1a.

```java
public class FR
{
public static int fibonacci(int n)
{
if (n <= 1)
{
return n;
}
return fibonacci(n - 1) + fibonacci(n - 2);
}
public static void main(String[] args) {
int n = 10;
System.out.println("Fibonacci series of first 10 numbers using recursive(looping) ");
for (int i = 0; i < n; i++)
{
System.out.print(fibonacci(i) + " ");
}
}
}
```

1b.

```java
public class FR1
{
public static void fibonacci(int n)
{
int a = 0, b = 1, c;
if (n > 0)
{
System.out.print(a + " " + b + " ");
for (int i = 2; i < n; i++)
{
c = a + b;
System.out.print(c + " ");
a = b;
b = c;
}
}
}
public static void main(String[] args)
{
int n = 10;
System.out.println("Fibonacci series 10 values using non recursive function(looping)"s;
fibonacci(n);
}
}
```

2.

```java
import java.util.Scanner;
public class Mat {
public static void main(String[] args) {
Scanner S=new Scanner(System.in);
System.out.println("Enter a 2x2 Matrix");
int[][] matrixA = S.nextInt();
int[][] matrixB = S.nextInt();
int[][]result = multiplyMatrices(matrixA, matrixB);
System.out.println("Result of matrix multiplication:");
for (int i = 0; i < result.length; i++) {
for (int j = 0; j < result[i].length; j++) {
System.out.print(result[i][j] + " ");
}
System.out.println();
}
}
public static int[][] multiplyMatrices(int[][] matrixA, int[][] matrixB) {
int rowsA = matrixA.length;
int colsA = matrixA[0].length;
int rowsB = matrixB.length;
int colsB = matrixB[0].length;
int[][] result = new int[rowsA][colsB];
for (int i = 0; i < rowsA; i++) {
for (int j = 0; j < colsB; j++) {
result[i][j] = 0;
for (int k = 0; k < colsA; k++) {
result[i][j] += matrixA[i][k] * matrixB[k][j];
}
}
}
return result;
}
}
```

3.

```java
import java.util.Scanner;
class Emp
{
public static void main(String args[])
{
Scanner sc=new Scanner(System.in);
System.out.println("enter Employee name:");
String name=sc.nextLine();
System.out.println("enter Employee Id:");
int id=sc.nextInt();
System.out.println("enter employee age:");
int age=sc.nextInt();
System.out.println("enter employee salary:");
```

```java
int sal=sc.nextInt();
System.out.println("Employee DetailS");
System.out.printf("Name is :%s\n",name);
System.out.println("ID is :"+id);
System.out.println("age is :"+age);
System.out.println("Sal is :"+sal);
}
}
```

4a.

```java
package Mypackage;
public class Student
{
String name;
String usn;
int sub1;
int sub2;
int sub3;
int marks;
double per;
public Student(String name,String usn,int sub1,int sub2,int sub3)
{
this.name=name;
this.usn=usn;
this.sub1=sub1;
this.sub2=sub2;
this.sub3=sub3;
}
public void display()
{
System.out.println("Name :"+name);
System.out.println("usn :"+usn);
System.out.println("Marks of 3 subjects are: " + sub1 + " " + sub2 + " " + sub3);
int marks=sub1+sub2+sub3;
double per=marks/3;
System.out.println("marks is "+marks);
System.out.println("Percentage is "+per);
}
}
```

4b.

```java
import Mypackage.Student;
class Main
{
public static void  main(String args[])
{
Student obj=new Student("manoj","ENG23MCA018",60,70,80);
obj.display();
```

```
}
}
```

5.

```java
import java.awt.*;
import java.applet.*;
/*<applet code="FirstApplet" width=500 height=300></applet>*/
public class FirstApplet extends Applet
{
public void paint(Graphics g)
{
g.setColor(Color.blue);
Font font=new Font("Arial",Font.BOLD,16);
g.setFont(font);
g.drawString("This is My First applet",60,110);
}
}
```

6a.

```java
import java.awt.*;
import java.applet.*;

/*<applet code="Fact" width=500 height=300></applet>*/
public class Fact extends Applet
{
    static int n = 6;

    public void paint(Graphics g)
    {
      int F = 1;
      for(int i = 1; i <= n; i++)
      {
        F *= i;
      }

      g.setColor(Color.blue);
      Font font = new Font("Arial", Font.BOLD, 16);
      g.setFont(font);
      g.drawString("Factorial of " + n + " is: " + F, 60, 110);
  }
}
```

6b.

```java
import java.awt.*;
import java.applet.*;
/*<applet code="Parameter" height="300" width="500">
<param name="PGM" value="Student details using parameter passing in applets" />
```

```java
<param name="name" value="Manoj k h" />
<param name="age" value="22" />
<param name="USN" value="ENG23MCA018" />
<param name="Course" value="MCA" />
</applet> */
public class Parameter extends Applet
{
String a;
String b;
String c;
String d;
String e;
public void init()
{
a=getParameter("PGM");
b=getParameter("name");
c=getParameter("age");
d=getParameter("USN");
e=getParameter("course");
}
public void paint(Graphics g)
{
g.setColor(Color.red);
Font font = new Font("Arial", Font.BOLD, 20);
g.setFont(font);g.drawString(a, 20,20);
g.drawString("Name: " + b,20,40);
g.drawString("Age: " + c, 20, 60);
g.drawString("USN : " + d, 20, 80);
g.drawString("Course: " + e, 20, 100);


}

}
```

7.

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

public class MouseEventDemo extends JFrame implements MouseListener,
MouseMotionListener
{
JLabel label;
public MouseEventDemo()
{
setTitle("Mouse Event Demo");
setSize(400, 400);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
label = new JLabel();
```

```java
label.setBounds(20, 50, 200, 20);
addMouseListener(this);
addMouseMotionListener(this);
add(label);
setLayout(null);
setVisible(true);
}
public void mouseClicked(MouseEvent e)
{
label.setText("Mouse Clicked at: (" + e.getX() + ", " + e.getY() + ")");
}
public void mousePressed(MouseEvent e)
{
label.setText("Mouse Pressed at: (" + e.getX() + ", " + e.getY() + ")");
}
public void mouseReleased(MouseEvent e)
{
label.setText("Mouse Released at: (" + e.getX() + ", " + e.getY() + ")");
}
public void mouseEntered(MouseEvent e)
{
label.setText("Mouse Entered the Frame");
}
public void mouseExited(MouseEvent e)
{
label.setText("Mouse Exited the Frame");
}
public void mouseDragged(MouseEvent e)
{
label.setText("Mouse Dragged at: (" + e.getX() + ", " + e.getY() + ")");
}
public void mouseMoved(MouseEvent e)
{
label.setText("Mouse Moved at: (" + e.getX() + ", " + e.getY() + ")");
}
public static void main(String[] args)
{
new MouseEventDemo();
    }
}
```

8.

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

public class KeyEventDemo extends JFrame implements KeyListener {

    JLabel label;

    public SimpleKeyEventDemo() {
```

```java
        setTitle("Simple Key Event Demo");
        setSize(300, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        label = new JLabel("Type something", SwingConstants.CENTER);
        label.setBounds(50, 100, 200, 50);

        addKeyListener(this);


        add(label);
        setLayout(null);
        setVisible(true);
        setFocusable(true); // Ensures the frame can capture key events
    }

    public void keyPressed(KeyEvent e) {
        label.setText("Key Pressed: " + e.getKeyChar());
    }

    public void keyReleased(KeyEvent e) {
        label.setText("Key Released: " + e.getKeyChar());
    }

    public void keyTyped(KeyEvent e) {
        label.setText("Key Typed: " + e.getKeyChar());
    }

    public static void main(String[] args) {
        new SimpleKeyEventDemo();
    }
}
```

9.

```java
import java.sql.*;
import javax.sql.*;


public class Pmg_connect
{
  public static void main(String[] args)
        {

        String url = "jdbc:mysql://localhost:3306/Pmg_connect?useSSL=false";
        String username = "root";
        String password = "Manojkh#2002";


        try
                {
```

```java
        Connection con = DriverManager.getConnection(url, username, password);
        System.out.println("Connected to MySQL database successfully!");

                        con.close();
    }
                catch (SQLException e)
                {
     e.printStackTrace();
    }

  }
}
```

10.

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class JDBC {
  public static void main(String[] args) {
    String jdbcURL = "jdbc:mysql://localhost:3306/Pmg_connect";
    String username = "root";
    String password = "Manojkh#2002";
    try (
Connection connection = DriverManager.getConnection(jdbcURL, username, password);
      Statement statement = connection.createStatement()) {

      statement.execute("CREATE TABLE IF NOT EXISTS employees (ID INT AUTO_INCREMENT,
Name VARCHAR(100), PRIMARY KEY (ID))");


      statement.execute("INSERT INTO employees (Name) VALUES ('John Doe'), ('Jane Doe')");


      ResultSet resultSet = statement.executeQuery("SELECT * FROM Users");
      while (resultSet.next()) {
        System.out.println("ID: " + resultSet.getInt("ID") + ", Name: " +
resultSet.getString("Name"));
      }

    } catch (SQLException e) {
      e.printStackTrace();
    }
  }
}
```