LIST OF PROGRAMS TO BE DONE AS PART OF EVERY WEEK:

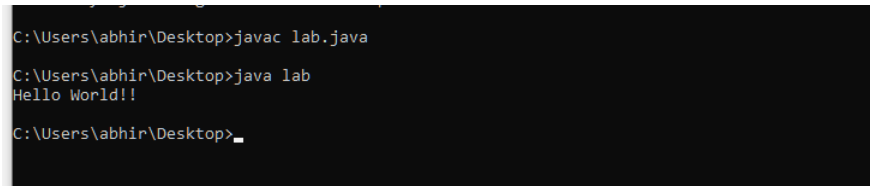
Week-I

1. Write a Java program print "Hello World".

PROGRAM:

```
import java.io.*;
public class helloworld  //class helloworld
{
        public static void main(String args[])
        {
                System.out.println("Hello World!! ");
        }
}
```

OUTPUT:



2. Write a Java program that prints all real and imaginary solutions to the quadratic equation ax2 + bx + c = 0. Read in a, b, c and use the quadratic formula

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
class qdteqn{
        public static void main(String args[]){
                Scanner in=new Scanner(System.in);
                int a,b,c,d;
                double r1,r2;
                System.out.print("Enter the values of a, b and c : ");
                a=in.nextInt();
                b=in.nextInt();
                c=in.nextInt();
                d=((b*b)-4*a*c);
                if(d>0){
```

```
                r1=((-b+Math.sqrt(d))/(2*a));
                r2=((-b-Math.sqrt(d))/(2*a));
                System.out.print("Roots are real and they are: "+r1+" and "+r2);
        }
        else if(d<0){
                r1=((-b)/(2*a));  //real part
                r2=((Math.sqrt(-d))/(2*a));  //imaginary part
                System.out.printf("Roots are imaginary and they are: %.2f+i%.2f and
%.2f-i%.2f ",r1,r2,r1,r2);
        }
        else{
                r1=((-b)/(2*a));
                System.out.print("Roots are real & distinct and they are: "+r1+" and
"+r1);
        }
    }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the values of a, b and c : 5 4 2
Roots are imaginary and they are: 0.00+i0.49 and 0.00-i0.49
C:\Users\abhir\Desktop>java lab
Enter the values of a, b and c : 2 3 4
Roots are imaginary and they are: 0.00+i1.20 and 0.00-i1.20
C:\Users\abhir\Desktop>java lab
Enter the values of a, b and c : 11 22 33
Roots are imaginary and they are: -1.00+i1.41 and -1.00-i1.41
```

3. Write a Java program to implement calculator operations.

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
class calc{
    public static void main(String args[]){
        float a, b, d;
        int c;
        Scanner in=new Scanner(System.in);
        System.out.println("Enter any 2 numbers: ");
        a=in.nextInt();
        b=in.nextInt();
        System.out.println("Enter your choice(1-ADD 2-SUBTRACT 3-MULTIPLY 4-
DIVIDE): ");
        c=in.nextInt();
        switch(c){
            case 1:{
```

```
                        System.out.print("Sum is "+(a+b));
                        break;
                }
                case 2:{
                        System.out.print("Difference is "+(a-b));
                        break;
                }
                case 3:{
                        System.out.print("Product is "+(a*b));
                        break;
                }
                case 4:{
                        System.out.print("Quotient is "+(a/b));
                        break;
                }
                default:
                        System.out.print("OOPS! invalid choice");
        }
    }
}
```

OUTPUT:

```
6
Roots are imaginary and they are: 0.00+i1.02 and 0.00-i1.02
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter any 2 numbers:
6 3
Enter your choice(1-ADD 2-SUBTRACT 3-MULTIPLY 4-DIVIDE):
1
Sum is 9.0
C:\Users\abhir\Desktop>
```

4. Write a java program to find prime factors of given number.

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
class facts
{
    public static void main(String args[])
    {
        int a,i,j,k;
        System.out.print("Enter the number: ");
        Scanner in=new Scanner(System.in);
        a=in.nextInt();
        System.out.print("Prime Factors are: ");
        for(i=2;i<=a;i++)
```

```
            {
               while(a%i == 0) {
              System.out.print(i+" ");
              a/=i;
            }
          }
          if(a >2){
             System.out.print(a);
              }
       }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the number: 5
Prime Factors are: 5
C:\Users\abhir\Desktop>java lab
Enter the number: 224
Prime Factors are: 2 2 2 2 2 7
C:\Users\abhir\Desktop>java lab
Enter the number: 543
Prime Factors are: 3 181
C:\Users\abhir\Desktop>
```

5. Write a java program to find whether given number is Palindrome or not.

PROGRAM:

```
import java.io.*;
import java.util.*;
class palindrome{
        public static void main(String args[]){
                int num, temp, rem=0, sum=0;
                System.out.print("Enter the number : ");
                Scanner vn=new Scanner(System.in);
                num=vn.nextInt();
                temp=num;
                while(num>0){
                        rem=num%10;
                        sum=rem+(sum*10);
                        num=num/10;
                }
                if(temp==sum)
                        System.out.println(temp+" is a palindrome number!");
                else
                        System.out.println(temp+" is not a palindrome number!");
        }
}
```

OUTPUT:

```
Prime Factors are: 3 181
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the number : 42432424
42432424 is not a palindrome number!

C:\Users\abhir\Desktop>java lab
Enter the number : 1234321
1234321 is a palindrome number!
```

6. Write an application that declares 5 integers, determines and prints the largest and smallest in the group.

PROGRAM:

```
import java.util.*;
public class lab {
   public static void main(String[]args){
            Scanner input = new Scanner(System.in);
            int number1, number2, number3, number4, number5, largest, smallest;
            System.out.print("Enter first integer: ");
            number1 = input.nextInt();
            System.out.print("Enter second integer: ");
            number2 = input.nextInt();
            System.out.print("Enter third integer: ");
            number3 = input.nextInt();
            System.out.print("Enter fourth integer: ");
            number4 = input.nextInt();
            System.out.print("Enter fifth integer: ");
            number5 = input.nextInt();
            largest = number1;
            smallest = number1;
            if(number2>largest){
               largest=number2;
            }
            if(number3>largest){
               largest=number3;
            }
            if(number4>largest){
               largest=number4;
            }
            if(number5>largest){
               largest=number5;
            }

            if(number2<smallest){
               smallest=number2;
            }
```

```
        if(number3<smallest){
           smallest=number3;
        }
        if(number4<smallest){
           smallest=number4;
        }
        if(number5<smallest){
           smallest=number5;
        }

        System.out.printf("Largest of five integers is %d, and smallest is %d%n", largest,
smallest);


   }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>java lab
Enter first integer: 55
Enter second integer: 2
Enter third integer: 44
Enter fourth integer: 24
Enter fifth integer: -1
Largest of five integers is 55, and smallest is -1
```

Week-II

1. Write a Java program to sort given list of numbers.

PROGRAM:

```
import java.io.*;
import java.util.*;
class sort{
        public static void main(String args[]){
                int n, i, j, t;
                Scanner vn1=new Scanner(System.in);
                Scanner vn=new Scanner(System.in);
                System.out.print("Enter the number of elements : ");
                n=vn1.nextInt();
                int a[]=new int[n];
                System.out.println("Enter the numbers : ");
                for(i=0; i<n; i++){
                        a[i]=vn.nextInt();
                }
                for(i=0;i<n;i++){
                        for(j=i+1;j<n;j++){
```

```
                        if(a[i]>a[j]){
                                t=a[i];
                                a[i]=a[j];
                                a[j]=t;
                              }
                        }
                }
            System.out.print("Sorted Order is :");
            for (i=0;i<n;i++)
            System.out.print(" "+a[i]);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the number of elements : 5
Enter the numbers :
4 6 1 -1 5
Sorted Order is : -1 1 4 5 6
C:\Users\abhir\Desktop>
```

2. Write a Java program to implement linear search.

PROGRAM:

```
import java.io.*;
import java.util.*;
class lsearch{
        public static void main(String arg[]){
                int n, find, i, search=0;
                Scanner vn1=new Scanner(System.in);
                System.out.print("Enter the number of elements : ");
                n=vn1.nextInt();
                int v[]=new int[n];
                System.out.println("Enter the numbers : ");
                for(i=0; i<n; i++){
                        v[i]=vn1.nextInt();
                }
                System.out.print("Enter the number to be searched : ");
                find=vn1.nextInt();
                for(i=0; i<n; i++){
                        if(find==v[i]){
                                search=1;
                                break;
                        }
                }
                if(search==1)
```

```
                System.out.println("Search Successful!! "+find+" found at position
"+(i+1));
            else
                System.out.println("Search Failed!! Match not found");
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the number of elements : 5
Enter the numbers :
4 3 -1 6 33
Enter the number to be searched : 6
Search Successful!! 6 found at position 4

C:\Users\abhir\Desktop>java lab
Enter the number of elements : 5
Enter the numbers :
4 3 -1 6 23
Enter the number to be searched : 0
Search Failed!! Match not found
```

3. Write a Java program to implement binary search.

PROGRAM:

```
import java.io.*;
import java.util.*;
class binsearch{
    public static void main(String args[]){
        int n, i, j, t, find, search=0, first, mid=0, last;
        Scanner vn1=new Scanner(System.in);
        System.out.print("Enter the number of elements : ");
        n=vn1.nextInt();
        int v[]=new int[n];
        System.out.println("Enter the numbers : ");
        for(i=0; i<n; i++){
            v[i]=vn1.nextInt();
        }
        System.out.print("Enter the number to be searched : ");
        find=vn1.nextInt();
        Arrays.sort(v); //sorting
        System.out.print("Sorted Order is : ");
        for (i=0;i<n;i++)
            System.out.print(" "+v[i]);
        first=0;
        last=n-1;
        for(i=0; i<n; i++){
```

```
                mid=(first+last)/2;
                if(find==v[mid]){
                        search=1;
                        break;
                }
                else if(find<v[mid])
                        last=mid-1;
                else
                        first=mid+1;
        }
        System.out.println("");
        if(search==1)
                System.out.println("Search Successful!! Match found");
        else
                System.out.println("Search Failed!! Match not found");
    }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the number of elements : 5
Enter the numbers :
11 12 -1 -2 4
Enter the number to be searched : 0
Sorted Order is :  -2 -1 4 11 12
Search Failed!! Match not found

C:\Users\abhir\Desktop>java lab
Enter the number of elements : 5
Enter the numbers :
11 12 -1 -2 4
Enter the number to be searched : -1
Sorted Order is :  -2 -1 4 11 12
Search Successful!! Match found
```

4. Write a java program to add two given matrices.

PROGRAM:

```
import java.io.*;
import java.util.*;
class addm{
    public static void main(String args[]){
        int m1, n1, m2, n2, i, j;
        Scanner vn1=new Scanner(System.in);
        System.out.print("Enter the number of rows and columns of matrix-1 : ");
        m1=vn1.nextInt();
        n1=vn1.nextInt();
        System.out.print("Enter the number of rows and columns of matrix-2 : ");
        m2=vn1.nextInt();
```

```
        n2=vn1.nextInt();
        if(m1==m2&&n1==n2){
            int add[][]=new int[m1][n1];
            int v1[][]=new int[m1][n1];
            System.out.println("Enter the elements of matrix-1 : ");
            for(i=0; i<m1; i++){
                for(j=0; j<n1; j++){
                    v1[i][j]=vn1.nextInt();
                }
            }
            int v2[][]=new int[m2][n2];
            System.out.println("Enter the elements of matrix-2 : ");
            for(i=0; i<m2; i++){
                for(j=0; j<n2; j++){
                    v2[i][j]=vn1.nextInt();
                }
            }
            System.out.println("The Matrix-1 is : ");
            for(i=0; i<m1; i++){
                for(j=0; j<n1; j++){
                    System.out.print(" "+v1[i][j]);
                }
                System.out.println(" ");
            }
            System.out.println("The Matrix-2 is : ");
            for(i=0; i<m2; i++){
                for(j=0; j<n2; j++){
                    System.out.print(" "+v2[i][j]);
                }
                System.out.println(" ");
            }
            //matrix addition
            for(i=0; i<m1; i++){
                for(j=0; j<n1; j++){
                    add[i][j]=v1[i][j]+v2[i][j];
                }
            }
            System.out.println("The Matrix Addition is : ");
            for(i=0; i<m1; i++){
                for(j=0; j<n1; j++){
                    System.out.print(" "+add[i][j]);
                }
                System.out.println(" ");
            }
        }
        else
            System.out.print("Rows and Columns not matched. Matrix addition not
possible!");
```

```
    }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>java lab
Enter the number of rows and columns of matrix-1 :  2 2
Enter the number of rows and columns of matrix-2 : 2 2
Enter the elements of matrix-1 :
 4 5
6 7
Enter the elements of matrix-2 :
5 6
7 8
The Matrix-1 is :
 4 5
 6 7
The Matrix-2 is :
 5 6
 7 8
The Matrix Addition is :
 9 11
 13 15
```

5. Write a java program to multiply two given matrices.

PROGRAM:

```
import java.io.*;
import java.util.*;
class multiplymatrix{
    public static void main(String args[]){
        int m1, n1, m2, n2, i, j, k;
        Scanner vn1=new Scanner(System.in);
        System.out.print("Enter the number of rows and columns of matrix-1 : ");
        m1=vn1.nextInt();
        n1=vn1.nextInt();
        System.out.print("Enter the number of rows and columns of matrix-2 : ");
        m2=vn1.nextInt();
        n2=vn1.nextInt();
        if(n1==m2){
            int mul[][]=new int[m1][n2];
            int v1[][]=new int[m1][n1];
            System.out.println("Enter the elements of matrix-1 : ");
            for(i=0; i<m1; i++){
                for(j=0; j<n1; j++){
                    v1[i][j]=vn1.nextInt();
                }
            }
            int v2[][]=new int[m2][n2];
            System.out.println("Enter the elements of matrix-2 : ");
            for(i=0; i<m2; i++){
                for(j=0; j<n2; j++){
```

```
                        v2[i][j]=vn1.nextInt();
                    }
                }
                System.out.println("The Matrix-1 is : ");
                for(i=0; i<m1; i++){
                    for(j=0; j<n1; j++){
                        System.out.print(" "+v1[i][j]);
                    }
                    System.out.println(" ");
                }
                System.out.println("The Matrix-2 is : ");
                for(i=0; i<m2; i++){
                    for(j=0; j<n2; j++){
                        System.out.print(" "+v2[i][j]);
                    }
                    System.out.println(" ");
                }
                //matrix multiplication
                for(i=0; i<m1; i++){
                    for(j=0; j<n2; j++){
                        for(k=0;k<n1;k++){
                            mul[i][j]+=v1[i][k]*v2[k][j];
                        }
                    }
                }
                System.out.println("The Matrix multiplication is : ");
                for(i=0; i<m1; i++){
                    for(j=0; j<n2; j++){
                        System.out.print(" "+mul[i][j]);
                    }
                    System.out.println(" ");
                }
            }
            else
                System.out.print("Rows and Columns not matched. Matrix Multiplication
not possible!");
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>java lab
Enter the number of rows and columns of matrix-1 : 2 4
Enter the number of rows and columns of matrix-2 : 2 4
Rows and Columns not matched. Matrix Multiplication not possible!
C:\Users\abhir\Desktop>java lab
Enter the number of rows and columns of matrix-1 : 2 2
Enter the number of rows and columns of matrix-2 : 2 2
Enter the elements of matrix-1 :
5 6
7 8
Enter the elements of matrix-2 :
11 22
33 44
The Matrix-1 is :
 5 6
 7 8
The Matrix-2 is :
 11 22
 33 44
The Matrix multiplication is :
 253 374
 341 506
```

6. Write a java program for sorting a given list of names.

PROGRAM:

```
import java.util.Arrays;
import java.util.Scanner;
class Sortnames{
        public static void main(String[] args) {
                int n;
                String temp;
                Scanner s = new Scanner(System.in);
                System.out.print("Enter the number of names you want to sort:");
                n = s.nextInt();
                String names[] = new String[n];
                Scanner s1 = new Scanner(System.in);
                System.out.println("Enter all the names:");
                for(int i = 0; i < n; i++){
                        names[i] = s1.nextLine();
                }
                for (int i = 0; i < n; i++) {
                        for (int j = i + 1; j < n; j++) {
                                if (names[i].compareTo(names[j])>0){
                                        temp = names[i];
                                        names[i] = names[j];
                                        names[j] = temp;
                                }
                        }
                }
                System.out.print("Names in Sorted Order:");
                for (int i = 0; i < n - 1; i++) {
                        System.out.print(names[i] + ",");
                }
                System.out.print(names[n - 1]);
```

```
            //System.out.print(Arrays.toString(names));
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the number of names you want to sort:5
Enter all the names:
abhiram
vishnu
karthik
vivek
shiva
Names in Sorted Order:abhiram,karthik,shiva,vishnu,vivek
C:\Users\abhir\Desktop>
```

7. Write a Java program to give an example for command line arguments.

PROGRAM:

```
import java.util.*;
class cmdlnag{
        public static void main(String args[]){
                int sum=0,i;
                for(i=0;i<args.length;i++)
                sum+=Integer.parseInt(args[i]);
                System.out.println("SUM:"+sum);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab 2 4 5 3
SUM:14
```

Week-III

1. Write a program to display details of the required employee based on his Id. The details of employee includes, Emp_name, Emp_age, Emp_gender, Emp_designation, Emp_salary, Emp_Address etc.,

PROGRAM:

```
//using class, methods
import java.io.*;
```

```
import java.util.*;
class employee{
	int id;
	String name;
	int age;
	String gender;
	String designation;
	int salary;
	String address;
	String number;
	void print(){
		System.out.println("\nEmployee ID found and the details are :\n");
		System.out.println("Employee ID \t\t: "+id);
		System.out.println("Employee Name\t\t: "+name);
		System.out.println("Employee Age\t\t: "+age);
		System.out.println("Employee Gender\t\t: "+gender);
		System.out.println("Employee Designation\t: "+designation);
		System.out.println("Employee Salary\t\t: "+salary);
		System.out.println("Employee Address\t: "+address);
		System.out.println("Employee Mobile number\t: "+number);
	}
	public static void main(String args[]){
		Scanner in=new Scanner(System.in);
		Scanner im=new Scanner(System.in);
		int find, i, n, flag=0, choice;
		System.out.print("Enter number of employees : ");
		n=in.nextInt();
		employee s[]=new employee[n];
		for(i=0;i<n;i++)
			s[i]=new employee();
		for(i=0;i<n;i++){
			System.out.println("\n");
			System.out.printf("Enter the ID of employee-%d : ",(i+1));
			s[i].id=in.nextInt();
			System.out.printf("Enter the Name of employee : ");
			s[i].name=im.nextLine();
			System.out.printf("Enter the Age \t\t\s\s\s: ");
			s[i].age=in.nextInt();
			System.out.printf("Enter the Gender \t\s\s\s: ");
			s[i].gender=im.nextLine();
			System.out.printf("Enter the Designation \t\s\s : ");
			s[i].designation=im.nextLine();
			System.out.printf("Enter the Salary\t\s\s : ");
			s[i].salary=in.nextInt();
			System.out.printf("Enter the Address  \t\s\s\s: ");
			s[i].address=im.nextLine();
			System.out.printf("Enter the Mobile Number\s\s\s : ");
			s[i].number=im.nextLine();
```

```
                System.out.println(" ");
            }
            do{
                System.out.print("Enter the ID of employee to be searched : ");
                find=in.nextInt();
                System.out.println("");
                for(i=0; i<n; i++){
                    if(find==s[i].id){
                        s[i].print();
                        flag=1;
                        break;
                    }
                }
                System.out.println("");
                if(flag==0)
                    System.out.println("Employee ID not found!!");
                System.out.println(" ");
                System.out.print("Enter 1 to continue the search operation or 0 to exit :");
                choice=in.nextInt();
                System.out.println("");
            }
            while(choice==1);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter number of employees : 3


Enter the ID of employee-1 : 123
Enter the Name of employee : abhiram
Enter the Age              : 19
Enter the Gender           : m
Enter the Designation      : prof
Enter the Salary           : 50000
Enter the Address          : hyd
Enter the Mobile Number    : 987654322



Enter the ID of employee-2 : 121
Enter the Name of employee : manish
Enter the Age              : 20
Enter the Gender           : m
Enter the Designation      : hr
Enter the Salary           : 89999
Enter the Address          : hyd
Enter the Mobile Number    : 987654567



Enter the ID of employee-3 : 423
Enter the Name of employee : manik
Enter the Age              : 22
Enter the Gender           : m
Enter the Designation      : manager
Enter the Salary           : 89284
Enter the Address          : pune
Enter the Mobile Number    : 98654323
```

```
Enter the ID of employee to be searched : 123


Employee ID found and the details are :

Employee ID             : 123
Employee Name           : abhiram
Employee Age            : 19
Employee Gender         : m
Employee Designation    : prof
Employee Salary         : 50000
Employee Address        : hyd
Employee Mobile number  : 987654322


Enter 1 to continue the search operation or 0 to exit :
```

2. A mail-order house sells five products whose retail prices are as follows : Product 1 : Rs. 99.90 , Product 2 : Rs. 20.20 , Product 3 : Rs. 6.87 , Product 4 : Rs. 45.50 and Product 5 : Rs. 40.49 . Each product has Prdouct_Id, Product_Name, Product_Quantity, Product_Price.
Write an application that reads a series of pairs of numbers as follows :
a) product Id
b) quantity sold your program use a switch statement to determine the retail price for each product. it should calculate and display the total retail value of all products sold.

PROGRAM:

```
import java.util.Scanner;
class product{
        public static void main(String []args){
                Scanner n=new Scanner(System.in);
                int product_1;
                int product_2;
                int product_3;
                int product_4;
                int product_5;
                double total_price=0;
                System.out.println("Choose product 1 to 5");
                System.out.println("Available products are ");
                System.out.println("Product 1: Rs.99\nProduct 2: Rs.89\n Product 3: Rs.49\n Product 4: Rs.199\n Product 5: Rs.9");
                int product_no=n.nextInt();
                System.out.println("enter the quantity sold");
                int quantity=n.nextInt();
                switch(product_no){
                case 1:
                        product_1=99;
                        total_price=(99*quantity);
                        System.out.println("the retail price is"+total_price);
                        break;
                case 2:
                        product_2=89;
                        total_price=(89*quantity);
                        System.out.println("the retail price is"+total_price);
                        break;
                case 3:
                        product_3=49;
```

```
                total_price=(49*quantity);
                System.out.println("the retail price is"+total_price);
                break;
            case 4:
                product_4=199;
                total_price=(199*quantity);
                System.out.println("the retail price is"+total_price);
                break;
            case 5:
                product_5=9;
                total_price=(9*quantity);
                System.out.println("the retail price is"+total_price);
                break;
            }
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Choose product 1 to 5
Available products are
Product 1: Rs.99
Product 2: Rs.89
 Product 3: Rs.49
 Product 4: Rs.199
 Product 5: Rs.9
1
enter the quantity sold
23
the retail price is2277.0
```

3. Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it's not a duplicate of any number already read display the complete set of unique values input after the user enters each new value.

PROGRAM:

```
import java.util.*;
public class lab {
   public static void main(String[] args) {
      int [] a={0,0,0,0,0};
      Scanner sc=new Scanner(System.in);
      for(int i=0;i<a.length;i++){
         int c=0;
         System.out.println("enter the element   ");
         int t=sc.nextInt();
         for(int j=0;j<a.length;j++){
            if(t==a[j])
            {
```

```
                c=1;
                System.out.println("number already exists,enter another number");
                i=i-1;
                break;
            }
            else if(t<10||t>100)
            {
                c=1;
                System.out.println("number is not in range (10&100)enter a valid number");
                i=i-1;
                break;
            }
        }
        if(c!=1)
        {
            a[i]=t;
            for (int k=0;k<=i;k++)
            {
                System.out.print(a[k]+" ");
            }
        }
    }
  }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
enter the element
34
34 enter the element
1
number is not in range (10&100)enter a valid number
enter the element
32
34 32 enter the element
56
34 32 56 enter the element
32
number already exists,enter another number
enter the element
44
34 32 56 44 enter the element
21
34 32 56 44 21
```

4. Write a java program : rolling a pair of dices 10 times [each attempt should be delayed by 10000 ms] and count number Successful attempts. successful attempt : If the pair of Dice results in same values.

PROGRAM:

```
import java.lang.Math;
import java.util.Random;
```

```
class Dices{
	public static void main(String[] args){
		int num1,num2,i,sa=0;
		for(i=0;i<10;i++){
			Random r=new Random();
			num1=r.nextInt(6)+1;//to produce from n1 to n2 produce from 0 to n2-n1 and add n1
			num2=r.nextInt(6)+1;
			System.out.println("num1 "+num1+"   num2 "+num2);
			if(num1==num2)
				sa++;
			try{
				Thread.sleep(10000);
				}
			catch(InterruptedException ie){
				System.out.println("interrupted");
				}
		}
		System.out.println("successful attempts = "+sa);
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
num1 1   num2 3
num1 2   num2 3
num1 3   num2 5
num1 5   num2 2
num1 6   num2 6
num1 1   num2 2
num1 4   num2 6
num1 4   num2 4
num1 5   num2 1
num1 5   num2 2
successful attempts = 2
```

5. Implement the following case study using OOP concepts in Java.
E-Book stall : Every book has Properties which includes : Book _Name, Book_Author, Book_Count ;
Every Customer is having properties as : Customer_Id, Customer_Name, Customer_Address and he can buy Books from E-Book stall.
Write a Program which will display the text book name and the remaining count of text books when a customer buys a text book.

PROGRAM:

```
//package com.company;
import java.util.*;
public class book_stall {
  static String[] book_name={"The Braille edition of the book Exam Warriors","Believe-What Life and
Cricket Taught Me","The Christmas Pig","Whereabouts"};
```

```
static String[] book_authors={"PM Narendra Modi","Suresh Raina","JK Rowling","Jhumpa Lahiri"};
static  int[] book_quant={10,10,10,10};
Scanner sc=new Scanner(System.in);
 void books_available(){

   System.out.println("-----------------------------------------------------------------------------------------");
   System.out.println("                              E Book Stall                                 ");
   System.out.println("\t\t\t\t available books are :  \t\t\t\t");
   for (int i=0;i<4;i++){
      System.out.println(book_name[i]+"  "+book_authors[i]+"  "+book_quant[i]);
   }
   System.out.println("-----------------------------------------------------------------------------------------");
}
void customer_details(int i,String s,String k){
   System.out.println("Customer id "+i);
   System.out.println("Customer name : "+s);
   System.out.println("Customer address : "+k);
}
void search(){
   Scanner scan=new Scanner(System.in);
   System.out.println("Enter the book name you want to buy :");
   String w_book=scan.nextLine();
   int r=0;
   for (int i=0;i<4;i++){
      if(Objects.equals(w_book, book_name[i])){
         System.out.println("Enter the num of books you want to buy: ");
         int qun=sc.nextInt();
         if(qun>book_quant[0]){
            System.out.println("Stock not available : ");
         }
         else {
            r = book_quant[i] - qun;
            System.out.println(r);
            System.out.println(r + "still num of books available:  ");
         }
      }
   }

}
public static void main(String []args){
   Scanner sc=new Scanner(System.in);
   book_stall a=new book_stall();
  a.books_available();
   System.out.println("enter customer id :");
   int id=sc.nextInt();
   System.out.println("enter customer name :");
   String name=sc.next();
   System.out.println("enter customer address :");
   String add=sc.next();
   a.customer_details(id,name,add);
   a.search();
```

```
  }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
-----------------------------------------------------------------------------------------
                                E Book Stall
                              available books are :
The Braille edition of the book Exam Warriors  PM Narendra Modi  10
Believe-What Life and Cricket Taught Me  Suresh Raina  10
The Christmas Pig  JK Rowling  10
Whereabouts  Jhumpa Lahiri  10
-----------------------------------------------------------------------------------------
enter customer id :
323
enter customer name :
abhiram
enter customer address :
hyd
Customer id 323
Customer name : abhiram
Customer address : hyd
Enter the book name you want to buy :
Whereabouts
Enter the num of books you want to buy:
4
6
6still num of books available:
```

Week-IV

1. Write an application that uses String method compareTo to compare two strings defined by the user.

PROGRAM:

```
import java.util.Scanner;
class CompareTo{
        public static void main(String args[]){
                String s1,s2;
                Scanner in=new Scanner(System.in);
                System.out.print("Enter two strings : ");
                s1=in.nextLine();
                s2=in.nextLine();
                if (s1.compareTo(s2)==0) System.out.println("Two strings are equal");
                else if(s1.compareTo(s2)<0) System.out.println(s2+" comes after "+s1);
                else if(s1.compareTo(s2)>0) System.out.println(s2+" comes before "+s1);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>java lab
Enter two strings : abhiram
abhi
abhi comes before abhiram

C:\Users\abhir\Desktop>java lab
Enter two strings : abhiram vishnu
vishnu
vishnu comes after abhiram vishnu

C:\Users\abhir\Desktop>vishnu
'vishnu' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\abhir\Desktop>java lab
Enter two strings : vishnu
abhiram
abhiram comes before vishnu
```

2. Write an application that uses String method equals and equalsIgnoreCase to tests any two string objects for equality.

PROGRAM:

```
import java.util.Scanner;
import java.lang.String;
class EqualsEqualsIgnore{
        public static void main(String args[]){
                Scanner in=new Scanner(System.in);
                String s1=new String(),s2=new String();
                System.out.println("Enter two strings : ");
                s1=in.next();
                s2=in.next();
                System.out.println("using Equals "+s1.equals(s2));
                System.out.println("using EqualsIgnoreCase "+s1.equalsIgnoreCase(s2));
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter two strings : abhiram
ABHIRAM
ABHIRAM comes before abhiram

C:\Users\abhir\Desktop>vishnu
'vishnu' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter two strings :
abhiram
ABHiram
using Equals false
using EqualsIgnoreCase true
```

3. Write an application that uses String method indexOf to determine the total number of occurrences of any given alphabet in a defined text.

PROGRAM:

```
import java.util.Scanner;
class Occurances{
	public static void main(String[] args){
		Scanner in=new Scanner(System.in);
		String s;
		char c;
		int count,i;
		System.out.println("Enter a string");
		s=in.next();
		System.out.println("Enter any character");
		c=in.next().charAt(0);
		for(count=0,i=0;i<s.length();i++) {	//see Occurances1.java for better version
			if(s.indexOf(c,i)!=-1){
				count++;
				i=s.indexOf(c,i);
			}
		}
		System.out.println("Number of "+c+"'s in "+s+" are "+count);
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter a string
abhiram
Enter any character
a
Number of a's in abhiram are 2

C:\Users\abhir\Desktop>java lab
Enter a string
abhiram
Enter any character
y
Number of y's in abhiram are 0

C:\Users\abhir\Desktop>
```

4. Write an application that uses String method concat to concatenate two defined strings.

PROGRAM:

```
import java.util.Scanner;
class StringConcat{
	public static void main(String[] args){
		Scanner in=new Scanner(System.in);
		String s1,s2,s3;
		System.out.println("Enter two strings");
		s1=in.next();
		s2=in.next();
		s3=s1.concat(s2);
		System.out.println("concated string is "+s3);
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter two strings
abhi
ram
concated string is abhiram

C:\Users\abhir\Desktop>
```

5. Write a Java program to print all vowels in given string and count number of vowels and consonants present in given string

PROGRAM:

```
import java.util.Scanner;
class VowelsConsonants{
	public static void main(String[] args){
	Scanner in=new Scanner(System.in);
	String s;
	char c;
	int i,count;
	System.out.println("Enter a string");
	s=in.nextLine();
	System.out.println("vowels are ");
	for(i=0,count=0;i<s.length();++i){
		c=s.charAt(i);
		if(c=='a'||c=='e'||c=='i'||c=='o'||c=='u'||c=='A'||c=='E'||c=='I'||c=='O'||c=='U'){
				System.out.print(c+" ");
				count++;
			}
```

```
        }
        System.out.println();
        System.out.println("total vowels are "+count);
        System.out.println("total consonants are "+(s.length()-count));
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter a string
abhiram
vowels are
a i a
total vowels are 3
total consonants are 4

C:\Users\abhir\Desktop>
```

6. Write an application that finds the length of a given string.

PROGRAM:

```
import java.util.Scanner;
class StringLength{
        public static void main(String[] args){
                Scanner in=new Scanner(System.in);
                String s;int length=0;
                System.out.println("Enter a string");
                s=in.next();
                char sa[]=s.toCharArray();
                for(char c:sa) length++;
                System.out.println("length of "+s+" is "+length);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter a string
abhiram
length of abhiram is 7

C:\Users\abhir\Desktop>
```

7. Write an application that uses String method charAt to reverse the string.

PROGRAM:

```
import java.util.Scanner;
class StringReverse{
	public static void main(String[] args){
	Scanner in=new Scanner(System.in);
	String s,rs="";
	int i;
	System.out.println("Enter a string");
	s=in.nextLine();
	for(i=s.length()-1;i>=0;i--){
		rs=rs+s.charAt(i);
	}
	System.out.println("reversed string is : "+rs);
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter a string
abhiram
reversed string is : marihba

C:\Users\abhir\Desktop>
```

8. Write an application that finds the substring from any given string using substring method and startsWith & endsWith methods.

PROGRAM:

```
import java.util.Scanner;
class SubString{
	public static void main(String[] args){
		Scanner in=new Scanner(System.in);
		System.out.println("Enter a string");
		String s=in.nextLine();
		System.out.println("Enter start,end indices for substring");
		int i=in.nextInt(),j=in.nextInt();
		String ss=s.substring(i-1,j-1);
		System.out.println("substring is "+ss);
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter a string
Abhiram
Enter start,end indices for substring
1 5
substring is Abhi

C:\Users\abhir\Desktop>
```

9. Write an application that changes any given string with uppercase letters, displays it, changes it back to lowercase letters and displays it.

PROGRAM:

```
import java.util.Scanner;
class UpperToLower{
        public static void main(String[] args){
        Scanner in=new Scanner(System.in);
        String s;
        System.out.println("Enter a string");
        s=in.nextLine();
        System.out.println("in uppercase "+s.toUpperCase());
        System.out.println("in lowercase "+s.toLowerCase());
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter a string
ABHiram
in uppercase ABHIRAM
in lowercase abhiram

C:\Users\abhir\Desktop>
```

Week-V

1. Write a Java Program to implement Wrapper classes and their methods.

PROGRAM:

```
import java.util.Scanner;
class wrapper1
{
	public static void main(String args[])
	{
		Scanner vn=new Scanner(System.in);
		System.out.print("Enter a Number :");
		int a=vn.nextInt();
		Integer c= new Integer(a);
		System.out.print("Enter another Number :");
		int b=vn.nextInt();
		Integer d= new Integer(b);
		System.out.println("Addition of given numbers is : "+(c+d));
		System.out.println("Subtraction of given numbers is : "+(c-d));
		System.out.println("Multiplication of given numbers is : "+(c*d));
		System.out.println("Division of given numbers is : "+(c/d));
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>java lab.java
lab.java:9: warning: [removal] Integer(int) in Integer has been deprecated and marked for removal
                Integer c= new Integer(a);
                           ^
lab.java:12: warning: [removal] Integer(int) in Integer has been deprecated and marked for removal
                Integer d= new Integer(b);
                           ^
2 warnings
Enter a Number :7
Enter another Number :3
Addition of given numbers is : 10
Subtraction of given numbers is : 4
Multiplication of given numbers is : 21
Division of given numbers is : 2
```

2. Write an application that prompts the user for the radius of a circle and uses a method called circleArea to calculate the area of the circle and uses a method circlePerimeter to calculate the perimeter of the circle.

PROGRAM:

```
import java.util.Scanner;
class calc
{
	double r;
	double circleArea()
```

```
	{
		return (22/7)*r*r;
	}
	double circlePerimeter()
	{
		return 2*(22/7)*r;
	}
}
class circle
{
	public static void main(String args[])
	{
		Scanner vn=new Scanner(System.in);
		calc c=new calc();
		System.out.print("Enter the radius of the circle : ");
		double R=vn.nextDouble();
		c.r=R;
		System.out.println("Perimeter of circle is:"+c.circlePerimeter());
		System.out.println("Area of circle is:"+c.circleArea());
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter the radius of the circle : 3
Perimeter of circle is:18.0
Area of circle is:27.0

C:\Users\abhir\Desktop>
```

3. Write a JAVA program for the following
a. Call by value

PROGRAM:

```
class value
{
	int age=22;
	void change_age(int age)
	{
		age=age+10;
		System.out.println("age in local : "+age);
	}
	public static void main(String args[])
	{
		value c1=new value();
		System.out.println("age before change :"+c1.age);
		c1.change_age(10);
		System.out.println("age after change : "+c1.age);
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
age before change :22
age in local : 20
age after change : 22

C:\Users\abhir\Desktop>
```

b. Call by object

PROGRAM:

```
class object
{
        int age=22;
        void change_age(object c1)
        {
                age=age+10;// cahnges will be done in the local variable only
                System.out.println("age in local : "+age);
        }
        public static void main(String args[])
        {
                object c1=new object();
                System.out.println("age before change : "+c1.age);
                c1.change_age(c1);// passing object through called method
                System.out.println("age after change : "+c1.age);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
age before change : 22
age in local : 32
age after change : 32

C:\Users\abhir\Desktop>
```

4. Create a class Account with an instance variable balance (double). It should contain a constructor that initializes the balance, ensure that the initial balance is greater than 0.0. Acct details: Acct_Name, Acct_acctno, Acct_Bal, Acct_Address. Create two methods namely credit and debit, getBalance. The Credit adds the amount (passed as parameter) to balance and does not return any data. Debit method withdraws money from an Account. GetBalance displays the amount. Ensure that the debit amount does not exceed the Account's balance. In that case the balance should be left unchanged

and the method should print a message indicating "Debit amount exceeded account balance".

PROGRAM:

```
import java.util.Scanner;
class Account
{
	String name;
	int acc_no;
	double balance=0;
	String address;
	double debit;
	Account(double bal)
	{
		balance=bal;
		if(balance<=0)
			System.out.println("You account has zero balance!");
	}
	void credit(double amount)
	{
		balance=balance+amount;
	}
	void debit(double withdraw)
	{
		debit=withdraw;
		if(debit>balance)
		{
			System.out.println("Debit amount exceeded account balance");
		}
		else if(debit<=balance)
		{
			balance=balance-debit;
		}
	}
	void getBalance()
	{
		System.out.print("available balance : "+balance);
	}
	public static void main(String args[])
	{
		Scanner vn=new Scanner(System.in);
		Account a=new Account(2000); //minimum balance
		int cond;
		int choice;
		double amount;
		System.out.print("Enter accountholder name and account number : ");
		a.name=vn.next();
		a.acc_no=vn.nextInt();
		do
		{
			System.out.print("\nEnter your choice(1=credit 2=debit 3=check balance): ");
			choice=vn.nextInt();
			switch(choice)
			{
```

```
                        case 1:
                        {
                                System.out.print("Enter amount to credit : ");
                                amount=vn.nextInt();
                                a.credit(amount);
                                break;
                        }
                        case 2:
                        {
                                System.out.print("Enter amount to debit : ");
                                amount=vn.nextInt();
                                a.debit(amount);
                                break;
                        }
                        case 3:
                        {
                                System.out.print("Available Balance is : ");
                                a.getBalance();
                                break;
                        }
                        default:
                        {
                                System.out.println("OOPS! 404 not found. ENJOY the course!!");
                                break;
                        }
                }
                System.out.print("\nPress 1 to continue operation or 0 to end : ");
                cond=vn.nextInt();
        }
        while(cond==1);
    }
}
```

OUTPUT:

```
\Users\abhir\Desktop>javac lab.java

\Users\abhir\Desktop>java lab
ter accountholder name and account number : Abhiram 89765344

ter your choice(1=credit 2=debit 3=check balance): 1
ter amount to credit : 23823

ess 1 to continue operation or 0 to end : 1

ter your choice(1=credit 2=debit 3=check balance): 3
ailable Balance is : available balance : 25823.0
ess 1 to continue operation or 0 to end :
```

5. Write Java program for the following
a. Example for this operator and the use of this keyword.

PROGRAM:

```
import java.util.*;
import java.io.*;
class THIS{
	int id;
	String branch;
	THIS(int id,String branch){
		this.id=id;
		this.branch=branch;
	}
	void display(){
		System.out.println("id is : "+id);
		System.out.println("branch is : "+branch);
	}
	public static void main(String []args){
		System.out.println("Enter id number and branch : ");
		Scanner sc=new Scanner(System.in);
		int i=sc.nextInt();
		String br=sc.next();
		THIS key=new THIS(i,br);
		key.display();
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac lab.java

C:\Users\abhir\Desktop>java lab
Enter id number and branch :
182817 ECE
id is : 182817
branch is : ECE

C:\Users\abhir\Desktop>
```

b. Example for super keyword.

PROGRAM:

```
class ParentClass
{
	int i;
	ParentClass(int i)
	{
		this.i=i;
	}
}
class ChildClass extends ParentClass
{
	int i=10;
	ChildClass(int i)
	{
		super(100);
```

```
                this.i=i;
        }
        void display()
        {
                System.out.println("i in Child class object : "+i);
                System.out.println("i in Parent class object: "+super.i);
        }
}
 class ExampleForSuper
{
        public static void main(String[] args)
        {
                ChildClass c=new ChildClass(1000);
                c.display();
        }
}
```

OUTPUT:

```
:\Users\abhir\Desktop>javac ExampleForSuper.java

:\Users\abhir\Desktop>java ExampleForSuper
 in Child class object : 1000
 in Parent class object: 100

:\Users\abhir\Desktop>
```

c. Example for static variables and methods.

PROGRAM:

```
class StaticBlock
{       static{  //Static Block
        System.out.println("You are in Static Block");
        }
        static void display(){  //static method
                System.out.println("I am Static method");
        }
        static int a=5;  //Static Variable
        public static void main(String args[]){
                StaticBlock.display();
                System.out.println(StaticBlock.a);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac StaticBlock.java

C:\Users\abhir\Desktop>java StaticBlock
You are in Static Block
I am Static method
5

C:\Users\abhir\Desktop>
```

Week-VI

1. Write a Java program to find Area and Circle of different shapes using polymorphism concept.

PROGRAM:

```
class shape
{
	void area(int x)
	{
		System.out.println("The area of the square: "+x*x+" sq.units");
	}
	void area(double x,double y)
	{
		System.out.println("The area of the rectangular: "+x*y+" sq.units");
	}
	void area(double x)
	{
		double z=3.14*x*x;
		System.out.println("The area of the circle "+z+" sq.units");
	}
	void area(int x,int y)
	{
		float r=(x*y)/2;
		System.out.println("The area of the trianle "+r+" sq.units");
	}
}
class valueshape
{
	public static void main(String args[])
	{
		shape s=new shape();
		s.area(10);
		s.area(20.0,30.0);
		s.area(5.0);
		s.area(2,5);
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac valueshape.java

C:\Users\abhir\Desktop>java valueshape
The area of the square: 100 sq.units
The area of the rectangular: 600.0 sq.units
The area of the circle 78.5 sq.units
The area of the trianle 5.0 sq.units

C:\Users\abhir\Desktop>
```

2. Write a Java program which can give example of
i) Method overloading.

PROGRAM:

```
class parent{
	static int add(int a, int b){
		return a+b;
	}
}
class child extends parent{
	static int add(int a, int b, int c){
		return a+b+c;
	}
	static double add(double a, double b, double c){
		return a+b+c;
	}
}
class overload{
	public static void main(String args[]){
		System.out.println("sum is");
		System.out.println(parent.add(2,4));
		System.out.println(child.add(2,4,6));
		System.out.println(child.add(2.6,4.4,3.2));
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac valueshape.java

C:\Users\abhir\Desktop>java valueshape
The area of the square: 100 sq.units
The area of the rectangular: 600.0 sq.units
The area of the circle 78.5 sq.units
The area of the trianle 5.0 sq.units

C:\Users\abhir\Desktop>javac overload.java

C:\Users\abhir\Desktop>java overload
sum is
6
12
10.2

C:\Users\abhir\Desktop>
```

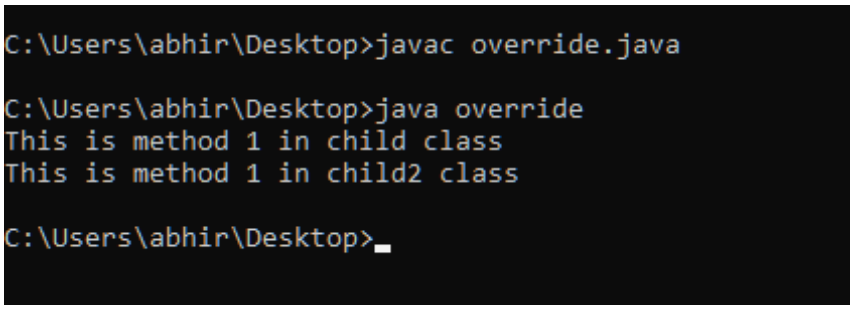
ii) Overriding.

PROGRAM:

```
class parent{
	static void method1(){
		System.out.println("This is method 1 in parent class");
	}
```

```
}
class child extends parent{
	static void method1(){
		System.out.println("This is method 1 in child class");
	}
}
class child2 extends parent{
	static void method1(){
		System.out.println("This is method 1 in child2 class");
	}
}
class override{
	public static void main(String args[]){
		child va=new child();
		va.method1();
		child2 va1=new child2();
		va1.method1();
	}
}
```

OUTPUT:



3. Write an application to create a super class Employee with information first name & last name and methods getFirstName(), getLastName() derive the sub-classes ContractEmployee and RegularEmployee with the information about department, designation & method displayFullName() , getDepartment(), getDesig() to print the salary and to set department name & designation of the corresponding sub-class objects respectively.

PROGRAM:

```
import java.util.Scanner;
class employe{
	String first_name;
	String last_name;
	void getfirstname(String f){
		first_name=f;
		System.out.println("first name is "+f);
	}
	void getlastname(String l){
		last_name=l;
		System.out.println("last name is "+l);
	}
}
```

```
class c_emp extends employe
{
	String full_name;
	String department;
	String designation;
	void displayfullname(String cfn){
		full_name=cfn;
		System.out.println(" CONTRACT Full NAME:"+cfn);
	}
	void getdepartment(String cd){
		department=cd;
		System.out.println("DEPARTMENT:"+cd);
	}
	void getdesignation(String cds){
		designation=cds;
		System.out.println("DESIGNATION:"+cds);
	}
}
class r_emp extends employe{
	String full_name;
	String department;
	String designation;
	void displayfullname(String rfn){
		full_name=rfn;
		System.out.println(" REGUAR Full NAME:"+rfn);
	}
	void getdepartment(String rd){
		department=rd;
		System.out.println("DEPARTMENT:"+rd);
	}
	void getdesignation(String rds){
		designation=rds;
		System.out.println("DESIGNATION:"+rds);
	}
}
class employee{
	public static void main(String []args){
	Scanner p=new Scanner(System.in);
	System.out.println("enter employee first name");
	String fn=p.next();
	System.out.println("enter empoyee second nane");
	String ln=p.next();
	System.out.println("enter contract employee full name");
	String cf=p.next();
	System.out.println("enter the contract employee department");
	String c_dep=p.next();
	System.out.println("enter contract employee designation");
	String c_des=p.next();
	System.out.println("enter regular employee full name");
	String rf=p.next();
	System.out.println("enter the regualar employee department");
	String r_dep=p.next();
	System.out.println("enter regular employee designation");
	String r_des=p.next();
	employe details=new employe();
	details.getfirstname(fn);
```

```
        details.getlastname(ln);
        c_emp detail=new c_emp();
        detail.displayfullname(cf);
        detail.getdepartment(c_dep);
        detail.getdesignation(c_des);
        r_emp detl=new r_emp();
        detl.displayfullname(rf);
        detl.getdepartment(r_dep);
        detl.getdesignation(r_des);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac employee.java

C:\Users\abhir\Desktop>java employee
enter employee first name
abhiram
enter empoyee second nane
rote
enter contract employee full name
shiva
enter the contract employee department
technical
enter contract employee designation
hr
enter regular employee full name
bunny
enter the regualar employee department
md
enter regular employee designation
 mr
first name is abhiram
last name is rote
 CONTRACT Full NAME:shiva
DEPARTMENT:technical
DESIGNATION:hr
 REGUAR Full NAME:bunny
DEPARTMENT:md
DESIGNATION:mr

C:\Users\abhir\Desktop>
```

4. Derive sub-classes of ContractEmployee namely HourlyEmployee & WeeklyEmployee with information number of hours & wages per hour, number of weeks & wages per week respectively & method calculateWages() to calculate their monthly salary. Also override getDesig () method depending on the type of contract employee.

PROGRAM:

```
import java.util.Scanner;
class contract_emp{
        String designation;
        void getdesignation(String d)
        {
                designation=d;
                System.out.println("designation is :"+d);
```

```
        }
}
class h_employee extends contract_emp{
        float hours;
        float hpay;
        float ht_salary;
        void calculate_wages(float h,float hp){
                hours=h;
                hpay=hp;
                ht_salary=hours*hpay*30;
                System.out.println("total monthly salary of hourly employee is : "+ht_salary);
        }
        void getdesignation(String hd){
                designation=hd;
                System.out.println("Designation of hourly employee is : "+hd);
        }
}
class w_employee extends contract_emp{
        float weeks;
        float wpay;
        float wt_salary;
        void calculate_wages(float w,float wp){
                weeks=w;
                wpay=wp;
                wt_salary=weeks*wpay*4;
                System.out.println("total monthly salary of hourly employee is : "+wt_salary);
        }
        void getdesignation(String wd){
                designation=wd;
                System.out.println("Designation of hourly employee is : "+wd);
        }
}
class wages{
        public static void main(String []args){
        Scanner s=new Scanner(System.in);
        System.out.println("enter the designation");
        String m_des=s.next();
        System.out.println("enter hourly pay");
        float hourly_pay=s.nextFloat();
        System.out.println("enter no of hours the employee worked in a month");
        float hours_worked=s.nextFloat();
        System.out.println("enter the designation of hourly employee");
        String h_des=s.next();
        System.out.println("enter weekly pay");
        float weekly_pay=s.nextFloat();
        System.out.println("enter no of weeks the employee worked in a month");
        float weeks_worked=s.nextFloat();
        System.out.println("enter the designation of weekly employee");
        String w_des=s.next();
        contract_emp z=new contract_emp();
        z.getdesignation(m_des);
        h_employee x=new h_employee();
        x.calculate_wages(hourly_pay,hours_worked);
        x.getdesignation(h_des);
        w_employee y=new w_employee();
        y.calculate_wages(weekly_pay,weeks_worked);
```

```
        y.getdesignation(w_des);
        }
}
```

OUTPUT:

```
:\Users\abhir\Desktop>javac wages.java

:\Users\abhir\Desktop>java wages
nter the designation
m
nter hourly pay
00
nter no of hours the employee worked in a month
00
nter the designation of hourly employee
m
nter weekly pay
000
nter no of weeks the employee worked in a month

nter the designation of weekly employee
rg
esignation is :gm
otal monthly salary of hourly employee is : 1.602E7
esignation of hourly employee is : gm
otal monthly salary of hourly employee is : 36000.0
esignation of hourly employee is : erg

:\Users\abhir\Desktop>
```

5. Write an application to create a super class Vehicle with information vehicle number, insurance number, color and methods getConsumption() displayConsumption(). Derive the sub-classes TwoWheeler and FourWheeler with method maintenance() and average() to print the maintenance And average of vehicle.

PROGRAM:

```
import java.util.Scanner;
class vehicle{
        int veh_no;
        int in_no;
        String veh_colour;
        double fuel;
        void info(int veh_no,int in_no,String veh_colour)
        {
        this.veh_no=veh_no;
        this.in_no=in_no;
        this.veh_colour=veh_colour;
        }
        void getconsumption(double fuel)
        {
        this.fuel=fuel;
        }
        void displayconsumption()
        {
        System.out.println("fuel consumption :"+fuel);
        }
        void display()
        {
        System.out.println("vehicle num :"+veh_no);
        System.out.println("vehicle insurance :"+in_no);
        System.out.println("vehicle colour :"+veh_colour);
```

```
        }
}
class two_wheeler extends vehicle{
        double avrg;
        double mtnce;
        void two_info(double avrg,double mtnce)
        {
        this.avrg=avrg;
        this.mtnce=mtnce;
        }
        void average()
        {
        System.out.println("average of two wheeler is"+avrg);
        }
        void maintainance()
        {
        System.out.println("Maintainance of two wheeler is"+mtnce);
        }
}
class four_wheeler extends vehicle{
        double avrg;
        double mtnce;
        void four_info(double avrg,double mtnce)
        {
        this.avrg=avrg;
        this.mtnce=mtnce;
        }
        void average()
        {
        System.out.println("average of three wheeler is"+avrg);
        }
        void maintainance()
        {
        System.out.println("Maintainance of three wheeler is"+mtnce);
        }
}
class vehicle_program{
        public static void main(String []args)
        {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter vehicle number, insurance number, colour, fuel, avrg, maintainance for
two wheeler");
        int veh_no=sc.nextInt();
        int in_no=sc.nextInt();
        String veh_colour=sc.next();
        Double fuel=sc.nextDouble();
        double avg=sc.nextDouble();
        double mtnce=sc.nextDouble();
        System.out.println("Enter vehicle number, insurance number, colour, fuel, avrg, maintainance for
four wheeler");
        int veh_n=sc.nextInt();
        int in_n=sc.nextInt();
        String veh_clr=sc.next();
        Double t_fuel=sc.nextDouble();
        double avrg=sc.nextDouble();
        double mtne=sc.nextDouble();
```

```
        two_wheeler tw= new two_wheeler();
        tw.info(veh_no,in_no,veh_colour);
        tw.getconsumption(fuel);
        tw.two_info(avg,mtnce);
        System.out.println("for two wheelers");
        tw.display();
        tw.displayconsumption();
        tw.maintainance();
        tw.average();
        four_wheeler thw= new four_wheeler();
        thw.info(veh_n,in_n,veh_clr);
        thw.getconsumption(t_fuel);
        thw.four_info(avrg,mtne);
        System.out.println("for three wheelers");
        thw.display();
        thw.displayconsumption();
        thw.maintainance();
        thw.average();
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>java vehicle_program
Enter vehicle number, insurance number, colour, fuel, avrg, maintainance for two wheeler
35542
4563423
red
42
40
34424
Enter vehicle number, insurance number, colour, fuel, avrg, maintainance for four wheeler
34242
435224
red
335
245
34525
for two wheelers
vehicle num :35542
vehicle insurance :4563423
vehicle colour :red
fuel consumption :42.0
Maintainance of two wheeler is34424.0
average of two wheeler is40.0
for three wheelers
vehicle num :34242
vehicle insurance :435224
vehicle colour :red
fuel consumption :335.0
Maintainance of three wheeler is34525.0
average of three wheeler is245.0

C:\Users\abhir\Desktop>
```

6. Extend the above TwoWheeler class with methods getType() and getName() which gives the information about the type and the name of the company.Create sub-classes Geared and NonGeared with method average() to print the average of a geared and non-geared two wheeler.

PROGRAM:

```
import java.util.Scanner;
class vehicle
{
        int vehno,insno;
        String colour;
        double fuel;
        void info(int vehno,int insno,String colour)
```

```
	{
		this.vehno=vehno;
		this.insno=insno;
		this.colour=colour;
	}
	void getconsumption(double fuel)
	{
		this.fuel=fuel;
	}
	void displayconsumption()
	{
		System.out.println("Fuel consumption : "+fuel);
	}
	void displayinfo()
	{
		System.out.println("Vehicle number : "+vehno);
		System.out.println("Vehicle insurence no :"+insno);
		System.out.println("Vehicle colour : "+colour);
	}
}
class twowheeler extends vehicle
{
	double avg,maintain;
	void specific(double avg,double maintain)
	{
	this.avg=avg;
	this.maintain=maintain;
	}
	double getmaintainence()
	{
	return maintain;
	}
	double getaverage()
	{
	return avg;
	}
}
class geared extends twowheeler
{
	String name;
	String type;
	geared(String type,String name)
	{
	this.type=type;
	this.name=name;
	}
	String gettype()
	{
	return type;
	}
	String getname()
	{
	return name;
	}
}
class nongeared extends twowheeler
```

```
{
String type;
String name;
nongeared(String type,String name)
{
	this.type=type;
	this.name=name;
	}
	String gettype()
	{
	return type;
	}
	String getname()
	{
	return name;
	}
}
class vehiclemain1
{
	public static void main(String []args)
	{
	//Veicle extends to twowheeler
	twowheeler tw=new twowheeler();
	Scanner sc=new Scanner(System.in);
	System.out.println("Enter the two wheeler vehicle number : ");
	int vehnot=sc.nextInt();
	System.out.println("Enter the two vehicle wheeler insurance number : ");
	int insnot=sc.nextInt();
	System.out.println("Enter the colour of two wheeler vehicle : ");
	String colourt=sc.next();
	tw.info(vehnot,insnot,colourt);
	System.out.println("Enter the fuel consumed by two wheeler vehicle : ");
	double fuelt=sc.nextDouble();
	System.out.println("Enter the average of two wheeler vehicle : ");
	double avgt=sc.nextDouble();
	System.out.println("Enter the maintainence of two wheeler vehicle : ");
	double maintainencet=sc.nextDouble();
	tw.getconsumption(fuelt);
	tw.specific(avgt,maintainencet);
	System.out.println("Enter the name of geared two wheeler vehicle : ");
	String nameg=sc.next();
	//extends to geared
	geared g=new geared("Geared",nameg);
	System.out.println("Enter the name of nongeared two wheeler vehicle : ");
	String nameng=sc.next();
	//Extends to nongeared
	nongeared ng=new nongeared("Nongeared",nameng);
	System.out.println("For Two Wheelers");
	tw.displayinfo();
	tw.displayconsumption();
	System.out.println("Maintainenece : "+tw.getmaintainence());
	System.out.println("Average : "+tw.getaverage());
	System.out.println("Type : "+g.gettype());
	System.out.println("Name of gearerd : "+g.getname());
	System.out.println("Type : "+ng.gettype());
	System.out.println("Name of non geared : "+ng.getname());
```

```
    }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac vehiclemain1.java

C:\Users\abhir\Desktop>java vehiclemain1
Enter the two wheeler vehicle number :
34242
Enter the two vehicle wheeler insurance number :
345326
Enter the colour of two wheeler vehicle :
red
Enter the fuel consumed by two wheeler vehicle :
334
Enter the average of two wheeler vehicle :
45
Enter the maintainence of two wheeler vehicle :
4532
Enter the name of geared two wheeler vehicle :
4
Enter the name of nongeared two wheeler vehicle :
5
For Two Wheelers
Vehicle number : 34242
Vehicle insurence no :345326
Vehicle colour : red
Fuel consumption : 334.0
Maintainenece : 4532.0
Average : 45.0
Type : Geared
Name of gearerd : 4
Type : Nongeared
Name of non geared : 5

C:\Users\abhir\Desktop>
```

Week-VII

1. Create an abstract class Shape which calculate the area and volume of 2-d and 3-d shapes with methods getArea() and getVolume(). Reuse this class to calculate the area and volume of square ,circle ,cube and sphere.

PROGRAM:

```
import java.util.Scanner;
import static java.lang.Math.*;
abstract class Shape{
	abstract double getArea();
	abstract double getVolume();
}
class Square extends Shape{
	double s;
	Square(double s){
		this.s=s;
	}
	double getArea(){return s*s;}
	double getVolume(){return 0;}
}
class Circle extends Shape{
	double r;
	Circle(double r){
		this.r=r;
	}
	double getArea(){return PI*r*r;}
	double getVolume(){return 0;}
}
class Cube extends Shape{
	double s;
	Cube(double s)
	{
		this.s=s;
	}
	double getArea(){return 6*s*s;}
	double getVolume(){return s*s*s;}
}
class Sphere extends Shape{
	double r;
	Sphere(double r){
		this.r=r;
	}
	double getArea(){return 4*PI*r*r;}
	double getVolume(){return 4.0/3*PI*r*r*r;}
}
class AreaVolume{
	public static void main(String[] args){
		Scanner in=new Scanner(System.in);
		System.out.print("Enter side of square : ");
		Square sq=new Square(in.nextDouble());
		System.out.print("Enter radius of circle : ");
		Circle ci=new Circle(in.nextDouble());
```

```
                System.out.print("Enter side of cube : ");
                Cube cu=new Cube(in.nextDouble());
                System.out.print("Enter radius of sphere : ");
                Sphere sp=new Sphere(in.nextDouble());
                System.out.println("Circle : area = "+ci.getArea()+"  volume = "+ci.getVolume());
                System.out.println("Square : area = "+sq.getArea()+"  volume = "+sq.getVolume());
                System.out.println("Cube   : area = "+cu.getArea()+"  volume = "+cu.getVolume());
                System.out.println("Sphere : area = "+sp.getArea()+"  volume = "+sp.getVolume());
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac AreaVolume.java

C:\Users\abhir\Desktop>java AreaVolume
Enter side of square : 3
Enter radius of circle : 5
Enter side of cube : 4
Enter radius of sphere : 2
Circle : area = 78.53981633974483  volume = 0.0
Square : area = 9.0  volume = 0.0
Cube   : area = 96.0  volume = 64.0
Sphere : area = 50.26548245743669  volume = 33.510321638291124

C:\Users\abhir\Desktop>
```

2. Create an abstract class Employee with methods getAmount() which displays the amount paid to employee. Reuse this class to calculate the amount to be paid to WeeklyEmployeed and HourlyEmployee according to no. of hours and total hours for HourlyEmployee and no. of weeks and total weeks for WeeklyEmployee.

PROGRAM:

```
import java.util.Scanner;
abstract class Employee{
        abstract double getAmount();
}
class HourlyEmployee extends Employee{
        double h,sh;
        HourlyEmployee(double h,double sh){
                this.h=h;
                this.sh=sh;
        }
        double getAmount(){return h*sh;}//method signature must be same
}
class WeeklyEmployee extends Employee{
        double w,sw;
        WeeklyEmployee(double w, double sw){
                this.w=w;
                this.sw=sw;
        }
        double getAmount(){return w*sw;}
}
class EmployeeAmount{
```

```
        public static void main(String[] args){
                Scanner in=new Scanner(System.in);
                System.out.println("Enter 1 for HourlyEmployee and 2 for WeeklyEmployee");
                int op=in.nextInt();
                if (op==1){
                        System.out.println("Enter total hours worked and salary per hour");
                        var h=new HourlyEmployee(in.nextDouble(),in.nextDouble());
                        System.out.println("amount to be paid "+h.getAmount());
                }
                if (op==2){
                        System.out.println("Enter total weeks worked and salary per week");
                        var w=new WeeklyEmployee(in.nextDouble(),in.nextDouble());
                        System.out.println("Amount to be paid "+w.getAmount());
                }
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac EmployeeAmount.java

C:\Users\abhir\Desktop>java EmployeeAmount
Enter 1 for HourlyEmployee and 2 for WeeklyEmployee
1
Enter total hours worked and salary per hour
444
424
amount to be paid 188256.0

C:\Users\abhir\Desktop>
```

3. Create an Interface payable with method getAmount (). Calculate the amount to be paid to Invoice and Employee by implementing Interface.

PROGRAM:

```
import java.util.Scanner;
interface Payable
{
        int getAmount(int n);  //public abstract by default//also cann't have constructors since variables will be public Sstatic fina
}
class Employee implements Payable
{
        int salary;
        Employee(int s)
        {
                salary=s;
        }
        public int getAmount(int months) //since default modifier more restrictive than public
```

```
	{
		return salary*months;
	}
}
class Invoice
{
	public int getAmount(int n) //since default modifier more restrictive than public
	{
		int bill=0;
		Scanner in=new Scanner(System.in);
		System.out.println("Enter the number of items,amount for each type of item ");
		for(int i=0;i<n;i++)
			bill=bill+in.nextInt()*in.nextInt();
		return bill;
	}
}
class EmployeeAmountInterface
{
	public static void main(String[] args)
	{
		Scanner in=new Scanner(System.in);
		System.out.println("Enter 1 for Employee and 2 for Invoice");
		int op=in.nextInt();
		if (op==1)
		{
			System.out.println("Enter salary/month and total months");
			int s=in.nextInt(),n=in.nextInt();
			Employee e=new Employee(s);
			System.out.println("Amount to be paid = "+e.getAmount(n));
		}
		else if(op==2)
		{
			Invoice i=new Invoice();
			System.out.println("Enter number of types of items");
			System.out.println("Total bill is = "+i.getAmount(in.nextInt()));
		}
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>java EmployeeAmountInterface
Enter 1 for Employee and 2 for Invoice
2
Enter number of types of items
3
Enter the number of items,amount for each type of item
3 43
4 535
2 44
Total bill is = 2357
```

4. Create an Interface Vehicle with method getColor(),getNumber(), getConsumption() calculate the fuel consumed, name and color for TwoWheeler and Four Wheeler By implementing interface Vehicle.

PROGRAM:

```
import java.util.Scanner;
interface Vehicle{
	String getName(); //public abstract by default
	String getColour();
	String getNumber();
	double getConsumption();
}
class TwoWheeler implements Vehicle{
	String name,colour,number;
	double mileage;
	TwoWheeler(String na,String co,String nu,double mi){
		name=na;
		colour=co;
		number=nu;
		mileage=mi;
	}
	public String getName(){return name;} //default access modifier causes error
	public String getColour(){return colour;}
	public String getNumber(){return number;}
	public double getConsumption()
	{
		return 1/mileage;
	}
}
class FourWheeler implements Vehicle
{
	String name,colour,number;
	double mileage;
	FourWheeler(String na,String co,String nu,double mi)
	{
		name=na;
		colour=co;
		number=nu;
		mileage=mi;
	}
	public String getName(){return name;}
	public String getColour(){return colour;}
	public String getNumber(){return number;}
	public double getConsumption()
	{
		return 1/mileage;
	}
}
class VehicleMain
{
	public static void main(String[] args)
	{
	Scanner in=new Scanner(System.in);
	System.out.println("Enter name,color,number and mileage for twowheeler");
```

```
        TwoWheeler t=new TwoWheeler(in.next(),in.next(),in.next(),in.nextDouble());
        System.out.println("Enter name,color,number and mileage for fourwheeler");
        FourWheeler f=new FourWheeler(in.next(),in.next(),in.next(),in.nextDouble());
        System.out.println("Details of twowheeler");
        System.out.println(t.getName()+"  "+t.getNumber()+"   "+t.getColour()+"
"+t.getConsumption()+"ltr/km");
        System.out.println("Details of fourwheeler");
        System.out.println(f.getName()+"  "+f.getNumber()+"   "+f.getColour()+"
"+f.getConsumption()+"ltr/km");
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac EmployeeAmountInterface.java

C:\Users\abhir\Desktop>javac VehicleMain.java

C:\Users\abhir\Desktop>java VehicleMain
Enter name,color,number and mileage for twowheeler
pulsur red 3213 60
Enter name,color,number and mileage for fourwheeler
cryasta black 424 24
Details of twowheeler
pulsur  3213   red    0.016666666666666666ltr/km
Details of fourwheeler
cryasta  424   black    0.041666666666666664ltr/km

C:\Users\abhir\Desktop>
```

5. Create an Interface Fare with method getAmount() to get the amount paid for fare of travelling. Calculate the fare paid by bus and train implementing interface Fare.

PROGRAM:

```
import java.util.Scanner;
interface Fare
{
        double tfare=0.5; //rupees/km       public static final as they are fixed
        double bfare=1;
        double getAmount(int d);
}
class Bus implements Fare
{
        public double getAmount(int d)//must be public
        {
                return bfare*d;
        }
}
class Train implements Fare
{
        public double getAmount(int d)
        {
                return tfare*d;
        }
}
```

```
class TravellingFare
{
	public static void main(String[] args)
	{
		Scanner in=new Scanner(System.in);
		Bus b=new Bus();
		Train t=new Train();
		System.out.println("Enter the distance travelled by bus and train in km");
		System.out.println("Fare for bus is "+b.getAmount(in.nextInt())+" rupees");
		System.out.println("Fare for train is "+t.getAmount(in.nextInt())+" rupees");
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac TravellingFare.java

C:\Users\abhir\Desktop>java TravellingFare
Enter the distance travelled by bus and train in km
4242
Fare for bus is 4242.0 rupees
4342
Fare for train is 2171.0 rupees
```

6. Create an Interface StudentFee with method getAmount(),getFirstName(),getLastName(), getAddress(), getContact(). Calculate the amount paid by the Hostler and NonHostler student by implementing interface Student Fee.

PROGRAM:

```
import java.util.Scanner;
interface StudentFee
{
	int nfee=30000,hfee=50000;//public statc final
	int getAmount();
	String getFirstName();//public abstract
	String getLastName();
	String getAddress();
	String getContact();
}
class NonHostler implements StudentFee
{
	String firstname,lastname,address,contact;
	NonHostler(String fn,String ln,String a,String c)
	{
		firstname=fn;
		lastname=ln;
		address=a;
		contact=c;
	}
	public int getAmount(){return nfee;}//must be public as we are overriding a public method
```

```
        // also must be same or less restrictivje
        public String getFirstName(){return firstname;}
        public String getLastName(){return lastname;}
        public String getAddress(){return address;}
        public String getContact(){return contact;}
}
class Hostler implements StudentFee
{
        String firstname,lastname,address,contact;
        Hostler(String fn,String ln,String a,String c)
        {
                firstname=fn;
                lastname=ln;
                address=a;
                contact=c;
        }
        public int getAmount(){return hfee;}
        public String getFirstName(){return firstname;}
        public String getLastName(){return lastname;}
        public String getAddress(){return address;}
        public String getContact(){return contact;}
}
class Student
{
        public static void main(String args[])
        {
                Scanner in=new Scanner(System.in);
                System.out.println("Enter firstname,lastname,address and contact details of nonhostler");
                NonHostler n=new NonHostler(in.next(),in.next(),in.next(),in.next());
                System.out.println("Enter firstname,lastname,address and contact details of hostler");
                Hostler h=new Hostler(in.next(),in.next(),in.next(),in.next());
                System.out.println("\n\nDetails of nonhostler");
                System.out.println(n.getFirstName()+" "+n.getLastName()+"    "+n.getAddress()+"
"+n.getContact());
                System.out.println("amount to be paid "+n.getAmount());
                System.out.println("\n\nDetails of hostler");
                System.out.println(h.getFirstName()+" "+h.getLastName()+"    "+h.getAddress()+"
"+h.getContact());
                System.out.println("amount to be paid "+h.getAmount());
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac Student.java

C:\Users\abhir\Desktop>java Student
Enter firstname,lastname,address and contact details of nonhostler
abhiram
rote
hyd
827474298
Enter firstname,lastname,address and contact details of hostler
ram
reddy
hyd
8472848


Details of nonhostler
abhiram rote    hyd    827474298
amount to be paid 30000


Details of hostler
ram reddy    hyd    8472848
amount to be paid 50000

C:\Users\abhir\Desktop>_
```

Week-VIII

1. Write a Program to create your own package. Package should have more than two classes. Write a Program that uses the classes from the package.

PROGRAM:

```
//creating department class in VARUN package (file-1)
package VARUN;
public class Dept{
	public void details(){
		System.out.println("Department of ECE, RGUKT-BASAR");
	}
}

//creating Year class in VARUN package (file-2)
package VARUN;
public class Year{
	public void details(){
		System.out.println("Engineering 2nd Year");
	}
}

//importing all required classes and using in program (file-3)
import VARUN.Dept;
import VARUN.Year;
class MyDetails
{
	public static void main(String args[])
	{
		Dept v1=new Dept();
		Year v2=new Year();
		v2.details();
		v1.details();
	}
}
```

OUTPUT:

```
:\Users\abhir\Desktop>javac -d . Dept.java

:\Users\abhir\Desktop>javac MyDetails.java

:\Users\abhir\Desktop>java MyDetails
ngineering 2nd Year
epartment of ECE, RGUKT-BASAR

:\Users\abhir\Desktop>
```

2. Create a package named org.shapes. Create some classes in the package representing some common geometric shapes like Square, Triangle, Circle and so on. write a Program that uses the classes from the package.

PROGRAM:

```
package org.shapes;
public class triangle{
	public double area(int length , int breadth){
		double area=0.5*length*breadth;
		System.out.println("Area of the triangle is "+area);
		return area;
	}
	public int perimeter(int a, int b, int c){
		int perimeter=a+b+c;
		System.out.println("Perimenter of a triangle is "+perimeter);
		return perimeter;
	}
}

package org.shapes;
public class circle{
	public double area(int radius){
		double area =Math.PI*radius*radius;
		System.out.println("area of a circle is "+area);
		return area;
	}
	public double perimeter(int radius ){
		double perimeter = 2*Math.PI*radius;
		System.out.println("perimeter of a circle is "+perimeter);
		return perimeter;
	}
}

package org.shapes;
public class rectangle{
public int area(int a, int b){
	int area =a*b;
	System.out.println("area of rectangle is "+area);
	return area;
	}
	public int perimeter(int a,int b ){
		int perimeter= 2*(a+b);
		System.out.println("perimeter of rectangle is "+perimeter);
		return perimeter;
	}
}

package org.shapes;
public class square{
	public int area (int side ){
		int area=side*side;
		System.out.println("area of a square of side "+side +" is  "+area );
		return area;
	}
```

```
        public double perimeter(int side){
                double perimeter=4*side;
                System.out.println("perimeterof a square  of side "+side +" is "+perimeter );
                return perimeter;
        }
}

import org.shapes.square;
import org.shapes.triangle;
import org.shapes.circle;
import org.shapes.rectangle;
class values{
        public static void main(String [] args){
                square a= new square();
                a.area(4);
                a.perimeter(4);
                triangle b= new triangle();
                b.area(3,5);
                b.perimeter(3,4,5);
                circle c=new circle();
                c.area(7);
                c.perimeter(7);
                rectangle d= new rectangle();
                d.area(3,6);
                d.perimeter(7,8);
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac -d . circle.java

C:\Users\abhir\Desktop>javac -d . square.java

C:\Users\abhir\Desktop>javac -d . rectangle.java

C:\Users\abhir\Desktop>javac values.java

C:\Users\abhir\Desktop>java values
area of a square of side 4 is  16
perimeterof a square  of side 4 is 16.0
Area of the triangle is 7.5
Perimenter of a triangle is 12
area of a circle is 153.93804002589985
perimeter of a circle is 43.982297150257104
area of rectangle is 18
perimeter of rectangle is 30

C:\Users\abhir\Desktop>
```

3. Write a Java program to create package called dept. Create four classes as CSE, ECE, ME and CE add methods in each class which can display subject names of your respect year. access this package classes from main class.

PROGRAM:

```
//creating ECE subjects in dept package (file-1)
package dept;
public class ECE{
	public void subjects(){
		System.out.println("ECE 2nd year subjects
are\nEDC\nAC\nTCCA\nSS\nEDC\nPTSP\nEMT\nCS");
	}
}

//creating CSE subjects in dept package (file-2)
package dept;
public class CSE{
	public void subjects(){
		System.out.println("CSE 2nd year subjects
are\nDS\nDEC\nAEC\nPS\nDM\nDAA\nDA\nDBMS\nCOA");
	}
}

//creating ME subjects in dept package (file-3)
package dept;
public class ME{
	public void subjects(){
		System.out.println("ME 2nd year subjects are\nED\nGraphics");
	}
}

//creating CE subjects in dept package (file-4)
package dept;
public class CE{
	public void subjects(){
		System.out.println("CE 2nd year subjects are\nMechanics\nFluedMechanics");
	}
}

//importing all dept classes and using in program (file-5)
import dept.ECE;
import dept.CSE;
import dept.ME;
import dept.CE;
class DeptSubs
{
	public static void main(String args[])
	{
		ECE S1=new ECE();
		CSE S2=new CSE();
		ME S3=new ME();
		CE S4=new CE();
		S1.subjects();
		S2.subjects();
		S3.subjects();
```

```
                S4.subjects();
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac -d . ECE.java

C:\Users\abhir\Desktop>javac -d . CSE.java

C:\Users\abhir\Desktop>javac -d . ME.java

C:\Users\abhir\Desktop>javac -d . CE.java

C:\Users\abhir\Desktop>javac DeptSubs.java

C:\Users\abhir\Desktop>java DeptSubs
ECE 2nd year subjects are
EDC
AC
TCCA
SS
EDC
PTSP
EMT
CS
CSE 2nd year subjects are
DS
DEC
AEC
PS
DM
DAA
DA
DBMS
COA
ME 2nd year subjects are
ED
Graphics
CE 2nd year subjects are
Mechanics
FluedMechanics

C:\Users\abhir\Desktop>
```

4. Write a Calculator program : Include all calculator operations in as classes in a Package “Calculator” and import in to main class.

PROGRAM:

```
//creating calculator package (file-1)
package calculator;
public class operations{
        public int add(int a,int b){
                return a+b;
        }
        public int sub(int a,int b){
                return a-b;
        }
        public double mul(int a,int b){
                return a*b;
        }
```

```
	public double div(int a, int b){
		return a/b;
	}
}

//importing calculatorpackage's class and using it (file-2)
import java.io.*;
import java.util.Scanner;
import calculator.operations;
class calc{
	public static void main(String [] args){
		operations v= new operations();
		Scanner vn=new Scanner(System.in);
		System.out.print("Enter 2 numbers to perform operations :");
		int a=vn.nextInt();
		int b=vn.nextInt();
		System.out.println("\nAdditon of given numbers "+v.add(a,b));
		System.out.println("Subtraction of given numbers "+v.sub(a,b));
		System.out.println("Multiplication of given numbers "+v.mul(a,b));
		System.out.println("Division of given numbers "+v.div(a,b));
	}
}
```

OUTPUT:

```
:\Users\abhir\Desktop>javac -d . operations.java

:\Users\abhir\Desktop>javac calc.java

:\Users\abhir\Desktop>java calc
nter 2 numbers to perform operations :3


dditon of given numbers 5
ubtraction of given numbers 1
ultiplication of given numbers 6.0
ivision of given numbers 1.0

:\Users\abhir\Desktop>
```

5. Write a program for the following
a. Example to use interfaces in Packages.

PROGRAM:

Main File:

```
import java.util.Scanner;
import static java.lang.Math.*;
import intpack.*;
//even if it is not compiled the compiler compiles it while importing
//thats why .java files should have same name as public class in a file
class Circle implements Measurable
{
	double r;
	public Circle(double r)
	{
		this.r=r;
	}
```

```
        public double getArea(){return PI*r*r;}
}
class InterfacePackageMain
{
        public static void main(String[] args)
        {
                Scanner in=new Scanner(System.in);

                System.out.println("Enter radius of circle");
                Circle c=new Circle(in.nextDouble());
                System.out.println("area of circle "+c.getArea());
        }
}
```

//File 2:

```
package intpack;
public interface Measurable
{
        public double getArea();
}
```

OUTPUT:

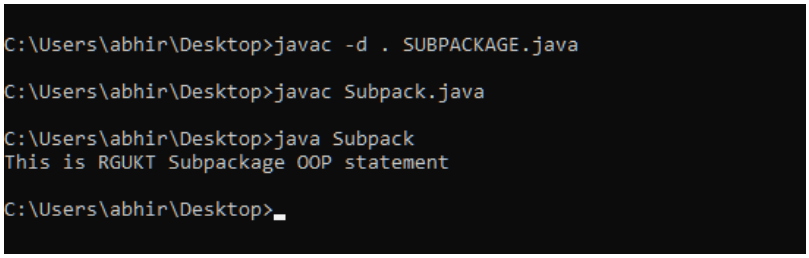
```
C:\Users\abhir\Desktop>javac InterfacePackageMain.java

C:\Users\abhir\Desktop>java InterfacePackageMain
Enter radius of circle
5
area of circle 78.53981633974483
```

b. Example to create sub package in a package.

PROGRAM:

```
// RGUKT is main package, ECE E2 OOPS are sub packages
package RGUKT.ECE.E2.OOP;
public class SUBPACKAGE
{
        public void show_status()
        {
                System.out.println("This is RGUKT Subpackage OOP statement");
        }
}

//accessing the user defined package RGUKT
import RGUKT.ECE.E2.OOP.SUBPACKAGE;
class Subpack
{       public static void main(String args[])
        {
                SUBPACKAGE E2=new SUBPACKAGE();
                E2.show_status();
        }
}
```

OUTPUT:



Week-IX

1. Program for demonstrating the use of throw, throws & finally - Create a class with a main() that throws an object of class Exception inside a try block. Give the constructor for Exception a String argument. Catch the exception inside a catch clause and print the String argument. Add a finally clause and print a message to prove you were there.

PROGRAM:

```
import java.io.IOException;
import java.util.Scanner;
class W91
{
	static String s;
	W91(String s)
	{
		this.s=s;
	}
	public static void main(String args[]) throws IOException
	{
		String a;
		Scanner in=new Scanner(System.in);
		System.out.println("Enter a sting you want");
		a=in.nextLine();
		W91 m=new W91(a);
		try
		{
			throw new IOException("exception occured");
		}
		catch(Exception e)
		{
			System.out.println(e);
			System.out.println("exception is handeled in catch block");
			System.out.println("the string u entered is  "+s);
		}
		finally
		{
			System.out.println("This is finally block");
		}
	}
}
```

OUTPUT:

```
:\Users\abhir\Desktop>java W91
nter a sting you want
p
ava.io.IOException: exception occured
xception is handeled in catch block
he string u entered is  hp
his is finally block

:\Users\abhir\Desktop>
```

2. Write a program that shows that the order of the catch blocks is important. If you try to catch a superclass exception type before a subclass type, the compiler should generate errors.

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
class Super
{
	int met1(int a, int b)
	{
		return a/b;
	}
}
class Sub extends Super
{
	int met2(String Str)
	{
		return Str.length();
	}
}
class orderofcatchExcMAIN
{
	public static void main(String args[])
	{
		Scanner in=new Scanner(System.in);
		int m,n;
		String j;
		System.out.println("Enter any two numbers");
		m=in.nextInt();
		n=in.nextInt();
		j=null;
		Super su1=new Super();
		Sub su2=new Sub();
		try
		{
			System.out.println(m+" divided by "+n+" is "+su1.met1(m,n));
			su2.met2(j);
		}
		catch(NullPointerException e)
		{
			System.out.println(e);
			System.out.println("subclass exception occurs");
		}
```

```
		catch(ArithmeticException e)
		{
			System.out.println(e);
			System.out.println("super class exception");
		}
		catch(Exception e)
		{
			System.out.println("Something went wrong");
		}
		System.out.println("remaining code");
	}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac orderofcatchExcMAIN.java

C:\Users\abhir\Desktop>java orderofcatchExcMAIN
Enter any two numbers
42 53
42 divided by 53 is 0
java.lang.NullPointerException: Cannot invoke "String.length()" because "<parameter1>" is null
subclass exception occurs
remaining code

C:\Users\abhir\Desktop>java orderofcatchExcMAIN
Enter any two numbers
2 0
java.lang.ArithmeticException: / by zero
super class exception
remaining code

C:\Users\abhir\Desktop>
```

3. Write a program to rethrow an exception – Define methods one() & two(). Method two() should initially throw an exception. Method one() should call two(), catch the exception and rethrow it Call one() from main() and catch the rethrown.

PROGRAM:

```
class Week93rethrow
{
	static void two()
	{
		int[] varun=new int[3];
		varun[5]=9;
	}
	static void one()
	{
		try
		{
			two();
		}
		catch(Exception v)
		{
			System.out.println("This is Exception in catch of method two()");
			System.out.println(v+"\n");
			throw v;
		}
	}
```

```
        public static void main(String args[])
        {
                try
                {
                        one();
                }
                catch(IndexOutOfBoundsException v)
                {
                        System.out.println("This is Exception in catch of method one()");
                        System.out.println(v);
                }
                finally
                {
                        System.out.println("\nThis is final statement");
                }
        }
}
```

OUTPUT:

```
:\Users\abhir\Desktop>javac Week93rethrow.java

:\Users\abhir\Desktop>java Week93rethrow
his is Exception in catch of method two()
ava.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 3

his is Exception in catch of method one()
ava.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 3

his is final statement

:\Users\abhir\Desktop>
```

4. Exception Handling program for ClassNotFoundException--thrown if a program can not find a class it depends at runtime (i.e., the class's ".class" file cannot be found or was removed from the CLASSPATH).

PROGRAM:

```
class Classexc
{
   public static void main(String[] args) {
      try{
         Class.forName("Class1");   // Class1 is not found as it is deleted
      }
      catch(ClassNotFoundException e){
         System.out.println(e);
         System.out.println("Class Not Found...");
      }
```

```
    }
}
```

OUTPUT:

```
:\Users\abhir\Desktop>javac Classexc.java

:\Users\abhir\Desktop>java Classexc
ava.lang.ClassNotFoundException: Class1
lass Not Found...

:\Users\abhir\Desktop>
```

5. Exception Handling program for NumberFormatException--thrown if a program is attempting to convert a string to a numerical datatype, and the string contains inappropriate characters (i.e. 'z' or 'Q').

PROGRAM:

```
class  NumtoStrExc{
    public static void main(String args[]){
        try {
            int num = Integer.parseInt("Varun"); //Varun is a String type
            System.out.println(num);
        }
        catch(NumberFormatException e) {
            System.out.println("Number format exception(Can't convert String to Integer)");
        }
    }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac NumtoStrExc.java

C:\Users\abhir\Desktop>java NumtoStrExc
Number format exception(Can't convert String to Integer)

C:\Users\abhir\Desktop>
```

6. Create your own exception class using the extends keyword. Write a constructor for this class that takes a String argument and stores it inside the object with a String reference. Write a method that prints out the stored String. Create a try- catch clause to exercise your new exception.

PROGRAM:

```
import java.io.*;
import java.util.Scanner;
class MyOwnException extends Exception{
        String var;
        MyOwnException(String Message){
                var=Message;
        }
        void printstring_method(){
                System.out.println(var);
```

```
        }
}
class Exceptionextends{
        public static void main(String args[]){
                int a;
                System.out.print("Enter a number greater than 3 : ");
                Scanner vn=new Scanner(System.in);
                a=vn.nextInt();
                try{
                        if(a<=3)
                                throw new MyOwnException("Given data is improper");
                        else
                                System.out.println("Given data is proper. No Exception occured");
                }
                catch(MyOwnException e){
                        e.printstring_method();
                }
                finally{
                        System.out.println("This is finally Block");
                }
        }
}
```

OUTPUT:

```
:\Users\abhir\Desktop>javac Exceptionextends.java

:\Users\abhir\Desktop>java Exceptionextends
nter a number greater than 3 : 23
iven data is proper. No Exception occured
his is finally Block

:\Users\abhir\Desktop>java Exceptionextends
nter a number greater than 3 : 1
iven data is improper
his is finally Block

:\Users\abhir\Desktop>
```

Week-X

1. Write a program to create MyThread class with run() method and then attach a thread to this MyThread class object.

PROGRAM:

```
import java.io.*;
class MyThread extends Thread{
        public void run() {
                System.out.println("Thread is running");
        }
}
class runthread{
        public static void main(String args[]) {
                MyThread m=new MyThread();
                m.start();
        }
```

```
}
```

OUTPUT:

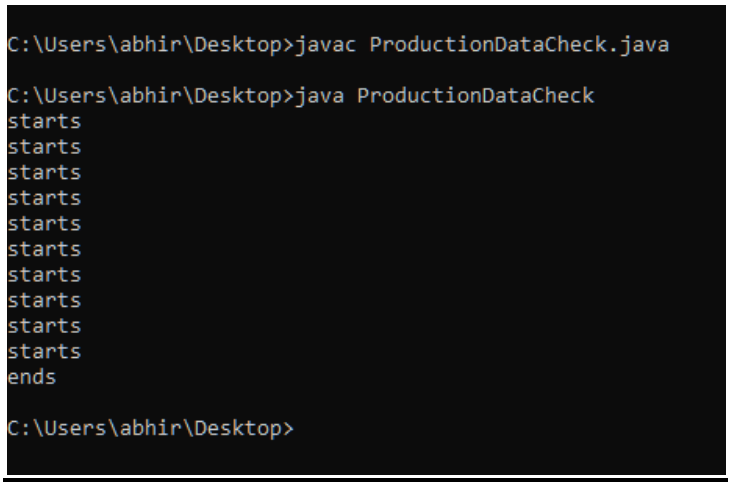
```
:\Users\abhir\Desktop>javac runthread.java

:\Users\abhir\Desktop>java runthread
hread is running

:\Users\abhir\Desktop>
```

2. Write a program where the consumer thread checks the data production status [is over or not] for every 10 ms.

PROGRAM:

```
import java.io.*;
class ConsumerThread extends Thread
{
public void run()
{
for(int i=1;i<=10;i++)
{
try
{
Thread.sleep(10);
}
catch(InterruptedException e)
{
System.out.println("Exception caught");
}
System.out.println("starts ");
}
System.out.println("ends");
}
}
class ProductionDataCheck
{
public static void main(String args[])
{
ConsumerThread obj =new ConsumerThread();
obj.start();
}
}
```

OUTPUT:



3. Write a Program using Threads to simulate a traffic light. The Signal lights should glow after each 10 second, one by one. For example: Firstly Red, then after 10 seconds, red will be put to off and yellow will start glowing and then accordingly green.

PROGRAM:

```
import java.util.Scanner;
class TrafficSignal extends Thread
{
int n;
TrafficSignal(int n)
{
this.n=n;
}
public void run()
{
for (int i=0;i<n;i++)
{
try
{
System.out.println("red light is glowing");
Thread.sleep(10000);
System.out.println("red light is off\n");
System.out.println("yellow light is glowing");
Thread.sleep(10000);
System.out.println("yellow light is off\n");
System.out.println("green light is glowing");
Thread.sleep(10000);
System.out.println("green light is off\n");
}
catch(Exception e)
```

```
{
System.out.println(e);
}
}
}
}
class Traffic
{
public static void main(String args[])
{
System.out.print("How many cycles : ");
int n=new Scanner(System.in).nextInt(); TrafficSignal
t=new TrafficSignal(n);
t.start();
}
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac Traffic.java

C:\Users\abhir\Desktop>java Traffic
How many cycles : 3
red light is glowing
red light is off

yellow light is glowing
yellow light is off

green light is glowing
green light is off

red light is glowing
red light is off

yellow light is glowing
yellow light is off

green light is glowing
green light is off

red light is glowing
red light is off

yellow light is glowing
yellow light is off

green light is glowing
green light is off


C:\Users\abhir\Desktop>
```

4. Write a Program using Threads for the following case study: Movie Theatre To watch a movie the following process is to be followed, at first get the ticket then show the ticket. Assume that N persons are trying to enter the Theatre hall all at once, display their sequence of entry into theater. Note: The person should enter only after getting a ticket and showing it to the boy.

PROGRAM:

```
import java.io.*;
import java.util.*;
class Theatre extends Thread{
	public void run(){
		Scanner sc= new Scanner(System.in);
		Random r = new Random();
		System.out.print("Enter the no. of persons : ");
		int n= sc.nextInt();// no. of persons
		int a[][]=new int[n][2];
		System.out.println("Enter 1 to issue ticket, 0 if not");
		for(int i=0;i<n;i++){
			System.out.printf("Enter status for person %d : ",i+1);
			a[i][0]=sc.nextInt();// person bought the ticket set as one
			if(a[i][0]==1)
				System.out.println("Ticket "+ (i+1) +" issued");
			else
				System.out.println("Ticket "+ (i+1) +" not issued");
		}
		try{
			Thread.sleep(5000);
		}
		catch(Exception e){}
		System.out.print("Enter 1 if ticket showed, 0 if not showed");
		System.out.println("\n");
		for(int i=0;i<n;i++){
			System.out.printf("Enter status for person %d : ",i+1);
			a[i][1]=sc.nextInt();
			if(a[i][1]==1)
				System.out.println("Ticket "+ (i+1) +" showed");
			else
				System.out.println("Ticket "+ (i+1) +" not showed");
		}
		try{
			Thread.sleep(5000);
		}
		catch(Exception e){}
		System.out.println("\n");
		for(int i=0;i<n;i++){
			int j=i;
			if(a[j][0]==1 && a[j][1]==1)
				System.out.println("Ticket No. "+(j+1)+" holder is entered");
		}
	}
}
class Tickets{
	public static void main(String arg[]){
```

```
        Theatre t =new Theatre();
        t.start();
    }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac Tickets.java

C:\Users\abhir\Desktop>java Tickets
Enter the no. of persons : 3
Enter 1 to issue ticket, 0 if not
Enter status for person 1 : 1
Ticket 1 issued
Enter status for person 2 : 0
Ticket 2 not issued
Enter status for person 3 : 1
Ticket 3 issued
Enter 1 if ticket showed, 0 if not showed

Enter status for person 1 : 1
Ticket 1 showed
Enter status for person 2 : 1
Ticket 2 showed
Enter status for person 3 : 0
Ticket 3 not showed


Ticket No. 1 holder is entered

C:\Users\abhir\Desktop>
```

5. Write a Program using Threads for the following case study: Train Reservation system To reserve a berth the following process need to be followed, at first check the number of available berths with the requested berths, if the number of requested berths are less than or equal to availabe berths then allot berth and print ticket or else display no berths are available. Assume that N persons are trying to reserve the berth, display their sequence of reservation status along with the number of available berths. Note : The person can print ticket only if berth is confirmed.

PROGRAM:

```
import java.util.*;

class Berths
{
    static int avlBerths=10;
    static{System.out.println("Available berths are "+avlBerths);}//executed when the
class is loaded

    static synchronized void berthAllotment(int reqBerths)
    {
        System.out.println(Thread.currentThread().getName());
        if(reqBerths<=avlBerths)
        {
            avlBerths=avlBerths-reqBerths;
            System.out.println("tickect confirmed.");
            System.out.println("tickect printed.");
            System.out.println("Available berths are : "+avlBerths);
        }
```

```
                else
                        System.out.println("Soory, No berths are available");
        }
}

class Person extends Thread
{
        int reqBerths;
        Person(int reqBerths,String personName)//here we receive a person name and set it as a thread name// super(personName);
        {
                super(personName);
                this.reqBerths=reqBerths;
        }
        public void run()
        //note that we cann't change the signature of run(), if we do so, it doesnot mean overriding.
        {
                Berths.berthAllotment(reqBerths);
        }
}

class TrainReservation
{
        public static void main(String[] args)
        {
                Person p1=new Person(4,"Abhiram");//(in reqBerths,int personName)
                Person p2=new Person(5,"Vishnu");
                Person p3=new Person(5,"Vivek");

                p1.start();
                p2.start();
                p3.start();
        }
}
```

OUTPUT:

```
C:\Users\abhir\Desktop>javac TrainReservation.java

C:\Users\abhir\Desktop>java TrainReservation
Available berths are 10
Abhiram
tickect confirmed.
tickect printed.
Available berths are : 6
Vishnu
tickect confirmed.
tickect printed.
Available berths are : 1
Vivek
Soory, No berths are available

C:\Users\abhir\Desktop>
```

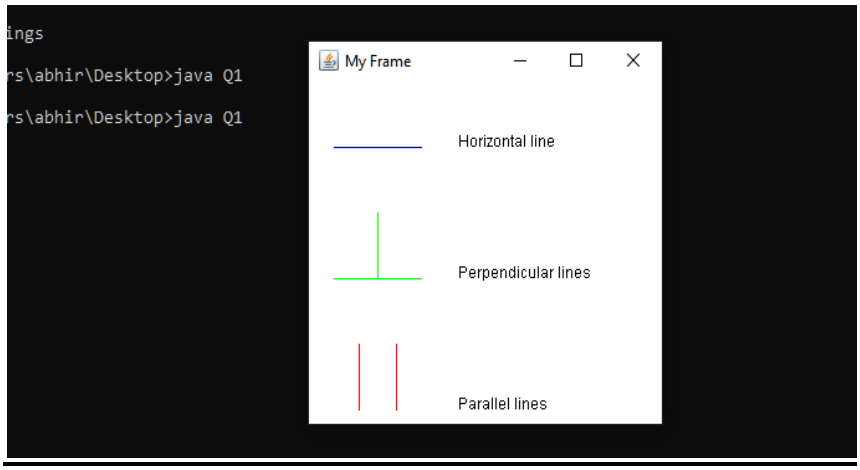
Week-XI

1. Write a program for the following
a. display a frame with title MyFrame
b. draw a horizontal line.
c. Draw one line perpendicular to other. One line parallel to other.

PROGRAM:

```
import java.awt.Graphics;
import java.awt.*;
import java.applet.Applet;
import java.awt.event.*;
public class Q1 extends Applet
{ public void paint(Graphics g){
g.drawString("Horizontal line", 120, 50);
g.drawString("Perpendicular lines", 120, 150);
g.drawString("Parallel lines", 120, 250);
g.setColor(Color.blue);
//Syntax For:- drawLine(int x1, int y1, int x2, int y2);
g.drawLine(20, 50, 90, 50);
g.setColor(Color.green);
g.drawLine(55, 100, 55, 150);g.setColor(Color.green);
g.drawLine(20, 150, 90, 150);
g.setColor(Color.red);
g.drawLine(40, 200, 40, 250);
g.setColor(Color.red);
g.drawLine(70, 200, 70, 250);
}
public static void main(String[] args) {
Frame f=new Frame();
f.setTitle("My Frame");
Applet app=new Q1();
f.add(app);
f.setSize(300,300);
f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
}}
);
} }
```

OUTPUT:



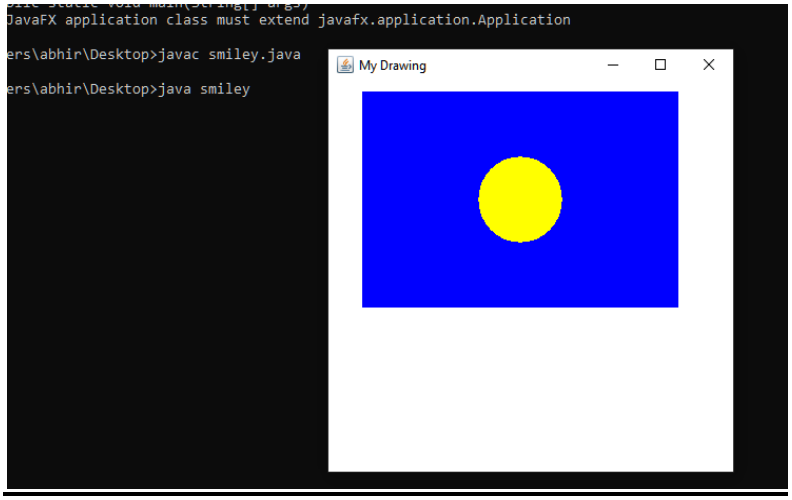
2. Create an application to display a circle within rectangle and fill different colors in the circle & rectangle

PROGRAM:

```
import java.awt.*;
import java.awt.event.*;
class Draw extends Frame
{
Draw()
{//Close the frame
this.addWindowListener(new
WindowAdapter()
{
public void
windowClosing(WindowEvent e)
{
System.exit(0);
}
});
}
//to refresh the frame contents
public void paint(Graphics g)
{
//set blue color for drawing
g.setColor(Color.blue);
//display a rectangle to contai drawing
g.fillRect(40,40,300,200);
//set yellow color
g.setColor(Color.yellow);
//circle
g.fillOval(150,100,80,80);
}
}
```

```
class smiley {
public static void main(String args[])
{
//create the frame
Draw d=new Draw();
//set the size and Title
d.setSize(400,400);
d.setTitle("My Drawing");
//Display the frame
d.setVisible(true);}
}
```

OUTPUT:



3. Write an application that displays any string. Choose color from combo box to change the color of this displayed string and choose its size & type respectively from another two combo boxes.

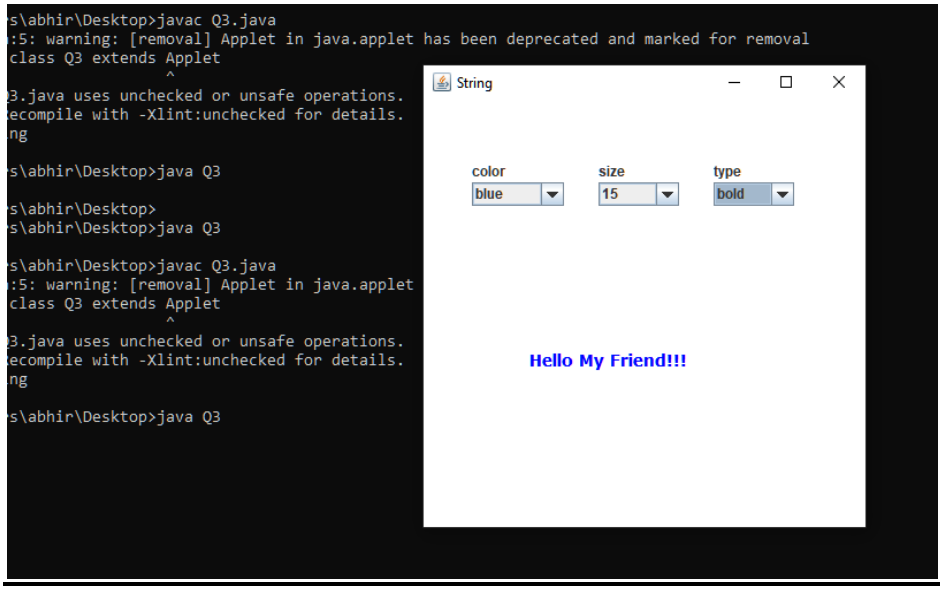
PROGRAM:

```
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Q3 extends Applet
{
public static JLabel l,l1,l2,l3;
public static JComboBox cb,cb1,cb2;
public static void main(String[] args){
Frame f = new Frame("String");
f.setSize(400, 400);
```

```
String colors[]={"select","red","blue","green"};
String sizes[]={"select","14","15","16"};
String type[]={"select","bold","italic","normal"};
l=new JLabel("Hello My Friend!!!");
l.setBounds(100,200,300,100);
l1=new JLabel("color");
l1.setBounds(50,80,80,20);
l2=new JLabel("size");
l2.setBounds(160,80,80,20);
l3=new JLabel("type");
l3.setBounds(260,80,80,20);
cb=new JComboBox(colors);
cb.setBounds(50,100,80,20);
cb1=new JComboBox(sizes);
cb1.setBounds(160,100,70,20);
cb2=new JComboBox(type);
cb2.setBounds(260,100,70,20);
f.add(cb);
f.add(cb1);
f.add(cb2);
f.add(l);
f.add(l1);
f.add(l2);
f.add(l3);cb.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
if (cb.getSelectedItem()=="red") {
l.setForeground(Color.red);
}else if (cb.getSelectedItem()=="blue") {
l.setForeground(Color.blue);
}else if(cb.getSelectedItem()=="green"){
l.setForeground(Color.green);
} } });
cb2.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
if (cb2.getSelectedItem()=="bold") {
l.setFont(new Font("Verdana", Font.BOLD, 14));
}if (cb2.getSelectedItem()=="italic") {
l.setFont(new Font("Verdana", Font.ITALIC, 15));
} if(cb2.getSelectedItem()=="normal"){
l.setFont(new Font("Verdana", Font.PLAIN, 16));
} } });
cb1.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
if (cb1.getSelectedItem()=="14") {
l.setFont(new Font("Verdana", Font.PLAIN, 14));
```

```
}else if (cb1.getSelectedItem()=="15") {
l.setFont(new Font("Verdana", Font.PLAIN, 15));
}else if(cb1.getSelectedItem()=="16"){
l.setFont(new Font("Verdana", Font.PLAIN, 16));
} } });
f.setLayout(null);
f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
} }); } }
```

OUTPUT:



5. Create a GUI with title STUDENT which has labels roll no., name, course, gender, class, address with textboxes for taking input from the user(without any functionality) and checkboxes for selecting the course, radio buttons for selecting gender with appropriate background color.

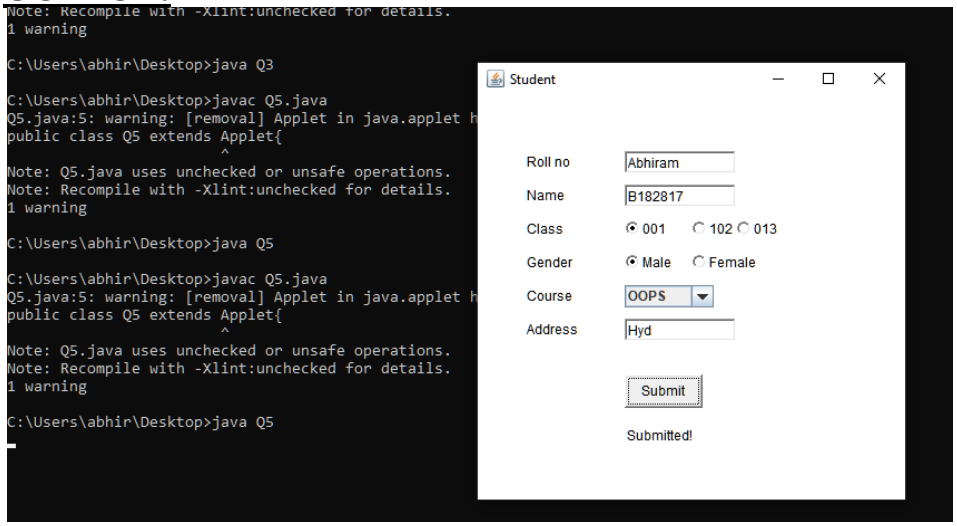
PROGRAM:

```
import java.applet.Applet;
import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
public class Q5 extends Applet{
public static Label l,l1,l2,l3,l4,l5,l6;
public static TextField t1,t2,t3,t4;
public static JComboBox cb;
public static void main(String[] args) {
```

```
Frame f=new Frame("Student");
l1=new Label("Roll no");
l1.setBounds(50,80,50,20);
l2=new Label("Name");
l2.setBounds(50,110,50,20);
l3=new Label("Class");
l3.setBounds(50,140,50,20);
l4=new Label("Gender");
l4.setBounds(50,170,50,20);
l5=new Label("Course");
l5.setBounds(50,200,50,20);
l6=new Label("Address");
l6.setBounds(50,230,50,20);
String course[]={"Ds","Daa","OOPS","c"};cb=new
JComboBox(course);
cb.setBounds(140,200,80,20);
t1=new TextField();
t1.setBounds(140,80,100,20);
t2=new TextField();
t2.setBounds(140,110,100,20);
t3=new TextField();
t3.setBounds(140,230,100,20);
CheckboxGroup cbg=new CheckboxGroup();
Checkbox box1=new Checkbox("001",false,cbg);
Checkbox box2=new Checkbox("102",false,cbg);
Checkbox box3=new Checkbox("013",false,cbg);
box1.setBounds(140,140,40,20);
box2.setBounds(200,140,40,20);
box3.setBounds(240,140,40,20);
CheckboxGroup cbg1=new CheckboxGroup();
Checkbox box4=new Checkbox("Male",false,cbg1);
Checkbox box5=new Checkbox("Female",false,cbg1);
box4.setBounds(140,170,60,20);
box5.setBounds(200,170,60,20);
Button b=new Button("Submit");
b.setBounds(140,280,70,30);
l=new Label("Submitted!");
l.setBounds(140,320,60,30);
l.setVisible(false);
b.addActionListener(new ActionListener(){
@Override
public void actionPerformed(ActionEvent e){
l.setVisible(true);
}}
);
f.add(l1);
f.add(l2);
f.add(l3);
```

```
f.add(l4);
f.add(l6);
f.add(t1);f.add(t2);
f.add(box1);
f.add(box2);
f.add(box3);
f.add(box4);
f.add(box5);
f.add(l5);
f.add(cb);
f.add(t3);
f.add(b);
f.add(l);
f.setSize(400,400);
f.setLayout(null);
f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
}}
); } }
```

OUTPUT:



6. Create a GUI application to display a calculator using grid Layout (You do not have to provide functionality).

PROGRAM:

```
import java.awt.event.*;
import javax.swing.*;
import java.awt.*;
class calculator extends JFrame implements ActionListener {
        // create a frame
        static JFrame f;
```

```
// create a textfield
static JTextField l;

// store operator and operands
String s0, s1, s2;

// default constructor
calculator()
{
        s0 = s1 = s2 = "";
}

// main function
public static void main(String args[])
{
        // create a frame
        f = new JFrame("calculator");

        try {
                // set look and feel

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
        }
        catch (Exception e) {
                System.err.println(e.getMessage());
        }

        // create a object of class
        calculator c = new calculator();

        // create a textfield
        l = new JTextField(16);

        // set the textfield to non editable
        l.setEditable(false);

        // create number buttons and some operators
        JButton b0, b1, b2, b3, b4, b5, b6, b7, b8, b9, ba, bs, bd, bm, be, beq, beq1;

        // create number buttons
        b0 = new JButton("0");
        b1 = new JButton("1");
        b2 = new JButton("2");
        b3 = new JButton("3");
        b4 = new JButton("4");
        b5 = new JButton("5");
        b6 = new JButton("6");
```

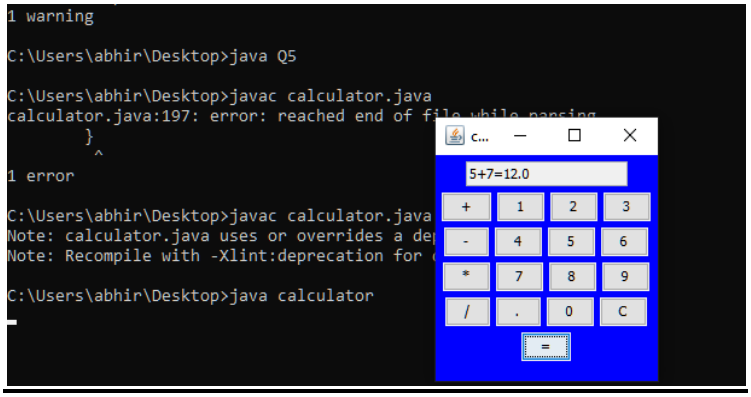
```
b7 = new JButton("7");
b8 = new JButton("8");
b9 = new JButton("9");

// equals button
beq1 = new JButton("=");

// create operator buttons
ba = new JButton("+");
bs = new JButton("-");
bd = new JButton("/");
bm = new JButton("*");
beq = new JButton("C");

// create . button
be = new JButton(".");

// create a panel
JPanel p = new JPanel();

// add action listeners
bm.addActionListener(c);
bd.addActionListener(c);
bs.addActionListener(c);
ba.addActionListener(c);
b9.addActionListener(c);
b8.addActionListener(c);
b7.addActionListener(c);
b6.addActionListener(c);
b5.addActionListener(c);
b4.addActionListener(c);
b3.addActionListener(c);
b2.addActionListener(c);
b1.addActionListener(c);
b0.addActionListener(c);
be.addActionListener(c);
beq.addActionListener(c);
beq1.addActionListener(c);

// add elements to panel
p.add(l);
p.add(ba);
p.add(b1);
p.add(b2);
p.add(b3);
p.add(bs);
p.add(b4);
p.add(b5);
```

```
        p.add(b6);
        p.add(bm);
        p.add(b7);
        p.add(b8);
        p.add(b9);
        p.add(bd);
        p.add(be);
        p.add(b0);
        p.add(beq);
        p.add(beq1);

        // set Background of panel
        p.setBackground(Color.blue);

        // add panel to frame
        f.add(p);

        f.setSize(200, 220);
        f.show();
    }
    public void actionPerformed(ActionEvent e)
    {
        String s = e.getActionCommand();

        // if the value is a number
        if ((s.charAt(0) >= '0' && s.charAt(0) <= '9') || s.charAt(0) == '.') {
            // if operand is present then add to second no
            if (!s1.equals(""))
                s2 = s2 + s;
            else
                s0 = s0 + s;

            // set the value of text
            l.setText(s0 + s1 + s2);
        }
        else if (s.charAt(0) == 'C') {
            // clear the one letter
            s0 = s1 = s2 = "";

            // set the value of text
            l.setText(s0 + s1 + s2);
        }
        else if (s.charAt(0) == '=') {

            double te;

            // store the value in 1st
            if (s1.equals("+"))
```

```
                te = (Double.parseDouble(s0) + Double.parseDouble(s2));
            else if (s1.equals("-"))
                te = (Double.parseDouble(s0) - Double.parseDouble(s2));
            else if (s1.equals("/"))
                te = (Double.parseDouble(s0) / Double.parseDouble(s2));
            else
                te = (Double.parseDouble(s0) * Double.parseDouble(s2));

            // set the value of text
            l.setText(s0 + s1 + s2 + "=" + te);

            // convert it to string
            s0 = Double.toString(te);

            s1 = s2 = "";
        }
        else {
            // if there was no operand
            if (s1.equals("") || s2.equals(""))
                s1 = s;
            // else evaluate
            else {
                double te;

                // store the value in 1st
                if (s1.equals("+"))
                    te = (Double.parseDouble(s0) + Double.parseDouble(s2));
                else if (s1.equals("-"))
                    te = (Double.parseDouble(s0) - Double.parseDouble(s2));
                else if (s1.equals("/"))
                    te = (Double.parseDouble(s0) / Double.parseDouble(s2));
                else
                    te = (Double.parseDouble(s0) * Double.parseDouble(s2));

                // convert it to string
                s0 = Double.toString(te);

                // place the operator
                s1 = s;

                // make the operand blank
                s2 = "";
            }

            // set the value of text
            l.setText(s0 + s1 + s2);
        }
    }
```

}

OUTPUT:



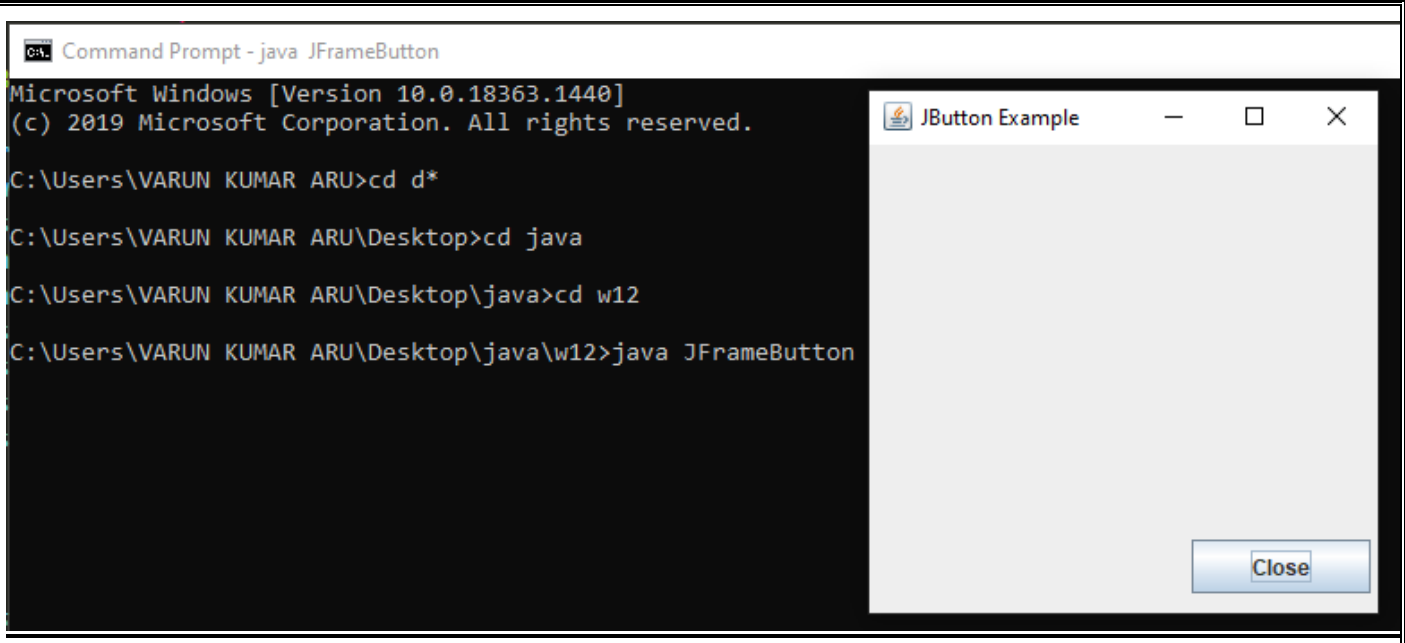
Week-XII

1. Write a program to create a frame by creating an object to Jframe class and include close button to terminate the application of the frame.

PROGRAM:

```
import java.awt.event.*;
import javax.swing.*;
class JFrameButton implements ActionListener{
        JFrameButton(){
                JFrame f=new JFrame("JButton Example");
                JButton b=new JButton("Close");
                b.setBounds(180,220,100,30);
                b.addActionListener(this);
                f.add(b);
      f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                f.setSize(300,300);
                f.setLayout(null);
                f.setVisible(true);
        }
        public void actionPerformed(ActionEvent ae){
                System.exit(0);
        }
        public static void main(String[] args){
                new JFrameButton();
        }
}
```

OUTPUT:



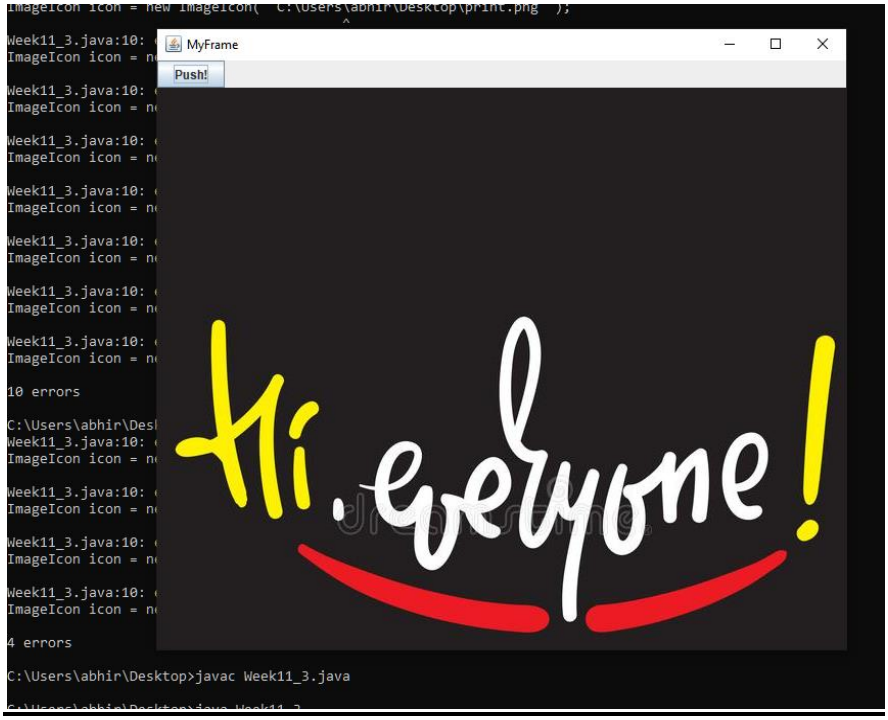
2. Write program for the following.
a. Display text in the frame by overriding PaintComponent() method of Jpanel class.
b. Display some text in the frame with the help of a Label.

PROGRAM:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class MyFrame extends JFrame {
MyFrame() {
JButton button = new JButton("Push!");
button.setBounds(100, 250, 50, 50);
button.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent ae) {
ImageIcon icon = new ImageIcon("C:/Users/abhir/Desktop/print.png");
JLabel label = new JLabel(icon);
add(label);
pack();
}
});
add(button);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setTitle("MyFrame");
setLayout(new BoxLayout(getContentPane(), BoxLayout.Y_AXIS));
setBounds(50, 200, 430, 330);
setVisible(true);
}
}
class Week11_3 {
public static void main(String[] args) {
new MyFrame();
```

```
}
}
```

OUTPUT:



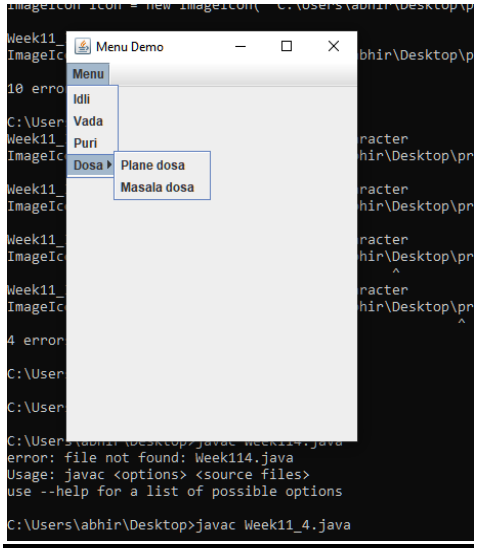
3. Write a program to create a push button , when the button is clicked an image is displayed in the frame.

PROGRAM:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class MyFrame extends JFrame {
MyFrame() {
JMenuBar menuBar = new JMenuBar();
JMenu menu = new JMenu("Menu");
menu.add(new JMenuItem("Idli"));
menu.add(new JMenuItem("Vada"));
menu.add(new JMenuItem("Puri"));
JMenu subMenu = new JMenu("Dosa");
subMenu.add(new JMenuItem("Plane dosa"));
subMenu.add(new JMenuItem("Masala dosa"));
menu.add(subMenu);
menuBar.add(menu);
setJMenuBar(menuBar);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setTitle("Menu Demo");
setLayout(null);
```

```
setBounds(50, 200, 300, 400);
setVisible(true);
}
}
class Week11_4 {
public static void main(String[] args) {
new MyFrame();
}
}
```

OUTPUT:



4. Write a program to create a menu with several menu items.

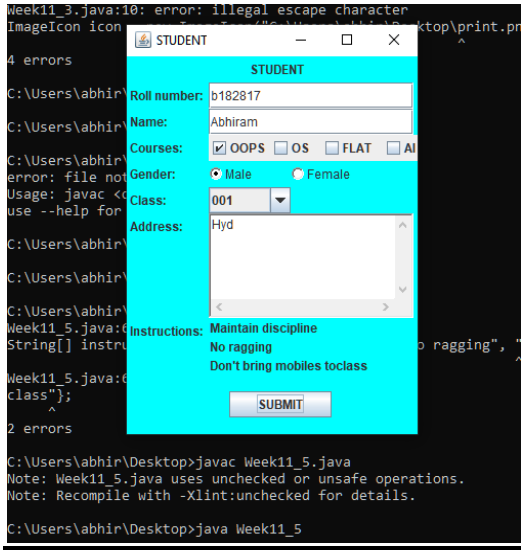
PROGRAM:

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class StudentForm extends JFrame {
StudentForm() {
JLabel stdLabel = new JLabel("STUDENT");
stdLabel.setBounds(120, 0, 100, 25);
add(stdLabel);
JLabel rnLabel = new JLabel(" Roll number: ");
rnLabel.setBounds(0, 25, 80, 25);
add(rnLabel);
JTextField rnTF = new JTextField(20);
rnTF.setBounds(80, 25, 200, 25);
add(rnTF);
JLabel nameLabel = new JLabel(" Name: ");
nameLabel.setBounds(0, 50, 80, 25);
```

```
add(nameLabel);
JTextField nameTF = new JTextField(20);
nameTF.setBounds(80, 50, 200, 25);
add(nameTF);
JLabel courseLabel = new JLabel(" Courses: ");
courseLabel.setBounds(0, 75, 80, 25);
add(courseLabel);
JCheckBox courseCB1 = new JCheckBox("OOPS");
courseCB1.setBounds(80, 75, 60, 25);
add(courseCB1);
JCheckBox courseCB2 = new JCheckBox("OS");
courseCB2.setBounds(140, 75, 50, 25);
add(courseCB2);
JCheckBox courseCB3 = new JCheckBox("FLAT");
courseCB3.setBounds(190, 75, 60, 25);
add(courseCB3);
JCheckBox courseCB4 = new JCheckBox("AI");
courseCB4.setBounds(250, 75, 60, 25);
add(courseCB4);
JLabel genderLabel = new JLabel(" Gender: ");
genderLabel.setBounds(0, 100, 80, 25);
add(genderLabel);
CheckboxGroup cbg = new CheckboxGroup();
Checkbox genderCB1 = new Checkbox("Male", cbg, true);
genderCB1.setBounds(80, 100, 60, 25);
add(genderCB1);
Checkbox genderCB2 = new Checkbox("Female", cbg, false);
genderCB2.setBounds(160, 100, 100, 25);
add(genderCB2);
JLabel classLabel = new JLabel(" Class: ");
classLabel.setBounds(0, 125, 80, 25);
add(classLabel);
String[] classes = {"001", "206", "207"};
JComboBox classComboBox = new JComboBox(classes);
classComboBox.setBounds(80, 125, 80, 25);
add(classComboBox);
JLabel addrLabel = new JLabel(" Address: ");
addrLabel.setBounds(0, 150, 80, 25);
add(addrLabel);
TextArea addrTA = new TextArea();
addrTA.setBounds(80, 150, 200, 100);
add(addrTA);
JLabel iLabel = new JLabel(" Instructions: ");
iLabel.setBounds(0, 250, 80, 25);
add(iLabel);
String[] instructions = {"Maintain discipline", "No ragging", "Don't bring mobiles toclass"};
JList iList = new JList(instructions);
iList.setBounds(80, 250, 200, 60);
```

```
iList.setBackground(Color.cyan);
add(iList);
JButton submitButton = new JButton("SUBMIT");
submitButton.setBounds(100, 320, 100, 25);
add(submitButton);
setTitle("STUDENT");
getContentPane().setBackground(Color.cyan);
setLayout(null);
setSize(300, 400);
setDefaultCloseOperation(EXIT_ON_CLOSE);
setVisible(true);
}
}
class Week11_5 {
public static void main(String[] args) {
new StudentForm();
}
}
```

OUTPUT:



Week-XIII

1. Write a program to insert data into Student Table.

PROGRAM:

```
import java.sql.*;
public class w13_1{
   public static void main (String []  args){
      try{
         String url="jdbc:mysql://localhost:3306/student";
         String user="root";
         String pswd="Abhi@8987";
         //String sql1="create table student(name varchar(255),id int,branch varchar(255))";
         String sql2="insert into student values ('abhi',182817,'ece')";
         Class.forName("com.mysql.cj.jdbc.Driver");
         Connection con=DriverManager.getConnection(url,user,pswd);
         System.out.println("Connected");
         //PreparedStatement p=con.prepareStatement(sql1);
         PreparedStatement p1=con.prepareStatement(sql2);
         System.out.println("prepared Statement");
         //p.execute();
         p1.execute();
         System.out.print(" data inserted ");
         con.close();
         System.out.println("Connection closed");
      }
      catch(Exception e){;
         System.out.println(e);
      }
   }
}
```

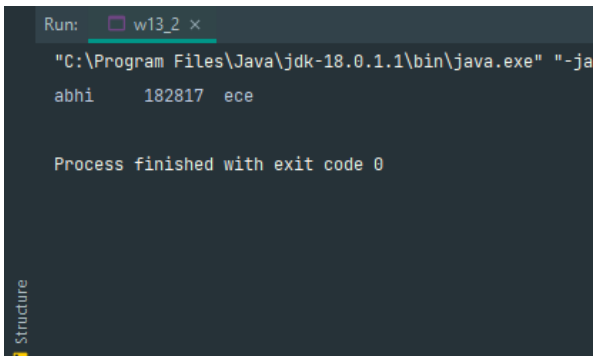
OUTPUT:

2. Write a program to retrieve the data from the table Student.

PROGRAM:

```
package com.company;

import java.sql.*;
public class w13_2{
   public static void main(String [] args){
      try{
         String url="jdbc:mysql://localhost:3306/student";
         String user="root";
         String pswd="Abhi@8987";
         String sql1="Select * from students";
         Class.forName("com.mysql.cj.jdbc.Driver");
         Connection c=DriverManager.getConnection(url,user,pswd);
         Statement s=c.createStatement();
         ResultSet r=s.executeQuery(sql1);
         while (r.next()) {
            System.out.println(r.getString(1) + "\t " + r.getInt(2) + "\t " + r.getString(3) );

         }

         c.close();
      }catch(Exception e){
         System.out.println(e);
      }

   }
}
```

OUTPUT:



3. Create a Form to insert and retrieve the data from Database as user prefer.

PROGRAM:

```
package com.company;

import java.sql.*;
import java.util.*;
public class w13_3{
   public static void main(String [] args){
      Scanner sc=new Scanner(System.in);
      try{
         String url="jdbc:mysql://localhost:3306/student";
         String user="root";
         String pswd="Abhi@8987";
         String name,branch;
         int id;

         Class.forName("com.mysql.cj.jdbc.Driver");
         Connection c=DriverManager.getConnection(url,user,pswd);
         String sql = "insert into students values(?,?,?)";
         PreparedStatement ptst=c.prepareStatement(sql);
         System.out.println("Enter name to insert");
         name=sc.next();
         System.out.println("Enter id to insert");
         id=sc.nextInt();
         System.out.println("Enter branch to insert");
         branch=sc.next();
         ptst.setString(1,name);
         ptst.setInt(2,id);
         ptst.setString(3,branch);
         ptst.executeUpdate();
         System.out.println("inserted");
         c.close();


      }
      catch(Exception e){
         System.out.println(e);
      }

   }
}
```

OUTPUT:


Run:
w13_3
"C:\Program Files\Java\jdk-18.0.1.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrain
Enter name to insert
shiva
Enter id to insert
189272
Enter branch to insert
cse
inserted
Process finished with exit code 0
Structure