

2. Write a program to find factorial of given number n , where $n=10$.

program:

```
import java.util.Scanner;
class Factorial
{
    public static void main (String s[]) {
        Scanner sc = new Scanner(System.in);
        System.out.print("enter your number to get
factorial:");
        int a = sc.nextInt();
        int fac = 1;
        for (int i = 1; i <= a; i++) {
            fac = fac * i;
        }
        System.out.println("factorial: " + a + " ! = " + fac);
    }
}
```

output:

enter your number to get factorial: 10
factorial: 10! = 3628800
[program finished]

note: 10 is entered as input

3) write a program to convert a decimal to binary number n where $n = 124$.

```
import java.util.Scanner;  
public class DecimalToBinary  
{  
    public static void main (String s[]){  
        Scanner sc = new Scanner (System.in);  
        int n = sc  
        System.out.print("enter your number to get  
binary number:");  
        int n = sc.nextInt();  
        System.out.print("binary number of "+n+" is");  
        System.out.println(Integer.toBinaryString(n));  
    }  
}
```

output:

enter your number to get binary : 124

binary number of 124 is 1111100

note: 124 is entered as input.

4) Write a program to check if a number is prime or not.

```
import java.util.Scanner;

public class Main {
    public static void main (String s[]) {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter your number to check if
it is prime or not,");
        int i;
        int n = s.nextInt();
        for (i = 2; i < n/2; i++) {
            if (n % i == 0) {
                System.out.println(n + " is not a prime number.");
                break;
            }
        }
        if (i == n/2) {
            System.out.println(n + " is a prime number.")
        }
    }
}
```

output:

Enter your number to check if it is prime or not.

5

5 is not a prime number

- 4) write a program to find sum of any number entered as command line argument.

```
public class Sum {  
    public static void main (String s[]) {  
        int x = Integer.parseInt(s[0]);  
        int y = Integer.parseInt(s[1]);  
        int sum = x + y;  
        System.out.println("The sum of x and y is " + sum);  
    }  
}
```

- 5) Write a program that shows working of different functions of String and StringBuffer class like setCharAt(), setLength(), append(), insert(), concat() and equals().

```
class A {  
    public static void main (String[] s) {  
        StringBuffer s = new StringBuffer("Aman");  
        String sn = "Aman";  
        String n = "Garima";  
        s.setLength(4);  
        System.out.println(s.charAt(2) + "\t\t\t charAt()");  
        System.out.println(s + "\t\t\t set length()");  
        System.out.println(s.append("garima") + "\t\t\t append()");  
        System.out.println(s.insert(2, "garima") + "\t\t\t insert()");  
        System.out.println(sn.concat(n) + "\t\t\t concat()");  
        System.out.println(s.equals("Aman") + "\t\t\t equals()");  
    }  
}
```

⑤ write a program to show difference between public and private access specifier.

```
class Specifier {  
    public void show() {  
        System.out.println("we are differentiating between  
public and private access specifier.");  
    }  
    private void check() {  
        System.out.println("you are in private section of  
code.");  
    }  
    public void check2() {  
        System.out.println("you are in public section of  
code.");  
    }  
}  
  
public class Main {  
    public static void main(String s[]){  
        Specifier sc = new Specifier();  
        sc.show();  
        // sc.check(); this will throw an error. since  
it is having private access.  
        sc.check2();  
    }  
}
```

Output: we are differentiating between public and private access specifier.

you are in public section of code.

⑦ write a program to show the use of static function and to pass variable length argument.

```
public class Argument {  
    static int sum(int... arr) {  
        int ans = 0;  
        for (int z : arr) {  
            result = result + z;  
        }  
        return result;  
    }  
    public static void main (String s[]) {  
        System.out.println ("Passing 2 parameters =" +  
sum(100, 10));  
        System.out.println ("Passing 3 parameters =" +  
sum(100, 50, 10));  
        System.out.println ("Passing 0 parameters =" +  
sum());  
    }  
}
```

output: Passing 2 parameters = 110
Passing 3 parameters = 160
Passing 0 parameters = 0

7
2. write a program to demonstrate the concept of boxing and unboxing.

```
public class BoxingUnboxing {  
    public static void main(String s[]) {  
        int a = 27;  
        char c = 'a';  
        float f = 5.9f;  
  
        //autoboxing  
        Integer iob = a;  
        Character cob = c;  
        Float fob = f;  
  
        System.out.println("printing object value");  
        System.out.println("Integer value object = " + iob);  
        System.out.println("Character value object = " + cob);  
        System.out.println("float value object = " + fob);  
  
        //unboxing  
        int z = iob;  
        char r = cob;  
        float e = fob;  
  
        System.out.println("primitive value");  
        System.out.println("integer = " + z);  
        System.out.println("character = " + r);  
        System.out.println("floating value = " + e);  
    }  
}
```

Output: printing object value
Integer value object = 27
Character value object = a
float value object = 5.9
primitive value
integer = 27
character = a
floating value = 5.9

- ⑩. write a program - DivideByZero, that takes two numbers ^⑧ a and b as input compute a/b , and invokes arithmetic expression to generate a message when the denominator is 0.

```
import java.util.Scanner;
public class DivideByZero DivideByZero {
    public static void main(String s[]){
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a number =");
        int a = sc.nextInt();
        System.out.println("Enter number that divides =");
        int b = sc.nextInt();
        try {
            int n = a/b;
            System.out.println("Quotient of number = " + n);
        } catch (Exception e) {
            System.out.println(e);
        }
        System.out.println("Program is executed.");
    }
}
```

out put: Enter a number = 6
Enter number that divides = 6
Quotient of number = 1
Program is executed.

17. write a java program to demonstrate static keyword
The program consist three methods with parameter and also use the display method to show the output of program.

```
class Static {  
    static float x, y, z;  
    static float add (float a, float b) {  
        x = a + b;  
        return x;  
    }  
    static float division (float a, float b) {  
        z = a / b;  
        return z;  
    }  
    static float multiply (float a, float b, float c) {  
        y = a * b * c;  
        return y;  
    }  
    void display () {  
        // display method  
        System.out.println("addition = " + x);  
        System.out.println("division = " + z);  
        System.out.println("multiplication = " + y);  
    }  
}  
  
public class Main {  
    public static void main (String args[]) {  
        Static.add (27, 20);  
        Static.division (20, 21);  
        Static.multiply (10, 6, 8);  
        Static T = new Static();  
        T.display();  
    }  
}
```

12. write a program to create your own type exception to handle the situation specific to your application

```
import java.util.Scanner;
class Exception extends RuntimeException {
    Exception() {
        super();
    }
    Exception (String message) {
        super(message);
    }
}
class Voting {
    public static void main (String[] args) {
        Scanner s = new Scanner (System.in);
        System.out.println ("enter your age:");
        int age = s.nextInt();
        try {
            if (age < 18) {
                throw new Exception ("you are
underage");
            }
            else {
                System.out.println ("you can vote
now");
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

(11)

9. write a program that creates classes/subclasses different level protection belonging to same package or different package.

first creating a package.

```
package Casio;  
public class program {  
    int w = 10;  
    public int x = 20;  
    protected int y = 30;  
    private int z = 40;  
}
```

now this package can be imported

```
import Casio.*;
```

```
class Check extends Casio {
```

```
    int a = 100;
```

```
    public int b = 200;
```

```
    protected int c = 300;
```

```
    private int d = 400;
```

```
    public void add() {
```

```
        System.out.println(a + b + c + d);
```

```
    }
```

```
    public void show() {
```

```
        // System.out.println(w); not accessible "default"
```

```
        System.out.println(x);
```

```
        System.out.println(y);
```

```
        // System.out.println(z); private - not accessible
```

```
    }
```

```

public class Test extends Check {
    public void display() {
        System.out.println(a);
        System.out.println(b);
        System.out.println(c);
        // System.out.println(d); private access.
    }
    public static void main(String s[]) {
        Check n = new Check();
        n.add();
        System.out.println(n.a);
        System.out.println(n.b);
        System.out.println(n.c);
        // System.out.println(n.d); private
        program p1 = new program();
        // System.out.println(p1.w); default.
        System.out.println(p1.x);
        /* System.out.println(p1.y); protected
        System.out.println(p1.z); private */
    }
}

```

13. write a program to demonstrate multiple threads

```

class Multi extends Thread {
    public void run() {
        int i = 0;
        while (i < 500) {
            System.out.println("we are in first
thread.");
            i++;
        }
    }
}

class Multi1 extends Thread {
    public void run() {
        int i = 0;
        while (i < 500) {
            System.out.println("we are in second
thread.");
            i++;
        }
    }
}

public class Main {
    public static void main(String args[]) {
        new Multi().start();
        new Multi1().start();
    }
}

```

- ①. write a program to show the use of try statement that emphasize the sequence of checking for catch handler statement.

```
public class nestedDemo {  
    public static void main(String s[]) {  
        try {  
            try {  
                int [] arr = { 1, 2, 3, 4, 5, 6 };  
                System.out.println(arr[9]);  
            } catch (ArrayIndexOutOfBoundsException e2) {  
                System.out.println(e2);  
            }  
            try {  
                int i = 8/0;  
            } catch (ArithmeticException e1) {  
                System.out.println(e1);  
            }  
            catch (Exception e) {  
                System.out.println(e);  
            }  
            System.out.println("\n code successful run");  
        }  
    }  
}
```

output :- java.lang. ArrayIndexOutOfBoundsException: Index
9 out of bounds for length 6
java.lang. ArithmeticException: /by zero
code successful run
[program finished]

14. Write a program to demonstrate different mouse handling events like mouse clicked(), mouse entered(), mouse exited(), mouse pressed(), mouse released(), mouse dragged(),

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code = "MouseEvent" width = 300 height = 100>
</applet>
*/
public class MouseEvent extends Applet implements
MouseListener, MouseMotionListener {
    String msg = " ";
    int mouseX = 0, mouseY = 0;
    public void init() {
        addMouseListener(this);
        addMouseMotionListener(this);
    }
    public void mouseClicked(MouseEvent me) {
        mouseX = 0;
        mouseY = 0;
        msg = "mouse clicked";
        repaint();
    }
    public void MouseEntered(MouseEvent me) {
        mouseX = 0;
        mouseY = 10;
        msg = "mouse entered";
        repaint();
    }
}
```

```

public void mouseExited (MouseEvent me) {
    mouse X = 0;
    mouse Y = 10;
    msg = "mouse exited ";
    repaint();
}

```

```

public void mousePressed (MouseEvent me) {
    mouse X = me.getX();
    mouse Y = me.getY();
    msg = "down";
    repaint();
}

```

```

public void mouseReleased (MouseEvent me) {
    mouse X = me.getX();
    mouse Y = me.getY();
    msg = "up";
    repaint();
}

```

```

public void mouseDragged (MouseEvent me) {
    mouse X = me.getX();
    mouse Y = me.getY();
    msg = "*";
    showStatus ("Dragging mouse at " + mouse X + ", " +
    mouse Y);
    repaint();
}

```

```

public void mouseMoved (MouseEvent me) {
    showStatus ("moving mouse at " + mouse X + ", " +
    mouse Y);
}

```

```

public void paint (Graphics g) {
    g.drawString (msg, mouse X, mouse Y);
}

```

15. write a program to show different keyboard handling exception. (17)

```
import java.awt.*;  
import java.awt.event.*;  
public class KeyListenerExample extends JFrame implements  
    KeyListener {  
    Label l;  
    TextArea area;  
    KeyListenerExample() {  
        l = new Label();  
        l.setBounds(20, 50, 100, 20);  
        area = new TextArea();  
        area.setBounds(20, 80, 300, 300);  
        area.addKeyListener(this);  
        add(l);  
        add(area);  
        setSize(400, 400);  
        setLayout(null);  
        setVisible(true);  
    }  
    public void keyPressed(KeyEvent e) {  
        l.setText("Key Pressed");  
    }  
    public void keyReleased(KeyEvent e) {  
        l.setText("Key Released");  
    }  
    public void keyTyped(KeyEvent e) {  
        l.setText("Key Typed");  
    }  
    public static void main(String[] args) {  
        new KeyListenerExample();  
    }  
}
```

18) write a program to display a message in window using applet.

```
import java.awt.* ;
import java.applet.* ;
/*
<applet code = "Demo" width = 300 height = 200>
</applet>
*/
public class Demo extends Applet {
    public void paint(Graphics g) {
        g.drawString("Hello!", 20, 20);
    }
}
```