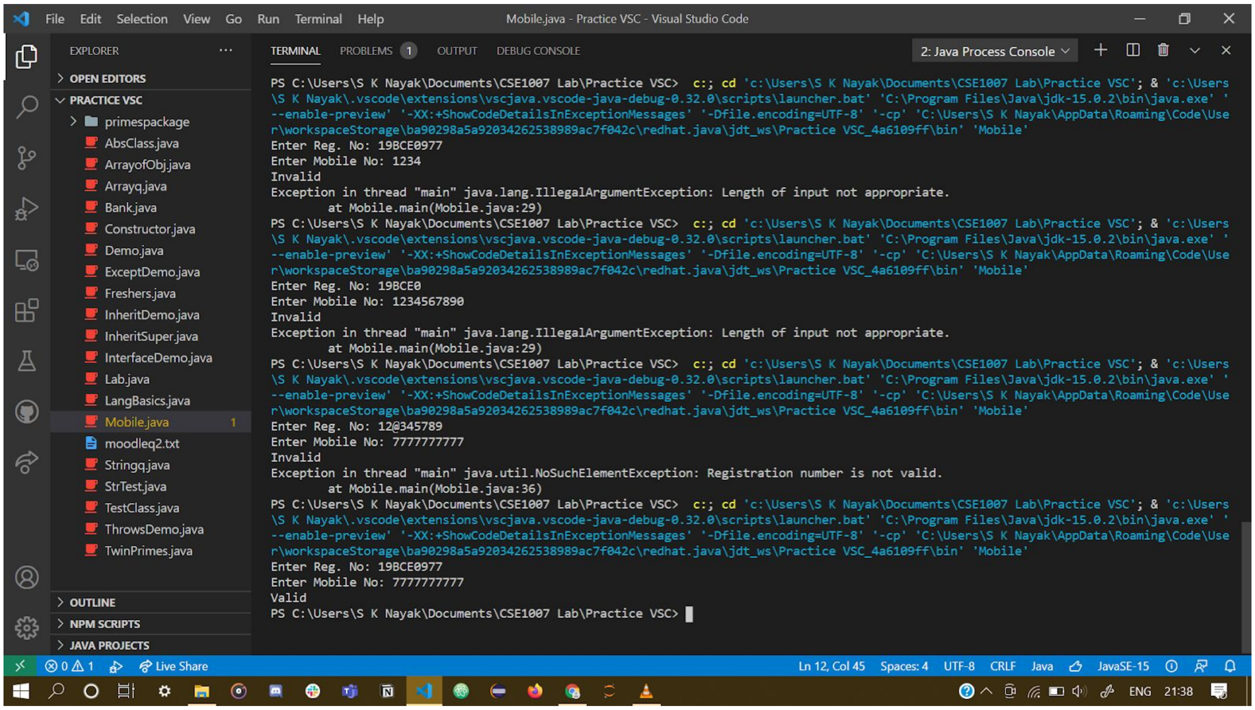
Eg., Read the Register Number and Mobile Number of a student. If the Register Number does not contain exactly 9 characters or if the Mobile Number does not contain exactly 10 characters, throw an IllegalArgumentException. If the Mobile Number contains any character other than a digit, raise a NumberFormatException. If the Register Number contains any character other than digits and alphabets, throw a NoSuchElementException. If they are valid, print the message 'valid' else 'invalid'

Code:

```
import java.util.NoSuchElementException;
import java.util.Scanner;
import java.lang.IllegalArgumentException;
import java.util.NoSuchElementException;

public class Mobile {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        String regno;
        long mobile;

        System.out.print("Enter Reg. No: ");
        regno = s.nextLine();

        System.out.print("Enter Mobile No: ");
        mobile = s.nextLong();
        long m = mobile;
        int digits = 0;

        s.close();

        while(m > 0){
            m /= 10;
            digits = digits + 1;
        }

       if(regno.length() != 9 || digits != 10){
            System.out.println("Invalid");
```

```
                  throw new IllegalArgumentException("Length of input not
appropriate.");
        }

        String regex = "^(?=.*[a-zA-Z])(?=.*[0-9])[A-Za-z0-9]+$";

        if(!regno.matches(regex)){
            System.out.println("Invalid");
              throw new NoSuchElementException("Registration number is not
valid.");
        }

        System.out.println("Valid");
    }
}
```

Output:



```
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC>  c:; cd 'c:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC'; & 'c:\Users
\S K Nayak\.vscode\extensions\vscjava.vscode-java-debug-0.32.0\scripts\launcher.bat' 'C:\Program Files\Java\jdk-15.0.2\bin\java.exe' '
--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\S K Nayak\AppData\Roaming\Code\Use
r\workspaceStorage\ba90298a5a92034262538989ac7f042c\redhat.java\jdt_ws\Practice VSC_4a6109ff\bin' 'Mobile'
Enter Reg. No: 19BCE0977
Enter Mobile No: 1234
Invalid
Exception in thread "main" java.lang.IllegalArgumentException: Length of input not appropriate.
        at Mobile.main(Mobile.java:29)
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC>  c:; cd 'c:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC'; & 'c:\Users
\S K Nayak\.vscode\extensions\vscjava.vscode-java-debug-0.32.0\scripts\launcher.bat' 'C:\Program Files\Java\jdk-15.0.2\bin\java.exe' '
--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\S K Nayak\AppData\Roaming\Code\Use
r\workspaceStorage\ba90298a5a92034262538989ac7f042c\redhat.java\jdt_ws\Practice VSC_4a6109ff\bin' 'Mobile'
Enter Reg. No: 19BCE0
Enter Mobile No: 1234567890
Invalid
Exception in thread "main" java.lang.IllegalArgumentException: Length of input not appropriate.
        at Mobile.main(Mobile.java:29)
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC>  c:; cd 'c:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC'; & 'c:\Users
\S K Nayak\.vscode\extensions\vscjava.vscode-java-debug-0.32.0\scripts\launcher.bat' 'C:\Program Files\Java\jdk-15.0.2\bin\java.exe' '
--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\S K Nayak\AppData\Roaming\Code\Use
r\workspaceStorage\ba90298a5a92034262538989ac7f042c\redhat.java\jdt_ws\Practice VSC_4a6109ff\bin' 'Mobile'
Enter Reg. No: 12@345789
Enter Mobile No: 7777777777
Invalid
Exception in thread "main" java.util.NoSuchElementException: Registration number is not valid.
        at Mobile.main(Mobile.java:36)
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC>  c:; cd 'c:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC'; & 'c:\Users
\S K Nayak\.vscode\extensions\vscjava.vscode-java-debug-0.32.0\scripts\launcher.bat' 'C:\Program Files\Java\jdk-15.0.2\bin\java.exe' '
--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\Users\S K Nayak\AppData\Roaming\Code\Use
r\workspaceStorage\ba90298a5a92034262538989ac7f042c\redhat.java\jdt_ws\Practice VSC_4a6109ff\bin' 'Mobile'
Enter Reg. No: 19BCE0977
Enter Mobile No: 7777777777
Valid
PS C:\Users\S K Nayak\Documents\CSE1007 Lab\Practice VSC> 
```