1) print "Hello world"

```
class Example
{
    public static void main (String a[]
    {
        System.out.print ("Hello world");
    }
}
```

output:-
Hello world

2) Addition

```
class Example
{
    public static void main( String a[])
    {
        int x=10;
        int y=20;
        int sum=x+y;
        System.out.print(sum);
    }
}
```

output
30

Scanned with OKEN Scanner

3) Subtraction

```
class Example
{
    public static void main (String a[])
    {
        int x = 30;
        int y = 20;
        int diff = x-y;
        System.out.print(diff);
    }
}
```

output:- 10

4) Multiplication

```
Class Example
{
    public static void main (String a[
    {
        int x=5;
        int y =6;
        int prod = x+y;
        System.out.print(prod);
    }
}
```

output:-
30

5) Division

```
class example
{
    public static void main (String a[])
    {
        int x = 10, y = 5, quotient;
        quotient = x/y;
        System.out.print (quotient);
    }
}
```

Output:
2

6. Check if a number is prime

```
class Example
{
    Public Static void main(String a[])
    {
        int n=29, flag=0;
        for (int i=2; i<= n/2; i++)
        {
            if (n % i == 0)
            {
                flag = 1;
                break;
            }
        }
        if (flag == 0)
        {
            System.Out.println(n+" is a prime
        }
```

Scanned with OKEN Scanner

DATE :

```
        else
        {
            System.Out.Print ln(n+" is not prime");
        }
    }
}
```

Output

29 is prime

Fibonacci series

```
class Example
{
    public static void main(String a[])
    {
        int n=10, term1=0, term2=1
        System.out.print ln("Fibonacci series +
            + n + "terms:");
        for (int i=1; i<=n; i++)
        {
            System.out.print (term1 + ",");
            int term3 = term1 + term2;
            term1 = term2;
            term2 = term3;
        }
    }
}
```

output:

Fibonacci series till 10 terms

0, 1, 1, 2, 3, 5, 8, 13, 21, 34

Lab 2:

PAGE NO: 5
DATE: 28/12/23

1) Roots of a Quadratic equation

```
class Example
{
    Public static void main(String arg[])
    {
        double a=1, b=-7, c=10;
        double root1, root2;
        double determinat = b*b - 4*a*c;
        if(determinant>0)
        {
        root1= (-b+ math.sqrt(d
```

1) Write a program to overload the method print that prints sum of n natural numbers when one variable is passed, & prints the prime no in a given range when 2 parameters passed.

```
class Overload{
    void print(int n){
        int sum=0;
        for(int i=1; i<=n; i++){
            sum+=i;
        }
        System.out.println("sum of"+n+" natu
        -ral numbers is"+sum);
    }
```

Scanned with OKEN Scanner

```
void print (int m, int n) {
        System.out.println("prime numbers
            int the range are");
        for (int i=m; i<=m; i++){
            int flag=0;
            for (int j=2; j<= i/2; j++){
                if (i%j==0){
                    flag=1;
                    break;
                }
            }
            if (flag==0) {
                System.Out.print(i);
            }
        }
    }
}
class OverloadDemo{
    public static void main (String[] args){
        Overload o= new Overload();
        o.print(5);
        o.print(7,13);
    }
}
```

2) Write a java program to create a class Grocer that has the variables C_name & C_phone create a method to accept 3 parameter to specify quantity of dal, quantity of Pulses & quantity of sugar. are metho to return the total price, Display the name, Ph.no & total bill of 3 customers.

```
class Grocery{
    String c_name;
    String c_ph;
    double total;

    Grocery(String c_name, String c_ph){
        this.c_name=c_name;
        this.c_ph = c_ph;
    }
    void calc(double q_dal, double q_pulses,
            double q_sugar){
        total= q_dal*100+q_pulses*80+
            q_sugar*50;
    }
    void display()
    {
        System.out.print("Name "+"phone no"+
            " "+"total");
        System.out.println(cname+" "+c_ph
            " "+total);
        System.out.Println();
    }
}

class GDemo{
    public static void main(String[] args){
        Grocery g1= new Grocery("Rama", "998998
            5232");
        Grocery g2= new Grocery("Krishna", "9979
            85233");
        Grocery g3 = new Grocery("Kalki", "895
            49952");
```

Scanned with OKEN Scanner

DATE :

```
        q1.calc(2,2,1);
        q1.display();
        q2.calc(3,5,2);
        q2.display();
        q3.calc(1,1,0.5);
        q3.display();
    }
}
```

Write a java program to calculate roots of a quadratic eqn.

```
import java.util.Scanner;
class Quad{
    int a,b,c;
    double root1, root2, d;
    Scanner s = new Scanner(System.in);

    void input(){
        System.out.println("Quadratic eqn is
          in the form: ax^2+bx+c");
        System.out.print("Enter a:");
        a=s.nextInt();
        System.out.print("Enter b:");
        b=s.nextInt();
        System.out.print("Enter c:");
        c=s.nextInt();
    }
    void discriminant(){
        d=(b*b)-(4*a*c);
    }
    void calculateRoots(){
        if(d>0){
```

PAGE NO :
DATE :

```
System.out.println("Roots are real & distinct");
root1 = (-b + Math.sqrt(d))/(2*a);
root2 = (-b - Math.sqrt(d))/(2*a);
System.out.println("First root is:" + root1);
System.out.println("Second root is:" + root2);
}
else if(d==0){
System.out.println("Roots are real & equal");
root1 = (-b + Math.sqrt(d))/(2*a);
System.out.println("Root:" + root1)
}
else
{
System.out.println("No real solns. Roots are imaginary");
double real = -b/(2*a);
double imaginary = Math.sqrt(-d)/(2*a)
System.out.println("Equation has two complex roots" + real + "+" + imaginary + "i & " + real + "-" + imaginary + "i");
}
}
}
class Main{
public static void main(String[] args){
Quad q = new Quad();
q.input();
q.discriminant();
q.calculateRoots();
}
}
```

DATE :

output 1:

Sum of 6 natural numbers is 21
prime numbers in the range are 7, 11, 13

output 2:

Name	Phone number	Total
Rama	8995922356	450.0

Name	Phone number	Total
Krishna	7353495526	700.0

Name	Phone number	Total
Kalki	9535491623	150.0

output 3:

Quadratic eqn is in the form of ax^2+bx+c
Enter a:
5
Enter b: 12
Enter c = 6
No real solutions. Roots are imaginary
The eqn has 2 complex roots:
0.0 + 1.255726i & 0.0 - 1.255726i

8/12/24

Scanned with OKEN Scanner

1) Write a java program to create a class student with members USN, name, marks (6 subjects). Include a method to calculate accept student details & marks. Also include a method to calculate the percentage & display appropriate details

```
import java.util.Scanner;

class scanner{
    String USN;
    String Name;
    double[] marks = new double[6];

void input details(){
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter USN: ");
    USN = Scanner.nextline();
    System.out.print("Enter name: ");
    name = scanner.nextline();
    System.out.printf("Enter marks for 6 subjects:");
    for (int i=0; i<6; i++){
        System.out.print("Subject" + (i+1) + ":");
        marks[i] = Scanner.nextDouble();
    }
}
double calculatePercentage(){
    double totalmarks = 0;
    for (double mark : marks){
        totalmarks += mark;
    }
    return (totalmarks / 6);
}
```

```
void displayDetails() {
    System.out.println("\n Student Details: ");
    System.out.println("USN: " + USN);
    System.out.println("Name: " + name);
    System.out.println("Percentage: " + calculatePer
        centage() + "%");
    }
}
class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of stude
        int numberOfStudents = scanner.nextInt();
        Student[] students = new Student[number
                                of students];
        for(int i=0; i<numberOfstudents; i++) {
            System.out.println("\n Enter details for
                    Student " + (i+1) + ":");
            students[i] = new Student();
            students[i].inputDetails();
        }

        for (Student student : students) {
            student.displayDetails();
        }
    }
}
```

Output:

Enter number of students: 1

Enter details for Student 1:
Enter USN: 1BM22CS308
Enter Name: Tejas
Enter marks for 6 Subjects:
Subject 1 : 95
Subject 2 : 92
Subject 3 : 90
Subject 4 : 99
Subject 5 : 98
Subject 6 : 89

Student details:
USN : 1BM22CS293
Name : Tejas
Percentage : 93.833

2) Create a class Book that contains four members: Name, Author, Price & Num pages. Include a constructor to set the values for the members. Include methods to set & get details of the Objects. Include a to String() method that could display the complete details of the book. Develop a java program to create n book Objects.

Scanned with OKEN Scanner

```
import java.util.scanner;
class Books
{
    String name;
    String Author;
    Int price;
    int numpages;
    Books(){ }
    Books(String name, string author, int price,
                int numpages)
    {
        this.name = name;
        this.author = author;
        this.price = price;
        this.numpages = numpages;
    }
    public String toString()
    {
        String name, author, price, numpages;
        name = "Book name:" + this.name + "\n"
        author = "Author name:" + this.author +
        price = "price" + this.price + "\n";
        numpages = "number of pages:" + this.
                numpages + "\n";
        return name + author + price + numpages
    }
}

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

```
int n;
String name;
String author;
int price;
int numpages;
System.out.println("Enter no of books");
n = s.nextInt();
Books b[];
b= new Book[n];

for (int i=0; i<n; i++)
{
    System.out.println("Book"+(i+1)+":");
    System.out.print("Enter name of book:");
    name = s.next();
    System.out.print("Enter author");
    author = s.next();
    System.out.print("Enter no of pages");
    numpages = s.nextInt()
    b[i] = new Book(name, author, price,
                    numpages);
}

for(int i=0; i<n; i++)
    System.out.println("Book"+(i+1)+":\n"+
                    b[i]);
}
}
```

Output :
Enter the number of books : 1

Book 1 :
Enter the name of Book : Jungle book
Enter the author of Book : Rudyard - kipling
Enter the price of the Book : 1000
Enter the number of pages of Book : 500

Book 1 :
Book name : Jungle Book
Author : Rudyard - kipling
price : 1000
Number of pages : 500

12/1/24

Scanned with OKEN Scanner

1) Develop a java program to create an abstract class named Shape that contain 2 integery & an empty method named Print Area(). provide 3 classes named Rectangle, triangle & circle such that each one of the classes extends to class shape. Each one of the classes contain method print area() that prints area of given shape

```
abstract class Shape{
        int side1;
        int side2;
        Shape(int a, int b){
            side1 = a;
            side2 = b;
        }
        abstract void printArea();
}
class Rectangle extends Shape{
        Rectangle (int length, int width){
            Super (length, width)
        }
        void printArea(){
            int area = side1 * side2
            System.out.println("Area of rectangle is "+
                                area);
        }
}
class Triangle extends shape{
        Triangle (int base, int height){
            Super ( base, height)
        }
        void printArea(){
            double area = (1/2) * side1 * side2
            System.out.println ("Area of rectangle
                is " + area);
        }
```

```
class Circle extends Shape {
        Circle (int radius, 0) {
            Super( radius, 0);
        }
        void printArea() {
        double Area = math.pi * Side1 * Side1;
        System.out.println("Area of circle is"+
            area);
}   }
}

~~class Main{~~
class Main {
    public static void main(String a[]) {
        Rectangle rectangle new Rectangle (2,3);
        rectangle.printArea()}
        Triangle triangle= new Triangle (2,3);
        triangle.printArea();
        Circle circle = new Circle (2,0);
        circle.printArea();
    }
}
```

output

Area of rectangle is 6.0
Area of Triangle is 3.0
Area of circle is 12.56

19/1/24

Scanned with OKEN Scanner

PAGE NO :
DATE :

2) Develop a java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest & withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holder should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed

Create a class Account that stores customer name, account no & type of account. From this derive the classes cur-acct & sav-acct to make them more specific to their requirement. Include the necessary methods in order to achieve the following tasks.

Sol:

```
import java.util.Scanner;
class Account
{
    String customerName;
    long AccountNumber;
    String accounttype;
    double balance
    public Account (String custname, long
      acno, String acc type, double balan
    {
        customerName = custname;
        accountNumber = acc no;
```

PAGE NO :
DATE :

```
        accountype = acctype;
        balance = balance;
    }
    Public void displayBalance()
    {
        System out.println("Account Numb
            + accountNumber);
        Systemout.println("customer Name"+
                customer name);
        System.out.println("Accountype"+
            account type);
        System.out.println("Balance":+
            balance);
    }
public void deposit(double amount)
{
        balance+ = amount;
        Systemout.println("Deposit of
            $"+amount+"Successfull);
        displayBalance();
    }
public void withdraw(double amou
{
        if (amount <= balance)
        {
            balance - = amount;
            system.out.println("
            withdraw $"+amount+"
            Successfull);
        else{
            system.out.println("Insu
            fficient balance
```

PAGE NO :
DATE :

```
            displayBalance();
        }
    }

class curAct extends Account
{
    private double minBalance =1000;
    private double Service charge =50;
    public curAct(String customerName,
        long accountNumber, double
        balance)
    {
        Super(customerName, Account
            number, "current", balan
            ce);
    }

public void withdraw(double amount)
{
    if (amount <= balance - minBalance)
    {
        balance -= amount
        system.out.println(" withdrawl of $"+
            amount + "Successfull");
    }
    else
    {
        System.out.println("Insufficient
            fund": Service charge of $"+
            Servicecharge+"imposed");
        balance -= service charge
        displayBalance();
    }
}
```

PAGE NO :
DATE :

```
class SavAct extends Account
{
    private double interest rate=0.05;
    public SavAct(String customerName,
        long Accountnumber double balance)
    {
        super(customerName, accountnumber, balance);
    }

    public void computeInterest()
    {
        double interest = balance * interest
                rate;
        balance += interest;
        System.out.println("Interest of $"+
            interest + "credited");
        displayBalance();
    }
}

class Page2
{
    public static void main(String args[])
    {
        Scanner S= new Scanner(System.in);
        SavAct SA = new SavAct("Tejas", 954321680
                8000);
        SA.displayBalance();
        SA.deposit(1000);
        SA.computeInterest();
        SA.withdraw(2000);
        CurrAct CA = new CurrAct("Smith", 516060
                10000);
```

PAGE NO :
DATE :

```
ca.displayBalance();
ca.deposit(5000);
ca.withdraw(9000);
sc.close();
}
}
```

Scanned with OKEN Scanner

PAGE NO :
DATE :

create a package CIE which has 2 classes - Student & Internals. The class Student has members like USN, Name, Sem. The class Internals derived from Student has an Array that stores the internal marks scored in 5 courses of current semester of the student. Create another package SEE which has class External which is derived from class Student. This class has an array that stores the SEE marks scored in 5 courses of the current semester of the student. Import 2 packages in a file that declares final marks of n students in all 5 courses.

```
package CIE;

public class Student{
    String usn;
    String name;
    int sem;
    public Student (String usn, String Name,
                    int sem){
        this.usn = usn;
        this.Name = Name;
        this.sem = sem;
    }
}

public class Internals extends Student{
    int [] internalMarks;
```

PAGE NO :
DATE :

```
public Internals (String usn, String name,
        int sem, int[] internalMarks)
{
        Super (usn, name, sem);
        this.internalMarks = internal
            marks;
    }
}

package SEE;

import CIE.Student;
public class External Extends Student{
        int[] seemarks;
    public External (String usn, String
        name, int sem, int[] semarks)
    {
        Super( usn, name, sem);
        this.seemarks = seemarks;
    }
}

import CIE.Internals;
import SEE.External;

Public class Marks
    public static void main(String[] args)
    {
        int[] internalMarks1 = {45, 45, 40, 42,
                42};
        Internals student1 = new Internals("
            1BM22CS250", "Tejas", 3, internal
            marks1);
```

PAGE NO :
DATE :

```
int[] seemarks1 = {49, 50, 38, 42, 45};
ExternalStudent1 = new ExternalC
    "IBM112(S)50", "Jyoti", 3, Sumar
    );

int[] finalmarks ;
for(int i=0; i<=5; i++)
{
    finalmarks[i] = Internalmarks1[i]
        + seemarks[i] ;
}
System.out.print ln("Student - finalmar
    i)" + finalmarks);

}
```

PAGE NO :
DATE :

2) Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called 'Father' & derived class called "Son" which extends the base class. In father class, implement a constructor which takes the age & throws exception WrongAge() when the input age<0. In Son class implement a constructor that takes both father & son's age & throws an exception if son's age is >= father's age

```
import java.util.Scanner;

class WrongAge extends Exception{
    WrongAge(String message){
        super(message);
    }
}
class Father{
    int age;
    Father(int age) throws WrongAge
        if (age<0){
            throw new WrongAge("
            Invalid age: Age cannot be
            negative");
        } this.age = age;
    }
class Son extends Father{
    int sonage;
    Son(int fatherAge, int sonage) throw
```

PAGE NO :
DATE :

```
WrongAge
{   super(fatherAge);
    if(sonAge >= fatherAge){
        throw new WrongAge("Invalid
            age");
    }
    this.sonage = sonage;
}
}
public class ExceptionInheritanceDemo
    psvm(String args[])
    {
        Scanner scanner = new Scanner(
                System.in);
        try
        {
            S.op("Enter Fathers age");
            int fatherAge = scanner.nextInt
                ();
            SOP("Enter Son age");
            int sonage = scanner.nextInt();

            Son son = newSon(fatherage, sona
                ge);
            S.op("Fathers age:" +son.age);
            SOP("Son's age" + Son.age);
        }
        catch (wrong e){
            SOP("Exception caught"+e.
                getmessage());
        }
        finally{
            Scanner.close();
```

PAGE NO :
DATE :

}
}
}

o/p

Enter Fathers age : 50
Enter Sons age : 25
Father age = 50
Sons age = 25

3) Write a program which creates two threads one thread displaying "BMS college of Engineering" once every ten seconds & another displaying "CSE" once every two seconds

```
class MessageThread extends Thread{
    private String message;
    private int interval;

    Public message Thread(String
        message, int interval){
        this.message = message;
        this.interval = interval;
    }
    Public void run(){
        try{
            while(time){
                system.out.println(messag
                thread.sleep(interval
                    1000);
            }
        }
```

PAGE NO :
DATE :

```
        catch (InterruptedException e)
            e.printStackTrace();
        }
    }
}
public class DisplayMessage {
    psvm(String [] args) {
        MessageThread thread1 = new
            MessageThread("
            BMS college of Engineering
            ");
        MessageThread thread2 = new
            MessageThread("CSE");
        thread1.start();
        thread2.start();
    }
}
```

```
of
CSE
BMS college of Engineering
CSE
CSE
CSE
BMS college of Engineering
CSE
CSE
CSE
BMS college of Engineering
!
!
```

Scanned with OKEN Scanner

PAGE NO :
DATE :

Creating label, button & textfield in a frame using AWT

```
import java.awt.*;
import java.awt.event.*;

public class AWTExample ~~extends~~ extends window
    Adapter{
        Frame f;
        AWTExample(){
            f = new Frame();
            f.addWindowListener(this);
            Label l = new Label("
                Employee id:");
            Button b = new button("Submi
                -t");
            TextField t = new Textfield();

            l.SetBounds(20,80,80,30);
            t.SetBounds(20,100,80,30);
            b.SetBounds(100,100,80,30);
            f.add(b);
            f.add(l);
            f.add(t);
            f.setSize(400,300);
            f.setTitle(" Employee info");
            f.setVisible(true);
        }
    public void windowClosing(Windo
            w(Event e){
            System.exit(0);
    }
```

Scanned with OKEN Scanner

PAGE NO :
DATE :

```
public static void main(String args[])
{
    AWTExample awt_Obj = new AWT
                    Example();
}
}
```

(Create a button & add a action listener for mouse click

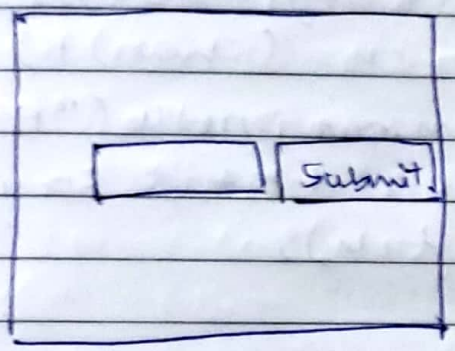
```
import java.awt.*;
import java.awt.event.*;
public class EventHandling extends WindowAdapter implements ActionListener
    frame f;
    TextField tf;
    EventHandling()
```

```
import java.io.*;

Public class ByteArray_ex
    public static void main(String args[]) throws IOException
        FileOutputStream fout1 = new FileOutputStream("Example.txt");
        FileOutputStream fout2 = new FileOutputStream("Example2.txt");
        ByteArrayOutputStream bout = new ByteArrayOutputStream();
        bout.write(65);
        bout.writeTo(fout1);
```

```
bout.writeTo(fout2);
bout.flush();
bout.close();
System.out.println("success...");
}
}
```

Output 1



Output 2

Success...

PAGE NO :
DATE :

3)
```
import java.io.*;
public class ByteArrayInput{
    public static void main (String args[])
        throws IOException{
        byte[] buf = {35,36,37,38};
        ByteArrayInputStream byt = new
            ByteArrayInputStream(buf);
        int k=0;
        while ((k=byt.read()) != -1){
            char ch = (char) k;
            System.out.println("ASII value
of character is :" + k + "; special char
acter is :" + ch);
    }
}
}
```

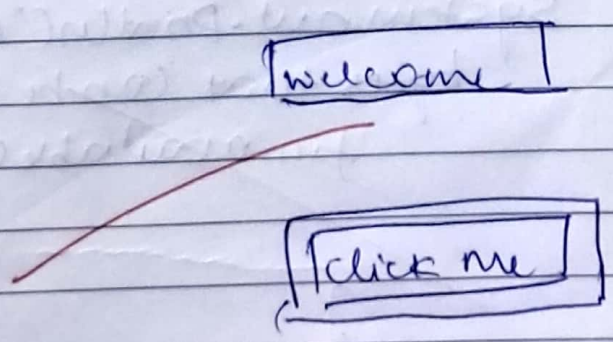
4) Creating button

```
import java.awt.*;
import java.awt.event.*;
public class Event handling Extends Window
Adapter implements ActionListener{
    Frame f;
    TextField tf;
    EventHandling(){
        f = new Frame();
        f.addWindowListener (this);
        tf = new TextField();
        tf.SetBounds (100,120,80,30);
```

PAGE NO :
DATE :

```
b.addActionListener(this);
f.add(b); f.add(tf);
f.setSize(300,300);
f.setLayout(null);
f.setVisible(true);
}
public void actionperformed(ActionEv
        ent e){
        tf.setText("welcome");
}
public void windowclosing(window
        Event e){
        System.exit(0);
}
Public Static void main(String args[]){
        new Eventhandling();
}
}
```

4) output:-



3 output

ASCII value 35 charater is #
ASCII value 36 character is $
ASCII value 37 charactor is %
ASCII value 38 charactor is &

Scanned with OKEN Scanner

PAGE NO :
DATE :

5) Public class FileEx{
    public static void main(String a[])
        throws IOException{
        FileInputStream fin = new FileInpu
        Stream("Example.txt");
        byte[] bytes = new byte[20];
        int i;
        char c;
        i = fin.read(bytes);
        System.out.println("Remaining
            bytes that can be read:"+
            fin.available());
        content = fin.read();
            System.out.print((char)con
                tent+" ");
            System.out.print(content+"")
        ~~System.~~
            System.out.println("Remai
                ning bytes that can be
                read:"+ fin.available());
        System.out.Println("Remaining
            bytes that can be read:"+
            fin.available());
    }
}

23/2/24

Scanned with OKEN Scanner