PRAKHAR LOHIYA
16BCE0721
LAB ASSESSMENT 2

Ques 1: String Handling

Write a program to demonstrate the knowledge of students in String handling.

Eg., Write a program to read a chemical equation and find out the count of the reactants and the products. Also display the count of the number of molecules of each reactant and product.

Eg., For the equation, 2NaOH + H2SO4 -> Na2SO4+ 2H2O, the O/P should be as follows.

Reactants are 2 moles of NaOH, 1 mole of H2SO4.

Products are 1 mole of Na2SO4 and 2 moles of H2O.

CODE :

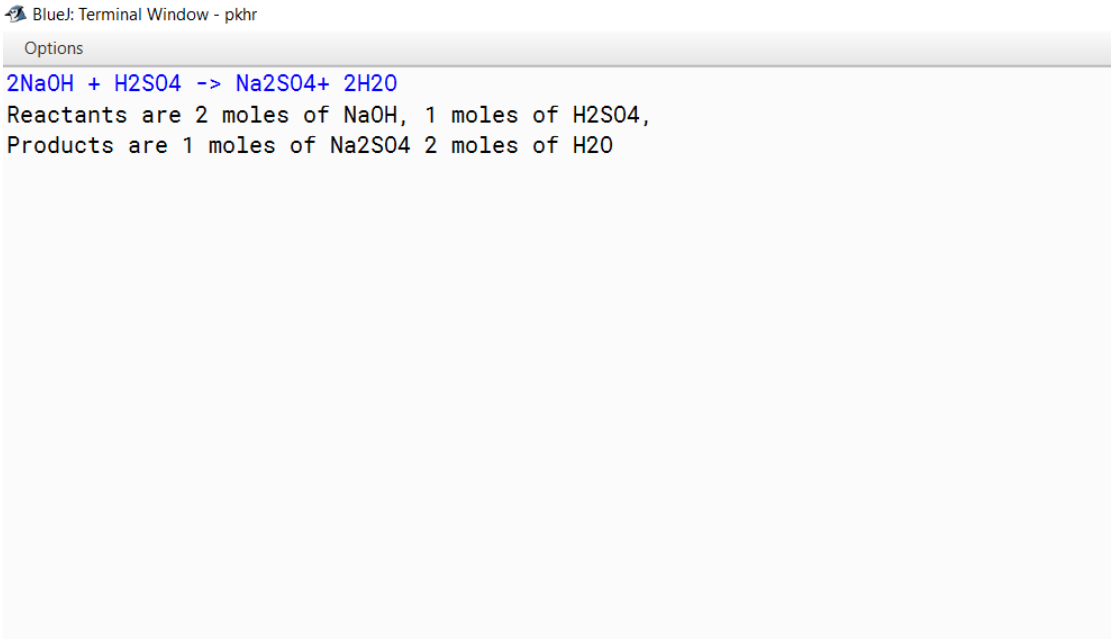
```
import java.util.Scanner;


class Equation{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        String s = sc.nextLine();
        String eq[] = s.split("[//->]+");
        String eqr[] = eq[0].split("//+");
        String eqp[] = eq[1].split("//+");

        for(String k: eqp)
            if(Character.isDigit(charAt(k[0]))
                System.out.println(k[0]+" moles of "+k.substring(k[1]);
            else
                System.out.println("1 moles of "+k);
```

```
        for(String k: eqr)
            if(Character.isDigit(charAt(k[0]))
                System.out.println(k[0]+" moles of "+k.substring(k[1]);
            else
                System.out.println("1 moles of "+k);
    }
}
```

OUTPUT:



Ques 2: Inheritance

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

Eg: Assume that a bank maintains two kinds of accounts for customers, one called as savings account and the other as current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides 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 penalty, necessary, and update the balance.

CODE :

```
import java.util.Scanner;


class Account {
    private String name, type;
    private int number;
    int balance;
    Account() {}
    Account(String name, String type, int number, int balance) {
        this.name = name;
        this.type = type;
        this.number = number;
        this.balance = balance;
```

```
    }
    void acceptDeposit(int amount) {
        balance += amount;
    }
    void displayBalance() {
        System.out.println();
        System.out.print("Name = " + name);
        System.out.println();
        System.out.print("Number = " + number);
        System.out.println();
        System.out.print("Account Type = " + type);
        System.out.println();
        System.out.print("Your balance = " + balance);
        System.out.println();

    }
    void withdrawal(int amount) {
        if (balance-amount >= 0)
            balance -= amount;
        else
            System.out.println("Not enough balance");
    }
}

class CurrentAccount extends Account {
    void computeFine(int minimum, Account account) {
        if (account.balance < minimum) {
            System.out.println("Fine levied = " + (0.05*account.balance));
            account.balance -= 0.05*account.balance;
        }
```

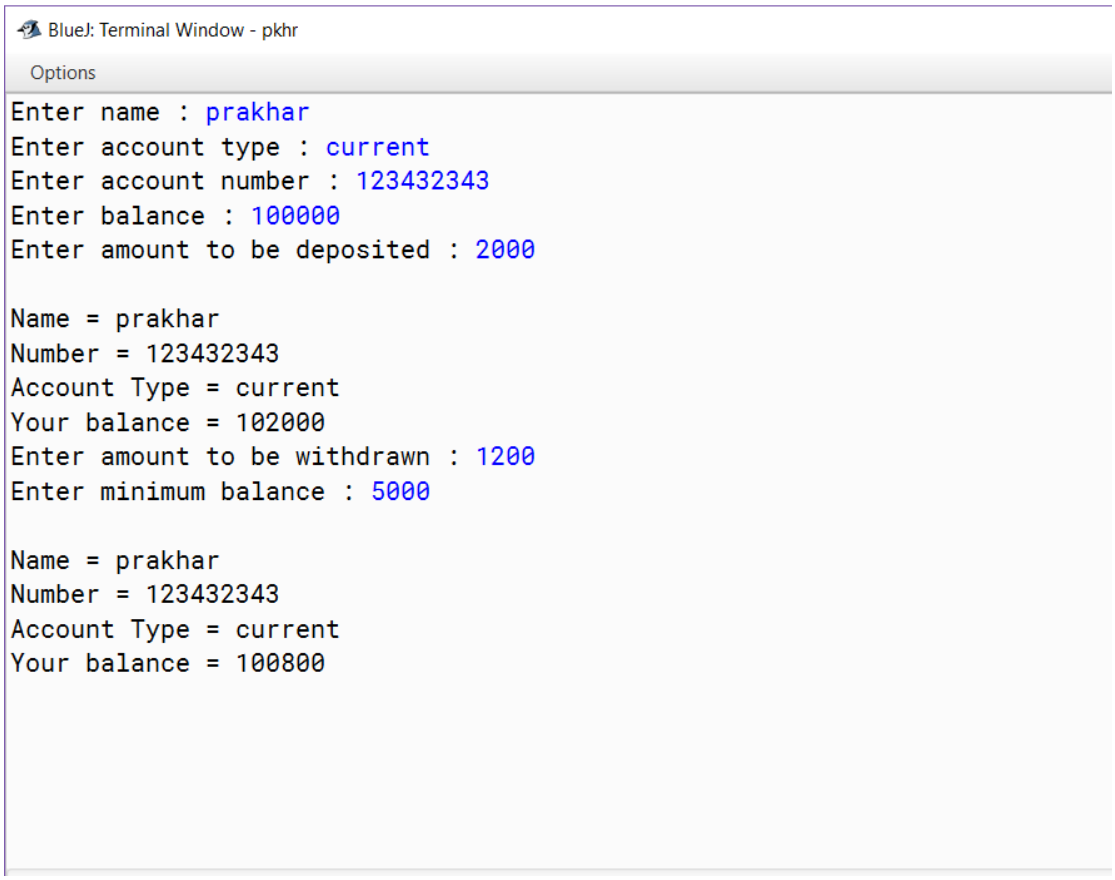
```
    }
}

class SavingsAccount extends Account {
    void computeInterest(int time, double rate, Account account) {
        int principal = account.balance;
        for (int i=0; i<time; i++) {
            principal += principal*rate*time;
        }
        System.out.print("Interest = " + (principal-account.balance));
        System.out.println();
        account.balance = principal;
    }
}

public class Bank {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter name : ");
        String name = in.nextLine();
        System.out.print("Enter account type : ");
        String type = in.nextLine();
        System.out.print("Enter account number : ");
        int number = in.nextInt();
        System.out.print("Enter balance : ");
        int balance = in.nextInt();
        Account account = new Account(name, type, number, balance);
        CurrentAccount currentAccount = new CurrentAccount();
        SavingsAccount savingsAccount = new SavingsAccount();
        System.out.print("Enter amount to be deposited : ");
```

```
        int amount = in.nextInt();
        account.acceptDeposit(amount);
        account.displayBalance();
        if (type.equalsIgnoreCase("savings")) {
            System.out.print("Enter time : ");
            int time = in.nextInt();
            System.out.print("Enter rate : ");
            double rate = in.nextDouble();
            savingsAccount.computeInterest(time, rate, account);
            account.displayBalance();
        }
        System.out.print("Enter amount to be withdrawn : ");
        amount = in.nextInt();
        account.withdrawal(amount);
        if (type.equalsIgnoreCase("current")) {
            System.out.print("Enter minimum balance : ");
            int minimum = in.nextInt();
            currentAccount.computeFine(minimum, account);
            account.displayBalance();
        }
    }
}
```

OUTPUT:


BlueJ: Terminal Window - pkhr
Options
Enter name : prakhar
Enter account type : current
Enter account number : 123432343
Enter balance : 100000
Enter amount to be deposited : 2000
Name = prakhar
Number = 123432343
Account Type = current
Your balance = 102000
Enter amount to be withdrawn : 1200
Enter minimum balance : 5000
Name = prakhar
Number = 123432343
Account Type = current
Your balance = 100800

Ques 3: User Defined Packages

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.

CODE :

Primes.java in primePackage directory

```
package primePackage;


public class Primes {
   public boolean checkForPrime(int x) {
      for (int i=2; i<x; i++)
         if (x%i==0) {
            return false;
         }
      return true;
   }
}
```

TwinPrimes.java

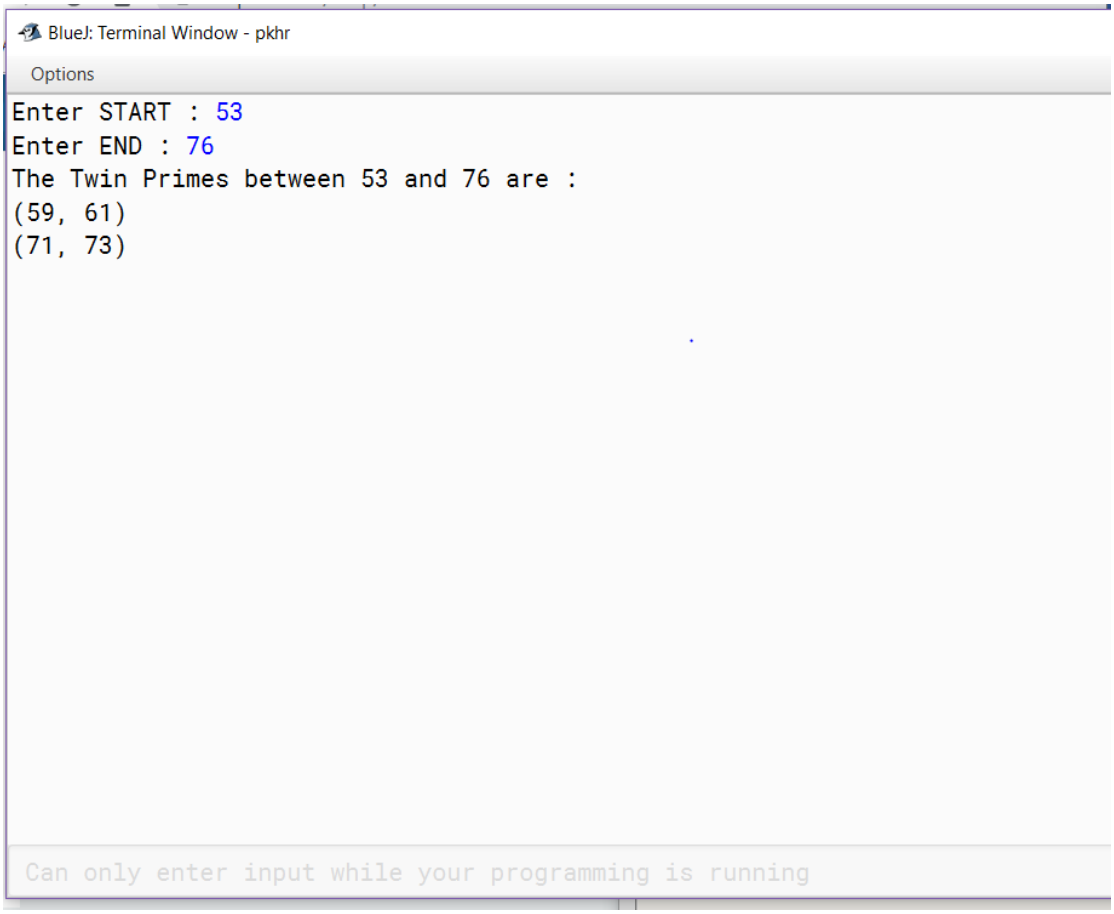
```
import java.util.Scanner;
import primePackage.Primes;


public class TwinPrimes {
   public static void main(String []args) {
      Primes obj = new Primes();
      Scanner in = new Scanner(System.in);
```

```
        System.out.print("Enter START : ");
        int start = in.nextInt();
        System.out.print("Enter END : ");
        int end = in.nextInt();

        System.out.println("The Twin Primes between " + start + " and " + end + " are :
");
        for (int i=start; i<=end-2; i++) {
            if (i==1)
                continue;
            if (obj.checkForPrime(i) && obj.checkForPrime(i+2))
                System.out.println("(" + i + ", " + (i+2) + ")");
        }
    }
}
```

OUTPUT:


BlueJ: Terminal Window - pkhr
Options
Enter START : 53
Enter END : 76
The Twin Primes between 53 and 76 are :
(59, 61)
(71, 73)
Can only enter input while your programming is running

Ques 4: Exception Handling

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.Scanner;
import java.util.NoSuchElementException;


public class ExceptionalHandling {
   public static void main(String []args) {
      try {
         Scanner in = new Scanner(System.in);
         System.out.print("Enter you Registration Number : ");
         String regNo = in.nextLine();
         System.out.print("Enter you Mobile Number : ");
         String mobNo = in.nextLine();


         if (mobNo.length() != 10 || regNo.length()!=9)
            throw new IllegalArgumentException("Mobile Number should be of 10 digits");
         else {
            for (int i=0; i<mobNo.length(); i++)
               if(!Character.isDigit(mobNo.charAt(i)))
                  throw new NumberFormatException("Mobile Number should have only Digits");
```

```
            for (int i=0; i<regNo.length(); i++)
                if(!Character.isDigit(regNo.charAt(i))
&& !Character.isLetter(regNo.charAt(i)))
                    throw new NoSuchElementException("Registration number should be
AlphaNumeric only");


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


    } catch (Exception e) {
        System.out.print("\n\n"+e.getMessage()+"\n");
    }
  }
}
```

OUTPUT:

```
Enter you Registration Number : 16BCE0721
Enter you Mobile Number : 9589776209
Valid Credentials
Enter you Registration Number : 12@3ED123
Enter you Mobile Number : 23132W1212


Mobile Number should have only Digits
```

Can only enter input while your programming is running