

JOURNAL-02

1. Define a class to represent a bank account. Include the following members:

Data Members:

- a.Name of the Depositor
- b.Account Number
- c.Type of Account
- d.Balance amount in the account

Data Methods:

- a.To assign initial values
- b.To deposit an amount
- c.To withdraw an amount
- d.To display name and balance.

```
import java.io.*;
```

```
import java.util.*;
```

```
import java.util.Scanner;
```

```
import java.util.Random;
```

```
class Bank
{
    public String nameOfDepositor;
    public int accNumber;
    public String accType;
    public double balanceAmount;

    public void assignValues(String nameOfDepositor, String accType, double
balanceAmount)
    {
        this.nameOfDepositor=nameOfDepositor;
        this.accType=accType;
        this.balanceAmount=balanceAmount;
        Random random = new Random();
        this.accNumber=random.nextInt(1000000);
        System.out.println("Your new account number is: "+accNumber);
    }

    public void depositAmount(double amount)
    {
        if(accNumber==0)

            System.out.println("!You don't have bank account to
deposit\nNote:please assign values to create an account");
        else
```

```
{
    balanceAmount+=amount;

    System.out.println("Amount deposited successfully...");
}

}

public void withdrawAmount(double amount)
{
    if(accNumber==0)

        System.out.println("!You don't have bank account to credit\nNote:please
assign values to create an account");

    else if(balanceAmount>amount)
    {
        balanceAmount-=amount;

        System.out.println("Amount credited successfully...");
    }
    else

        System.out.println("! Insufficient balance");
}

public void displayDetails()
{
    if(accNumber==0)
```

```
        System.out.println("!You don't have bank account\nNote:please assign  
values to create an account");
```

```
    else
```

```
    {
```

```
        System.out.println("Name of the Depositor: "+nameOfDepositor);
```

```
        System.out.println("Balance amount in the account: "+balanceAmount);
```

```
    }
```

```
}
```

```
public void getInput()
```

```
{
```

```
    System.out.println("How can i help you?");
```

```
    System.out.println("1. Open account");
```

```
    System.out.println("2. Deposit amount");
```

```
    System.out.println("3. Withdraw amount");
```

```
    System.out.println("4. Account details");
```

```
    System.out.println("5. Exit");
```

```
    System.out.print("Please Enter Your choose : ");
```

```
}
```

```
}
```

```
class PRG_01
```

```
{
```

```
    public static void main(String[] s) throws IOException
```

```
{  
    System.out.println("::::::::: WELCOME TO BANK OF BARODA ::::::::::");  
    Bank newAccount=new Bank();  
    Scanner scan=new Scanner(System.in);  
    boolean process=true;  
    int continueState=0;  
    while(continueState!=5)  
    {  
        newAccount.getInput();  
        int currentProcess=scan.nextInt();  
  
        if(currentProcess==1)  
        {  
  
            System.out.println("*****  
*****");  
            System.out.print("Enter your name: ");  
            String nameOfDepositor=scan.next();  
            System.out.print("Enter your account type: ");  
            String accType=scan.next();  
            System.out.print("Enter your opening balance: ");  
            double balanceAmount=scan.nextDouble();
```

```
        newAccount.assignValues(nameOfDepositor, accType, balanceAmount);
    }

    else if(currentProcess==2)

    {

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

        System.out.print("Enter amount to deposit: ");

        newAccount.depositAmount(scan.nextDouble());

    }

    else if(currentProcess==3)

    {

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

        System.out.println("Your current Balance in the account:
        "+newAccount.balanceAmount);

        System.out.print("Enter amount to withdraw: ");

        newAccount.withdrawAmount(scan.nextDouble());

    }

    else if(currentProcess==4)

    {
```

[illegible]

Output :

[illegible]

2. Write a program to print Floyd's triangle where n is command line input.

```
1
2 3
4 5 6
7 8 9 10
```

```
class PRG_02
{
    static void printFloydTriangle(int n)
    {
        int i, j, val = 1;
        for (i = 1; i <= n; i++)
        {
            for (j = 1; j <= i; j++)
            {
                System.out.print(val + " ");
                val++;
            }
            System.out.println();
        }
    }
    public static void main(String[] args)
    {
        int i= Integer.parseInt(args[0]);
        printFloydTriangle(i);
    }
}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_02.java
```

```
E:\Java\JOURNAL-2>java PRG_02 5
```

```
1
```

```
2 3
```

```
4 5 6
```

```
7 8 9 10
```

```
11 12 13 14 15
```

```
E:\Java\JOURNAL-2>
```

3. Design a class Cricketer having data member name and a number of matches and appropriate member function to set the values. Derived two classes Batsman and Bowler from cricketer class with data member total number of runs and wickets respectively. Batsman class is having method to calculate average wicket. Write a program to create two objects and display information of one batsman and bowler along with average run and wicket.

```
import java.util.Scanner;

class Cricketer
{
    public String cname;

    public int nom;

    public void setDataMain()
    {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the name of Cricketer: ");

        cname = sc.nextLine();

        Scanner sc2 = new Scanner(System.in);

        System.out.print("Enter the Number of matches of Cricketer: ");

        nom = sc2.nextInt();
    }

    public void displayDataMain()
    {
        System.out.println("Name " +cname);

        System.out.println("Matches " +nom);
    }
}
```

```
class Batsman extends Cricketer
{
    public int total_run;
    public float average;
    public void setData()
    {
        Scanner sc4 = new Scanner(System.in);
        System.out.print("\nEnter the Total Number of Runs: ");
        total_run = sc4.nextInt();
    }
    public void displayData()
    {
        System.out.println("Total Runs "+total_run);
    }
}
```

```
class Bowler extends Cricketer
{
    public int wickets;
```

```
public float average;

public void setData()
{
    Scanner sc3 = new Scanner(System.in);

    System.out.print("Enter the number of wickets: ");

    wickets = sc3.nextInt();
}

public void displayData()
{
    System.out.println("Wickets "+wickets);
}
}

public class PRG_03
{
    public static void main(String[] args)
    {
        Bowler bowl = new Bowler();

        Batsman bat = new Batsman();

        Cricketer cal = new Cricketer();
    }
}
```

```
        cal.setDataMain();

        bat.setData();

        bowl.setData();

        cal.displayDataMain();

        bat.displayData();

        bowl.displayData();

        bowl.average = (float) bowl.wickets/cal.nom;

        bat.average = (float) bat.total_run/cal.nom;

        System.out.println("Average_Run's: "+bat.average);

        System.out.println("Average_Wicket's: "+bowl.average);

    }

}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_03.java

E:\Java\JOURNAL-2>java PRG_03
Enter the name of Cricketer: Dhoni
Enter the Number of matches of Cricketer: 300

Enter the Total Number of Runs: 4000
Enter the number of wickets: 650
Name Dhoni
Matches 300
Total Runs 4000
Wickets 650
Average_Run's: 13.333333
Average_Wicket's: 2.1666667

E:\Java\JOURNAL-2>
```

4. Write a program that will accept two strings or two numbers from command line and create overloaded method that add these two numbers or concatenate two strings.

```
import java.io.*;

class PRG_04
{
    static boolean isNumber(String s)
    {
        for(int i=0;i<s.length();i++)
            if(Character.isDigit(s.charAt(i))==false)
                return false;

        return true;
    }

    void add(int a, int b)
    {
        System.out.println("Result is : "+(a+b)) ;
    }

    void add(String a, String b)
    {
        System.out.println("Result is : "+(a+b));
    }

    public static void main(String[] args)
    {
```



```
PRG_04 obj = new PRG_04();  
if(isNumber(args[0])&&isNumber(args[1]))  
{  
    int a=Integer.parseInt(args[0]);  
    int b=Integer.parseInt(args[1]);  
    obj.add(a,b);  
}  
else  
    obj.add(args[0],args[1]);  
}  
}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_04.java
```

```
E:\Java\JOURNAL-2>java PRG_04 1 2
```

```
Result is : 3
```

```
E:\Java\JOURNAL-2>java PRG_04 laxman sirvi
```

```
Result is : laxmansirvi
```

```
E:\Java\JOURNAL-2>_
```

5. Write a program that accept a number from command line and check whether it is palindrome or not.

```
public class PRG_05 {  
  
    public static void main(String args[])  
    {  
  
        int n = Integer.parseInt(args[0]);  
  
        int sum = 0, r;  
  
        int temp = n;  
  
        while(n>0)  
        {
```

```
        r = n % 10;

        sum = (sum*10)+r;

        n = n/10;
    }

    if(temp==sum)

        System.out.println("It is a Palindrome No.");

    else

        System.out.println("It's Not a palindrome No.");

}

}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_05.java
```

```
E:\Java\JOURNAL-2>java PRG_05 131  
It is a Palindrome No.
```

```
E:\Java\JOURNAL-2>java PRG_05 125  
It's Not a palindrome No.
```

```
E:\Java\JOURNAL-2>_
```

6. Write a program that will accept a string from command line and arrange all the characters in alphabetical order.

E.g. input- computer

output-cemoprut

```
public class PRG_06  
{  
    public static void main(String args[])  
    {  
  
        String str = args[0];
```

```
str = str.toLowerCase();  
  
int len = str.length();  
  
String sortedStr = ""; //Empty String  
for (char ch = 'a'; ch <= 'z'; ch++) {  
    for (int i = 0; i < len; i++) {  
        char strCh = str.charAt(i);  
        if (ch == strCh) {  
            sortedStr += strCh;  
        }  
    }  
}  
  
System.out.println("Alphabetical order : "+sortedStr);  
  
}  
}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_06.java
```

```
E:\Java\JOURNAL-2>java PRG_06 computer  
Alphabetical order : cemopr tu
```

```
E:\Java\JOURNAL-2>
```

7. Write a program to create interface Area. Create three classes called rectangle, triangle and square calculate areas respectively.

```
import java.util.Scanner;
```

```
interface area
```

```
{
```

```
    double calc(double x,double y);
```

```
}
```

```
class rectangle implements area
```

```
{
```

```
    public double calc(double x,double y)
```

```
    {
```

```
        return(x*y);  
    }  
}
```

class triangle implements area

```
{  
    public double calc(double x,double y)  
    {  
        return((x*y)/2);  
    }  
}
```

class square implements area

```
{  
    public double calc(double x,double y)  
    {  
        return(x*x);  
    }  
}
```

class PRG_07

```
{  
    public static void main(String arg[])  
    {
```

```
int p,q;

Scanner in = new Scanner(System.in);

rectangle r = new rectangle();

triangle c = new triangle();

square s = new square();

area a;

a = r;

System.out.print("\nEnter hight of Rectangle : ");

p=in.nextInt();

System.out.print("Enter width of Rectangle : ");

q=in.nextInt();

System.out.println("\nArea of Rectangle is : " +a.calc(p,q));

a = c;

System.out.print("\nEnter hight of Triangle : ");

p=in.nextInt();

System.out.print("Enter Breath of Triangle : ");

q=in.nextInt();

System.out.println("\nArea of Triangle is : " +a.calc(p,q));

a = s;

System.out.print("\nEnter Side of Square : ");

p=in.nextInt();

System.out.println("\nArea of Square is : " +a.calc(p,p));
```



```
}  
}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_07.java  
  
E:\Java\JOURNAL-2>java PRG_07  
  
Enter hight of Rectangle : 10  
Enter width of Rectangle : 20  
  
Area of Rectangle is : 200.0  
  
Enter hight of Triangle : 10  
Enter Breath of Triangle : 10  
  
Area of Triangle is : 50.0  
  
Enter Side of Square : 5  
  
Area of Square is : 25.0  
  
E:\Java\JOURNAL-2>
```

8. Write a program that will accept a number from command line and raise a user defined exception if the number consists of odd number of digits.

```
class OddNumberOfDigitsException extends Exception {  
    public OddNumberOfDigitsException(String message) {  
        super(message);  
    }  
}  
  
public class PRG_08 {  
    public static void main(String[] args) {  
        int numberString = Integer.parseInt(args[0]);  
        int temp=numberString;  
        int x;  
        try {  
            while(numberString>0){  
                x=numberString%10;  
                numberString=numberString/10;  
                if (x % 2 != 0) {  
                    throw new OddNumberOfDigitsException("The  
number has an odd number of digits");  
                }  
            }  
        }  
        catch (ArrayIndexOutOfBoundsException e) {
```

```
        System.out.println("Please provide a number as a command-line  
argument.");  
    } catch (OddNumberOfDigitsException e) {  
        System.out.println(e.getMessage());  
    }  
}  
}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_08.java  
  
E:\Java\JOURNAL-2>java PRG_08 24688  
  
E:\Java\JOURNAL-2>java PRG_08 25286  
The number has an odd number of digits  
  
E:\Java\JOURNAL-2>
```

9. Write a java application which accepts 10 names of student and their age. Sort names and age in descending order. (Using Array)

```
import java.util.Arrays;
```

```
import java.util.Scanner;

public class PRG_09 {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        String[] names = new String[10];
        int[] ages = new int[10];

        for (int i = 0; i < 10; i++) {
            System.out.print("Enter name of student " + (i + 1) + ": ");
            names[i] = scanner.nextLine();
            System.out.print("Enter age of student " + (i + 1) + ": ");
            ages[i] = scanner.nextInt();
            scanner.nextLine();
        }

        while (true) {
            System.out.println("\nSelect an option:");
            System.out.println("1. Sort via Name.");
            System.out.println("2. Sort via Age.");
            System.out.println("3. Exit");
            System.out.print("\nSelect Your Choice : ");
            int choice = scanner.nextInt();
            scanner.nextLine();

            switch (choice) {
                case 1:
                    for (int i = 0; i < 10; i++) {
                        for (int j = i + 1; j < 10; j++) {
                            if (names[i].compareToIgnoreCase(names[j]) < 0) {
                                String tempName = names[i];
                                names[i] = names[j];
                                names[j] = tempName;
                            }
                        }
                    }
                    break;
            }
        }
    }
}
```

```

        int tempAge = ages[i];
        ages[i] = ages[j];
        ages[j] = tempAge;
    }
}
}
System.out.println("\nSorted Names in Descending Order:");
for (int i = 0; i < 10; i++) {
    System.out.println(names[i] + " - " + ages[i]);
}
break;
case 2:
    for (int i = 0; i < 10; i++) {
        for (int j = i + 1; j < 10; j++) {
            if (ages[i] < (ages[j])) {
                int tempage = ages[i];
                ages[i] = ages[j];
                ages[j] = tempage;
                String tempname = names[i];
                names[i] = names[j];
                names[j] = tempname;
            }
        }
    }
    System.out.println("\nSorted Ages in Descending Order:");
    for (int i = 0; i < 10; i++) {
        System.out.println(ages[i] + " - " + names[i]);
    }
    break;
case 3:
    System.out.println("Exiting program...");
    System.exit(0);
    break;
default:

```

```
System.out.println("Invalid choice. Try again.");
```

```
    }  
    }  
    }  
}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_09.java  
  
E:\Java\JOURNAL-2>java PRG_09  
Enter name of student 1: laxman  
Enter age of student 1: 20  
Enter name of student 2: jamu  
Enter age of student 2: 16  
Enter name of student 3: ashish  
Enter age of student 3: 19  
Enter name of student 4: abhishek  
Enter age of student 4: 22  
Enter name of student 5: ambulance  
Enter age of student 5: 18  
Enter name of student 6: 32gb  
Enter age of student 6: 19  
Enter name of student 7: cow  
Enter age of student 7: 14  
Enter name of student 8: buffalo  
Enter age of student 8: 80  
Enter name of student 9: ankit  
Enter age of student 9: 21  
Enter name of student 10: shanti  
Enter age of student 10: 36  
  
Select an option:  
1. Sort via Name.  
2. Sort via Age.  
3. Exit
```

```
Select Your Choice : 1

Sorted Names in Descending Order:
shanti - 36
laxman - 20
jamu - 16
cow - 14
buffalo - 80
ashish - 19
ankit - 21
ambulance - 18
abhishek - 22
32gb - 19

Select an option:
1. Sort via Name.
2. Sort via Age.
3. Exit

Select Your Choice : 2

Sorted Ages in Descending Order:
80 - buffalo
36 - shanti
22 - abhishek
21 - ankit
20 - laxman
19 - ashish
19 - 32gb
18 - ambulance
16 - jamu
14 - cow

Select an option:
1. Sort via Name.
2. Sort via Age.
3. Exit

Select Your Choice : 3
Exiting program...

E:\Java\JOURNAL-2>
```

10. Design a class MyString having a data member of type String and add member functions to achieve following task. (i) Reverse string (ii) String in Titlecase. (iii) Extract N-characters from right-end of the string Write a menu driven program to call these methods of MyString class. The program should not terminate abruptly.

```
import java.util.Scanner;

public class PRG_10 {
    private String str;

    public PRG_10(String str) {
        this.str = str;
    }

    public String reverse() {
        StringBuilder sb = new StringBuilder(str);
        return sb.reverse().toString();
    }

    public String titleCase() {
        String[] words = str.split("\\s+");
        StringBuilder sb = new StringBuilder();
```



```
for (String word : words) {  
    if (word.length() > 0) {  
        sb.append(Character.toUpperCase(word.charAt(0)));  
        sb.append(word.substring(1).toLowerCase());  
        sb.append(" ");  
    }  
}  
return sb.toString().trim();  
}
```

```
public String extractFromRight(int n) {  
    if (n >= str.length()) {  
        return str;  
    }  
    return str.substring(str.length() - n);  
}
```

```
public static void main(String[] args) {  
    Scanner scanner = new Scanner(System.in);  
  
    System.out.print("Enter a string: ");  
    String inputString = scanner.nextLine();  
}
```

```
PRG_10 myString = new PRG_10(inputString);

while (true) {

    System.out.println("\nSelect an option:");

    System.out.println("1. Reverse the string");

    System.out.println("2. Convert the string to title case");

    System.out.println("3. Extract N characters from the right-end of the
string");

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

        System.out.print("\nSelect Your Choice : ");

    int choice = scanner.nextInt();

    scanner.nextLine();

    switch (choice) {

        case 1:

            System.out.println("Reversed string: " + myString.reverse());

            break;

        case 2:

            System.out.println("Title case string: " + myString.titleCase());

            break;

        case 3:

            System.out.print("Enter the number of characters to extract: ");
```

```
        int n = scanner.nextInt();

        scanner.nextLine();

        System.out.println("Extracted string: " +
myString.extractFromRight(n));

        break;
    case 4:

        System.out.println("Exiting program...");

        System.exit(0);

        break;
    default:

        System.out.println("Invalid choice. Try again.");

    }

}

}

}
```

Output :

```
E:\Java\JOURNAL-2>javac PRG_10.java

E:\Java\JOURNAL-2>java PRG_10
Enter a string: sybca the great class

Select an option:
1. Reverse the string
2. Convert the string to title case
3. Extract N characters from the right-end of the string
4. Exit

Select Your Choice : 1
Reversed string: ssalc taerg eht acbys

Select an option:
1. Reverse the string
2. Convert the string to title case
3. Extract N characters from the right-end of the string
4. Exit

Select Your Choice : 2
Title case string: Sybca The Great Class

Select an option:
1. Reverse the string
2. Convert the string to title case
3. Extract N characters from the right-end of the string
4. Exit

Select Your Choice : 3
Enter the number of characters to extract: 9
Extracted string: eat class

Select an option:
1. Reverse the string
2. Convert the string to title case
3. Extract N characters from the right-end of the string
4. Exit

Select Your Choice : 4
Exiting program...

E:\Java\JOURNAL-2>
```