

JAVA LAB

WEEK 1:

AIM: Accept two strings from the user and print it on console with concatenation of “and” in the middle of the strings.

a)Command line:

Program:

```
class Twostrings
{
    public static void main(String args[])
    {
        String a=args[0];
        String b=args[1];
        System.out.println("2 arguments are "+a+" and "+b);
    }
}
```

Output:

```
E:\books and pdfs\sem4 pdfs\java lab\week1>javac Twostrings.java
E:\books and pdfs\sem4 pdfs\java lab\week1>java Twostrings.java hello world
2 arguments are hello and world
E:\books and pdfs\sem4 pdfs\java lab\week1>
```

b)scanner class

```
import java.util.Scanner;
class Twostrings_scanner
{
```

```
public static void main(String args[])
{
    Scanner sc=new Scanner(System.in);
    String a=sc.nextLine();
    String b=sc.nextLine();
    sc.close();
    System.out.println("2 strings are "+a+" and "+b);
}
}
```

```
E:\books and pdfs\sem4 pdfs\java lab\week1>javac Twostings_scanner.java
E:\books and pdfs\sem4 pdfs\java lab\week1>java Twostings_scanner.java
hello world!
welcome to the world of java
2 strings are hello world! and welcome to the world of java
```

AIM:

Accept 12-hour time zone and convert into its corresponding 24-hour time zone. Note: Accept hours, minutes and seconds separately from the user (e.g. 07 05 45 PM should be displayed as 19:05:45)

a)Command line:

Program:

```
class Time_conversion{
    public static void main(String args[]){
        int hr=Integer.parseInt(args[0]);
        int min=Integer.parseInt(args[1]);
        int sec=Integer.parseInt(args[2]);
        String ampm=args[3];
```

```
        if((ampm.equals("pm") || ampm.equals("PM")) && hr<12)
        {
            hr=hr+12;
        }
        if((ampm.equals("pm") || ampm.equals("PM")) &&
hr==12)
        {
            hr=0;
        }

        System.out.printf("time in 24 hrs is
"+String.format("%02d:%02d:%02d",hr,min,sec));
    }
}
```

```
E:\books and pdfs\sem4 pdfs\java lab\week1>javac Time_conversion.java
E:\books and pdfs\sem4 pdfs\java lab\week1>java Time_conversion.java 5 6 7 pm
time in 24 hrs is 17:06:07
E:\books and pdfs\sem4 pdfs\java lab\week1>
```

b)Scanner class

```
import java.util.*;
class Time_conversion_sc{
    public static void main(String args[]){
        Scanner sc=new Scanner(System.in);
        int hr=sc.nextInt();
        int min=sc.nextInt();
        int sec=sc.nextInt();
        String ampm=sc.next();
        if((ampm.equals("pm") || ampm.equals("PM")) && hr<12)
        {
            hr=hr+12;
        }
    }
}
```

```
    }  
    if((ampm.equals("pm") || ampm.equals("PM")) &&  
hr==12)  
    {  
        hr=0;  
    }  
  
    System.out.printf("time in 24 hrs is %02d: %02d :  
%02d ",hr,min,sec);  
    System.out.println();  
    System.out.format("time in 24 hrs is %02d: %02d :  
%02d ",hr,min,sec);  
    System.out.println();  
    System.out.printf("time in 24 hrs is  
"+String.format("%02d:%02d:%02d",hr,min,sec));  
    }  
}
```

Output:

```
E:\books and pdfs\sem4 pdfs\java lab\week1>javac Time_conversion_sc.java  
E:\books and pdfs\sem4 pdfs\java lab\week1>java Time_conversion_sc.java  
4  
12  
55  
pm  
time in 24 hrs is 16:12:55  
E:\books and pdfs\sem4 pdfs\java lab\week1>
```

Aim:

Accept a number „n” and print the list of „n” Fibonacci terms recursively.

a)command line:

Program:

```
class Fibrec{
    protected static int fibb(int a){
        if(a==1||a==0)
            return 1;
        else
            return fibb(a-1)+fibb(a-2);
    }
    public static void main(String args[])
    {
        int a=Integer.parseInt(args[0]);
        for(int i=0;i<a;i++){
            System.out.println(fibb(i)+" is "+(i+1)+" th term
fibb series");
        }
    }
}
```

Output:

```
E:\books and pdfs\sem4 pdfs\java lab\week1>javac Fibrec.java

E:\books and pdfs\sem4 pdfs\java lab\week1>java Fibrec.java 10
1 is 1 th term fibb series
1 is 2 th term fibb series
2 is 3 th term fibb series
3 is 4 th term fibb series
5 is 5 th term fibb series
8 is 6 th term fibb series
13 is 7 th term fibb series
21 is 8 th term fibb series
34 is 9 th term fibb series
55 is 10 th term fibb series
```

b)scanner

```
import java.util.*;
class Fibbrec_sc{
    private static int fibbrec(int a){
        if(a==1||a==0)
            return 1;
        else
            return fibbrec(a-1)+fibbrec(a-2);
    }
    public static void main(String args[])
    {
        Scanner sc=new Scanner(System.in);
        int a=sc.nextInt();
        for(int i=0;i<a;i++){
            System.out.println(fibbrec(i)+" is "+(i+1)+" th
term fibb series");
        }
    }
}
```

Output:

```
E:\books and pdfs\sem4 pdfs\java lab\week1>javac Fibbrec_sc.java
E:\books and pdfs\sem4 pdfs\java lab\week1>java Fibbrec_sc.java
10
1 is 1 th term fibb series
1 is 2 th term fibb series
2 is 3 th term fibb series
3 is 4 th term fibb series
5 is 5 th term fibb series
8 is 6 th term fibb series
13 is 7 th term fibb series
21 is 8 th term fibb series
34 is 9 th term fibb series
55 is 10 th term fibb series
```

WEEK 2:

1)AIM:

Write a program that accepts the set of inputs from the user of various integer data types and determines the primitive data type that is capable of properly storing that input.

PROGRAM:

```
import java.util.*;
import java.io.*;
public class Integer_data_types
{
    public static void main(String... arg)
    {
        int n;
        Scanner sc=new Scanner(System.in);
        n=sc.nextInt();
```

```
for(int i=0;i<n;i++)
{
    try{
        long num=sc.nextLong();
        if(num>=-127 && num<128){
            System.out.println("BYTE");
        }
        if(num>Short.MIN_VALUE &&
num<Short.MAX_VALUE)
        {
            System.out.println("SHORT");
        }
        if(num >= Integer.MIN_VALUE && num <=
Integer.MAX_VALUE)
            System.out.println("*int");
        if(num >= Long.MIN_VALUE && num <=
Long.MAX_VALUE)
            System.out.println("*long");
        }
        catch(Exception e){
            System.out.println("No integer databtype
suits the input");
        }
    }
}
```



```
E:\books and pdfs\sem4 pdfs\java lab\week2>javac Integer_data_types.java
E:\books and pdfs\sem4 pdfs\java lab\week2>java Integer_data_types.java
6
123
BYTE
1354
SHORT
235436536
*int
235436536655658
*long
343
SHORT
23423
SHORT
```

2)AIM:Write a program that accepts an array of integers and print those which are both odd and prime. If no such element in that array print “Not found”.

PROGRAM:

```
import java.util.Scanner;
class Oddprime{
    public static int prime(int n){
        int i;
        for(i=2;i<=n/2;i++)
        {
            if(n%i==0)
                return 0;
        }
        return 1;
    }
    public static void main(String args[])
    {
        int i,n;
        Scanner sc=new Scanner(System.in);
```

```
n=sc.nextInt();
int arr[]=new int[n];
for(i=0;i<n;i++){
    arr[i]=sc.nextInt();
    if(arr[i]%2!=0){
        int p=prime(arr[i]);

    }
}
System.out.println("ODD Prime numbers are ");
for(i=0;i<n;i++){

    if(arr[i]%2!=0){
        int p=prime(arr[i]);
        if(p==1)
        {
            System.out.print(arr[i]+" ");
        }
    }
}

}
```

OUTPUT:

```
E:\books and pdfs\sem4 pdfs\java lab\week2>javac Oddprime.java
E:\books and pdfs\sem4 pdfs\java lab\week2>java Oddprime.java
5
41 45 55 11 81
ODD Prime numbers are
41 11
E:\books and pdfs\sem4 pdfs\java lab\week2>
```

3)AIM: Write a program to accept contents into an Integer Array and print the frequency of each number in the order of their number of occurrences.

PROGRAM:

```
import java.util.Scanner;
class Frequency {
    public static void main(String[] args) {
        //Initialize array
        int n;
        Scanner sc=new Scanner(System.in);
        n=sc.nextInt();
        int arr[] = new int[n];
        for(int i=0;i<n;i++)
            arr[i]=sc.nextInt();
        //Array fr will store frequencies of element
        int [] fr = new int [n];
        int visited = -1;
        for(int i = 0; i < arr.length; i++){
            int count = 1;
            for(int j = i+1; j < arr.length; j++){
                if(arr[i] == arr[j]){
                    count++;//4
                    //To avoid counting same element again
                    fr[j] = visited;
                }
            }
            if(fr[i] != visited)
                fr[i] = count;
        }
        int i,j;
```

```
for(i=0;i<arr.length;i++)
{
    for(j=i+1;j<arr.length;j++)
    {
        if(fr[i]>fr[j])
        {
            fr[i]=fr[i]+fr[j];
            fr[j]=fr[i]-fr[j];
            fr[i]=fr[i]-fr[j];

            arr[i]=arr[i]+arr[j];
            arr[j]=arr[i]-arr[j];
            arr[i]=arr[i]-arr[j];
        }
    }
}
//Displays the frequency of each element present in
array

System.out.println("-----
");
    System.out.println(" Element | Frequency");

System.out.println("-----
");
    for(i = 0; i < fr.length; i++){
        if(fr[i] != visited)
            System.out.println("      " + arr[i] + "      |
" + fr[i]);
    }

System.out.println("-----
-");
```

```
}  
}
```

```
E:\books and pdfs\sem4 pdfs\java lab\week2>java Frequency.java  
10  
23 42 22 23 33 42 22 22 10 10
```

```
-----  
Element | Frequency  
-----
```

```
33 | 1  
23 | 2  
10 | 2  
42 | 2  
22 | 3  
-----
```

WEEK 3:

1)AIM:

Write a program that accepts an „n” ordered square matrix elements into a single dimension array and print the elements of leading diagonal (top left to bottom right).

Program:

```
import java.util.Scanner;  
class Diagonal{  
    public static void main(String... arg){  
        int n,m,i,j;  
        Scanner sc=new Scanner(System.in);
```

```
n=sc.nextInt();
m=sc.nextInt();
int arr[][]=new int[n][m];
for(i=0;i<n;i++)
{
    for(j=0;j<m;j++)
    {
        arr[i][j]=sc.nextInt();
    }
}
for(i=0;i<n;i++){
    for(j=0;j<m;j++){
        if(i==j||(i+j)==n-1)
            System.out.print(arr[i][j]);
        else
            System.out.print("-");
    }
    System.out.println();
}
}
```

Output:

```
E:\books and pdfs\sem4 pdfs\java lab\week3>javac Diagonal.java
E:\books and pdfs\sem4 pdfs\java lab\week3>java Diagonal.java
3 3
1 2 3 4 5 6 7 8 9
1-3
-5-
7-9
```

2)Write a program that accepts an „m x n” double dimension array, where „m” represents financial years and „n” represents Ids of the items sold. Each element in the array represents the number of items sold in a particular year. Identify the year and id of the item which has more demand.

Program:

```
import java.util.Scanner;
class Week3_b{
    public static void main(String... arg){
        Scanner sc=new Scanner(System.in);
        int m=sc.nextInt();
        int n=sc.nextInt();
        int years[]=new int[m];
        int itemid[]=new int[n];
        System.out.println("enter the years");
        for(int i=0;i<m;i++){
            years[i]=sc.nextInt();
        }
        System.out.println("enter the years");
        for(int i=0;i<n;i++){
            itemid[i]=sc.nextInt();
        }
        int arr[][]=new int[2022][2022];
        System.out.println("enter the no of items sold");
        for(int i:years){
            for(int j:itemid){
                arr[i][j]=sc.nextInt();
            }
        }
    }
}
```

```
int mdemand=arr[m-1][n-1];
int dyear=years[m-1];
int did=itemid[n-1];
for(int i:years){
    for(int j:itemid){
        if(arr[i][j]>mdemand){
            mdemand=arr[i][j];
            dyear=i;
            did=j;
        }
    }
}
System.out.println("Item id "+did+" has max demand
of "+mdemand+" in year "+dyear);

}

}
```


Output:

```
E:\books and pdfs\sem4 pdfs\java lab\week3>java Week3_b.java
5 5
enter the years
2016 2017 2018 2019 2020
enter the years
1000 1001 1002 1003 1004
enter the no of items sold
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
Item id 1004 has max demand of 25 in year 2020

E:\books and pdfs\sem4 pdfs\java lab\week3>
```

3)AIM:

Write a program that accepts an „n” ordered square matrix and calculate the absolute difference between the sums of elements in their diagonals.

Program:

```
import java.util.scanner;
class Absdiff{
    public static void main(String... arg){
        int n,m,i,j;
        Scanner sc=new Scanner(System.in);
        n=sc.nextInt();
        m=sc.nextInt();
        int arr[][]=new int[n][m];
        for(i=0;i<n;i++)
        {
            for(j=0;j<m;j++)
            {
                arr[i][j]=sc.nextInt();
            }
        }
    }
}
```

```
int sum1=0,sum2=0;
for(i=0;i<n;i++){
    for(j=0;j<m;j++){
        if(i==j)
        {
            sum1+=arr[i][j];
        }
        if((i+j)==n-1)
        {
            sum2+=arr[i][j];
        }
    }
}
System.out.println(sum1+" "+sum2);
if((sum1-sum2)>0){
    System.out.println(sum1-sum2);
}
else
    System.out.println(sum2-sum1);
}
```

Output:

```
E:\books and pdfs\sem4 pdfs\java lab\week3>javac Absdiff.java
E:\books and pdfs\sem4 pdfs\java lab\week3>java Absdiff.java
3 3
1 2 3 4 5 6 7 8 9
diagonal sums are 15 15
difference is 0
```

WEEK 4:

- 1) AIM: Create a class Box that uses a parameterized constructor to initialize the dimensions of a box. The dimensions of the Box are width, height, depth. The class should have a method that can return the volume of the box. Create an object of the Box class and test the functionality.

Program:

```
import java.util.*;
class P4_1{
    static float height,depth,width;
    P4_1(float h,float w,float d){
        height=h;
        width=w;
        depth=d;
    }
    double volume(){
        return height*width*depth;
    }
    public static void main(String... arg){
        Scanner sc=new Scanner(System.in);
        float h=sc.nextFloat();
        float w=sc.nextFloat();
        float d=sc.nextFloat();
        P4_1 ob1=new P4_1(h,w,d);
        System.out.print("Volume is "+ob1.volume());
    }
}
```

OUTPUT:

```
E:\books and pdfs\sem4 pdfs\java lab\week4>javac P4_1.java
E:\books and pdfs\sem4 pdfs\java lab\week4>java P4_1.java
12.3
5.1
10.6
Volume is 664.9380493164062
E:\books and pdfs\sem4 pdfs\java lab\week4>
```

- 2) AIM: Create a new class called Calculator with the following methods:
- o A static method called powerInt(int num1,int num2) This method should return num1 to the power num2.
 - o A static method called powerDouble(double num1,double num2). This method should return num1 to the power num2.
 - o Invoke both the methods and test the functionality. Also count the number of objects created.

PROGRAM:

```
import java.util.*;
class P4_2{
    public static void main(String... arg){
        float a,b;
        int c,d;
        Scanner sc=new Scanner(System.in);
        a=sc.nextFloat();
        b=sc.nextFloat();
        c=sc.nextInt();
        d=sc.nextInt();
        Calculator mycal=new Calculator();
```

```
        System.out.println("Power double function  
"+mycal.powerDouble(a,b));  
        System.out.println("Power int function  
"+mycal.powerInt(c,d));  
        System.out.println("Number of times object is created  
"+mycal.count);
```

```
    }  
    }  
    class Calculator{  
        int count=1;  
        Calculator(){  
            ++count;  
        }  
        public static double powerDouble(double num1,double  
num2){  
            return Math.pow(num1,num2);  
        }  
        public static int powerInt(int num1,int num2){  
            double k=Math.pow(num1,num2);  
            int p=1;  
            for(int i=1;i<=num1;i++){  
                p*=num2;  
            }  
            return p;  
        }  
    }
```

OUTPUT:

```
E:\books and pdfs\sem4 pdfs\java lab\week4>JAVAC P4_2.java

E:\books and pdfs\sem4 pdfs\java lab\week4>java P4_2.java
1.2
2.3
4
5
Power double function 1.5209568803356843
Power int function 625
Number of times object is created 2
```

WEEK 5

- 1) AIM: Accept a String and a number n from the user. Divide the given string into substrings each of size n and sort them lexicographically.

Program:

```
import java.util.*;
class Strdivide{
    public static String[] Lexicographic(String ele,int n){
        String[] arr=new String[ele.length()/n];
        int s=0,i=0;
        while(s<ele.length()){

            String val=ele.substring(s,s+n);
            arr[i]=val;
            i+=1;
            s+=n;
        }
        return arr;
    }
}
```

```
}  
public static void main(String... arg){  
    Scanner sc=new Scanner(System.in);  
    System.out.println("string:");  
    String ele=sc.nextLine();  
    System.out.println("size");  
    int n=sc.nextInt();  
    String[] arr=new String[ele.length()%n];  
    ele=ele.replace(" ", "");  
    if(ele.length()%n!=0){  
        System.out.print("not possible");  
    }  
  
    else  
    {  
        arr=Lexicographic(ele,n);  
        Arrays.sort(arr);  
        System.out.println("substrings are");  
        for(String i:arr){  
            System.out.print(i+" ");  
        }  
    }  
}
```

OUTPUT:

```
E:\books and pdfs\sem4 pdfs\java lab\week5>javac Strdivide.java

E:\books and pdfs\sem4 pdfs\java lab\week5>java Strdivide.java
string:
gayatri vidya parishad college of engineering autonomous
size
2
substrings are
ad au co ee eg eo fe ga id in iv ll mo ng ng no pa ri ri sh to tr us ya ya
E:\books and pdfs\sem4 pdfs\java lab\week5>
```

- 2) AIM: Accept an array of strings and display the number of vowels and consonants occurred in each string.

PROGRAM:

```
import java.util.*;
class Vowels_consonents{
    public static void vowels_con(String s){
        int s1=0,s2=0;
        int j=0,k=0;
        for(int i=0;i<s.length();i++){
            if(s.charAt(i)=='a' || s.charAt(i)=='e' || s.charAt(i)=='i' || s.
            charAt(i)=='o' || s.charAt(i)=='u' || s.charAt(i)=='A' || s.charA
            t(i)=='E' || s.charAt(i)=='I' || s.charAt(i)=='O' || s.charAt(i)=
            ='U'){
                s1++;
                j++;
            }
            else{
                k++;
            }
        }
    }
}
```



```
        s2++;
        k++;
    }
}
System.out.println("string "+s+" number of vowels
"+s1);
System.out.println("string "+s+" number of
consonents "+s2);
System.out.println();
}
public static void main(String... arg){
    int n;
    Scanner sc=new Scanner(System.in);
    n=sc.nextInt();
    sc.nextLine();
    String[] arr=new String[n+1];
    System.out.println(n);

    for(int i=0;i<n;i++){
        {
            arr[i]=sc.nextLine().trim();
        }
        System.out.println(arr[0]);
        System.out.println(arr[1]);
        for(int i=0;i<n;i++){
            if(arr[i]!=""){
                vowels_con(arr[i]);
            }
        }
    }
}
```

OUTPUT:

```
E:\books and pdfs\sem4 pdfs\java lab\week5>javac Vowels_consonents.java

E:\books and pdfs\sem4 pdfs\java lab\week5>java Vowels_consonents.java
3
3
gayatri vidya parishad college of engineering
vizag
andhra pradesh
string gayatri vidya parishad college of engineering number of vowels 17
string gayatri vidya parishad college of engineering number of consonents 28

string vizag number of vowels 2
string vizag number of consonents 3

string andhra pradesh number of vowels 4
string andhra pradesh number of consonents 10
```

- 3) AIM: Accept two strings from the user and determine if the strings are anagrams or not.

PROGRAM:

```
import java.util.*;
public class Anagrams
{
    public static void main(String... arg){
        String s1,s2;

        Scanner sc=new Scanner(System.in);
        s1=sc.nextLine();
        s2=sc.nextLine();
        char []c1=s1.toCharArray();
```

```
char []c2=s2.toCharArray();
if(s1.length()!=s2.length()){
    System.out.println("Strings "+s1+" and "+s2+"
are not anagrams");
    System.exit(0);
}
for(int i=0;i<s1.length();i++){
    c1[i]=s1.charAt(i);
    c2[i]=s2.charAt(i);
}
Arrays.sort(c1);
Arrays.sort(c2);
int flag=1;
for(int i=0;i<s1.length();i++){
    if(c1[i]!=c2[i]){
        flag=0;
        break;
    }
}
if(flag==1){
    System.out.println("Strings "+s1+" and "+s2+"
are anagrams");
}
else
{
    System.out.println("Strings "+s1+" and "+s2+"
are not anagrams");
}
}
}
```

OUTPUT:

```
E:\books and pdfs\sem4 pdfs\java lab\week5>javaC Anagrams.java

E:\books and pdfs\sem4 pdfs\java lab\week5>java Anagrams.java
apple
pleap
Strings apple and pleap are anagrams

E:\books and pdfs\sem4 pdfs\java lab\week5>java Anagrams.java
hello
heloo
Strings hello and heloo are not anagrams
```