Sr.No	Date	Program	Remark
1	16-12-2022	Write a program in Java to print first 20 elements of Fibonacci series	
2	16-12-2022	Write a program in Java to find the minimum and maximum of the 10 elements present in an array. Write member function to find the minimum and maximum numbers.	
3	16-12-2022	Write a program in Java to assign elements to two 2-dimensional arrays and write a member function to add and subtract the matrices.	
4	16-12-2022	Write a program in Java to create a class of objects cubes with parameterized and non-parameterized constructor. Design a program to print the volume and outer area of the cubes of side 10 meters.	
5	29-12-2022	Write a program in Java to demonstrate method overloading by writing a Java program to print the volume and total outer area of the cubes of two cylinders whose radius and height is known.	
6	29-12-2022	Write a program in Java to define a class called ice cream with the following attributes.	
7	29-12-2022	Write a program in Java to demonstrate the use of “Static”, by counting the number of objects created.	
8	06-01-2023	Write a program in Java to demonstrate the use of “this” as	
9	06-01-2023	Write a program in Java to create an abstract class pen with methods write () and refill () as abstract methods. Create a concrete class fountain pen with additional method change Nib ()	
10	13-01-2023	Write a program in Java to override the method “area” in three different classes called as Rectangle, Square and Circle.	
11	13-01-2023	Write a program in Java to create a package “MyPackage” with class “Student”. Display the roll no and name of the student.	
12	20-01-2023	Write a program in Java to demonstrate the ArithmeticException	
13	20-01-2023	Write a program in Java to display the length of two different Strings. Compare both the Strings and display the result.	
14	27-01-2023	Write a program in Java to demonstrate creation of two threads. Make each one of them sleep for 1000 miliseconds. Display the thread name. Display which thread is exiting.	
15	03-02-2023	Write a program in Java to get “Employee number”, “Name” and “Salary” from the user and display.	
16	10-02-2023	Write a program in Java to design a login form containing Login id and Password with “Submit” and “Reset” buttons using GridLayout.	
17	10-02-2023	Write a program to implement N queen Problem using backtracking approach.	

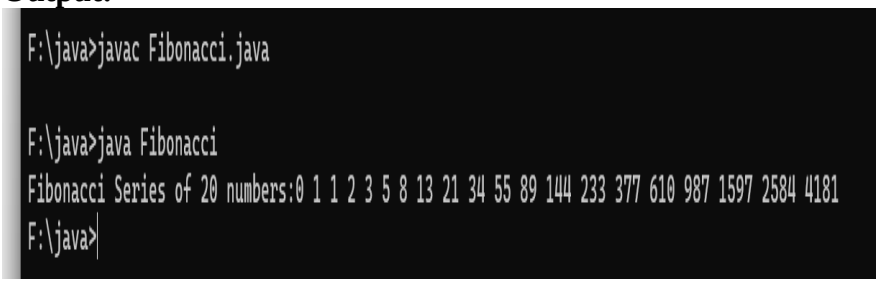
18	17-02-2023	Write a Java program to create “Student_Master” table in MySQL having fields roll no, name, marks for 3 subjects. a. Insert minimum 5 records. b. Calculate Total Marks, Percentage and Grade for each student. c. Display the marksheet for each student in proper format.	
19	17-02-2023	Write a Java Servlet which accepts the username & greets him/her with appropriate message based on the time of day. For example, if it is in the morning, display “Good Morning!”. Also display current date & time.	
20	17-02-2023	Write JSP code for the Sign UP (Registration) & Sign In (Login) Module.	

Q.1) Write a program in Java to print first 20 elements of Fibonacci series

Code:

```
public class Fibonacci {
   public static void main(String[] args) {
      int n = 20, a = 0, b = 1, c;
      System.out.print("Fibonacci Series of "+n+" numbers:");
      for (int i = 1; i <= n; i++) {
         System.out.print(a + " ");
         c = a + b;
         a = b;
         b = c;
      }
   }
}
```

Output:



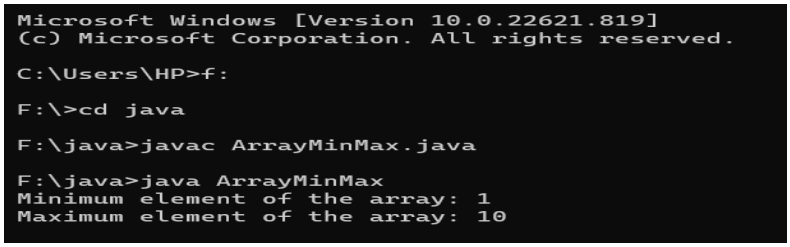
Q.2) Write a program in Java to find the minimum and maximum of the 10 elements present in an array. Write member function to find the minimum and maximum numbers.

Code:

```
public class ArrayMinMax {
   public static void main(String[] args) {
      int[] arr = {4, 5, 1, 8, 2, 10, 6, 7, 9, 3};
      int min = findMin(arr);
      int max = findMax(arr);
      System.out.println("Minimum element of the array: "+min);
      System.out.println("Maximum element of the array: "+max);
   }

   public static int findMin(int[] arr) {
      int min = arr[0];
      for (int i = 1; i < arr.length; i++) {
         if (arr[i] < min) {
```

```
            min = arr[i];
        }
    }
    return min;
}

public static int findMax(int[] arr) {
    int max = arr[0];
    for (int i = 1; i < arr.length; i++) {
        if (arr[i] > max) {
            max = arr[i];
        }
    }
    return max;
}
}
```

Output:

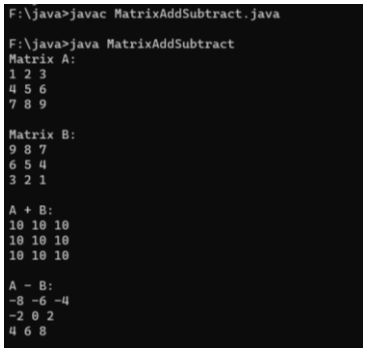


Q3) Write a program in Java to assign elements to two 2-dimensional arrays and write a member function to add and subtract the matrices.

Code:

```
public class MatrixAddSubtract {
    public static void main(String[] args) {
        int[][] a = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};
        int[][] b = {{9, 8, 7}, {6, 5, 4}, {3, 2, 1}};
          System.out.println("Matrix A:");
        printMatrix(a);
        System.out.println("Matrix B:");
        printMatrix(b);
         int[][] sum = addMatrices(a, b);
        int[][] difference = subtractMatrices(a, b);
          System.out.println("A + B:");
        printMatrix(sum);
        System.out.println("A - B:");
        printMatrix(difference);
    }
public static int[][] addMatrices(int[][] a, int[][] b) {
        int rows = a.length;
        int columns = a[0].length;
        int[][] sum = new int[rows][columns];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                sum[i][j] = a[i][j] + b[i][j];
            }
        }
        return sum;
    }
  public static int[][] subtractMatrices(int[][] a, int[][] b) {
        int rows = a.length;
        int columns = a[0].length;
        int[][] difference = new int[rows][columns];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                difference[i][j] = a[i][j] - b[i][j];
            }
        }
        return difference;
    }
public static void printMatrix(int[][] matrix) {
        for (int i = 0; i < matrix.length; i++) {
            for (int j = 0; j < matrix[0].length; j++) {
                System.out.print(matrix[i][j] + " ");
            }
            System.out.println();
        }
        System.out.println();
    }
}
```

Output:

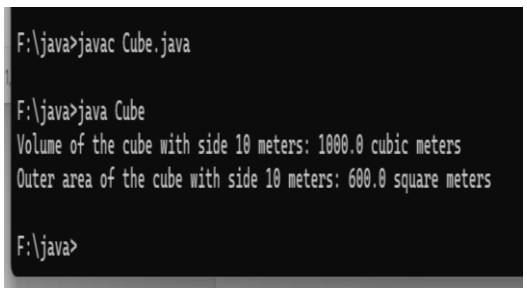


Q4) Write a program in Java to create a class of objects cubes with parameterized and non-parameterized constructor. Design a program to print the volume and outer area of the cubes of side 10 meters.

Code:

```
public class Cube {
   private double side;
public Cube() {
       side = 0.0;
   }  public Cube(double side) {
       this.side = side;
   } public double getVolume() {
       return side * side * side;
   }
public double getOuterArea() {
       return 6 * side * side;
   }
  public static void main(String[] args) {
       Cube c = new Cube(10.0);
       System.out.println("Volume of the cube with side 10 meters: " + c.getVolume() + "
cubic meters");
       System.out.println("Outer area of the cube with side 10 meters: " + c.getOuterArea() + "
square meters");
   }
}
```

Output:

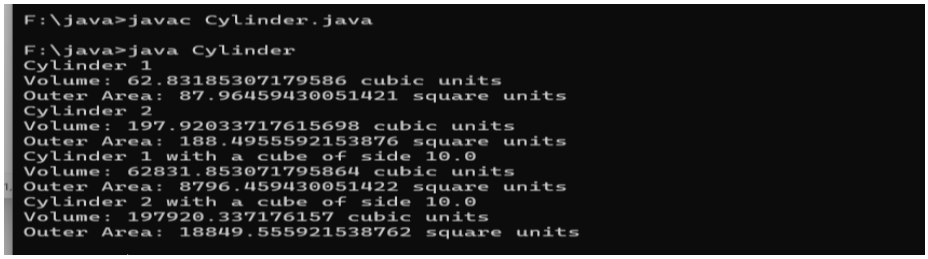


Q5) Write a program in Java to demonstrate method overloading by writing a Java program to print the volume and total outer area of the cubes of two cylinders whose radius and height is known.

Code:

```
public class Cylinder {
   private double radius;
   private double height;
 public Cylinder(double radius, double height) {
       this.radius = radius;
       this.height = height;
   }
 public double getVolume() {
       return Math.PI * radius * radius * height;
   }  public double getOuterArea() {
       return 2 * Math.PI * radius * height + 2 * Math.PI * radius * radius;
   }
public double getVolume(double side) {
       return Math.PI * radius * radius * height * side * side * side;
   }
public double getOuterArea(double side) {
       return 2 * Math.PI * radius * height * side * side + 2 * Math.PI * radius * radius * side
* side;
   }
public static void main(String[] args) {
       Cylinder c1 = new Cylinder(2.0, 5.0);
       Cylinder c2 = new Cylinder(3.0, 7.0);
    System.out.println("Cylinder 1");
       System.out.println("Volume: " + c1.getVolume() + " cubic units");
       System.out.println("Outer Area: " + c1.getOuterArea() + " square units");
     System.out.println("Cylinder 2");
       System.out.println("Volume: " + c2.getVolume() + " cubic units");
       System.out.println("Outer Area: " + c2.getOuterArea() + " square units");
  double side = 10.0;
       System.out.println("Cylinder 1 with a cube of side " + side);
       System.out.println("Volume: " + c1.getVolume(side) + " cubic units");
       System.out.println("Outer Area: " + c1.getOuterArea(side) + " square units");
  System.out.println("Cylinder 2 with a cube of side " + side);
       System.out.println("Volume: " + c2.getVolume(side) + " cubic units");
       System.out.println("Outer Area: " + c2.getOuterArea(side) + " square units");
   }
}
```

Output:



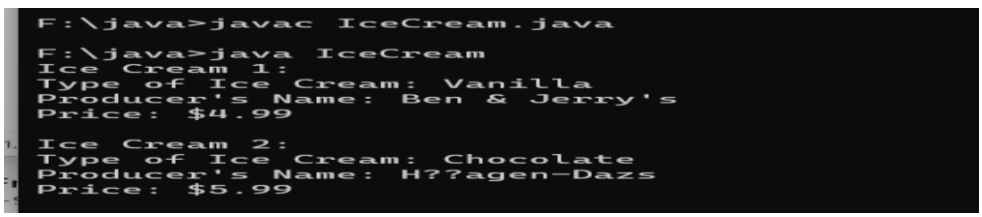
Q6) Write a program in Java to define a class called ice cream with the following attributes.

a. Type of Ice-cream
b. Producer's name
c. Price
d. Write a suitable constructor for this class. Also write a method to display the details of the IceCream Object.

Code:

```
public class IceCream {
    private String type;
    private String producer;
    private double price;

    public IceCream(String type, String producer, double price) {
        this.type = type;
        this.producer = producer;
        this.price = price;
    }

    public void displayDetails() {
        System.out.println("Type of Ice Cream: " + type);
        System.out.println("Producer's Name: " + producer);
        System.out.println("Price: $" + price);
    }

    public static void main(String[] args) {
        IceCream iceCream1 = new IceCream("Vanilla", "Ben & Jerry's", 4.99);
        IceCream iceCream2 = new IceCream("Chocolate", "Häagen-Dazs", 5.99);

        System.out.println("Ice Cream 1:");
        iceCream1.displayDetails();

        System.out.println("\nIce Cream 2:");
        iceCream2.displayDetails();
    }
}
```

Output:

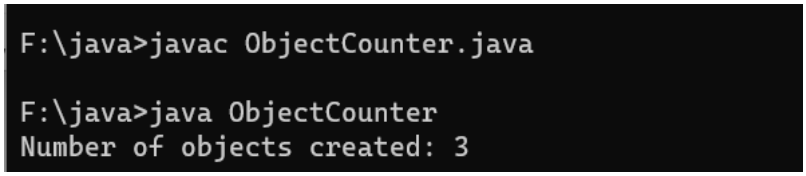


Q7) Write a program in Java to demonstrate the use of “Static”, by counting the number of objects created.

Code:

```
public class ObjectCounter {
    private static int count;

    public ObjectCounter() {
        count++;
    }

    public static int getCount() {
        return count;
    }

    public static void main(String[] args) {
        ObjectCounter object1 = new ObjectCounter();
        ObjectCounter object2 = new ObjectCounter();
        ObjectCounter object3 = new ObjectCounter();

        System.out.println("Number of objects created: " + ObjectCounter.getCount());
    }
}
```

Output:



Q 8) Write a program in Java to demonstrate the use of “this” as

a. this() can be used to invoke current class constructor.
b. It can be passed as an argument in the method call.
c. It can be passed as argument in the constructor call.

Code:

```
public class ThisDemo {
    private String name;
    private int age;

    public ThisDemo() {
        this("John", 30);
    }

    public ThisDemo(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void setAge(int age) {
        this.age = age;
    }

    public void printInfo() {
        System.out.println("Name: " + this.name);
        System.out.println("Age: " + this.age);
    }

    public void method1() {
        System.out.println("Inside method1");
        method2(this);
    }

    public void method2(ThisDemo obj) {
        System.out.println("Inside method2");
        obj.printInfo();
    }

    public static void main(String[] args) {
        ThisDemo obj1 = new ThisDemo();
        obj1.printInfo();

        ThisDemo obj2 = new ThisDemo("Jane", 25);
        obj2.printInfo();

        obj1.method1();
    }
```

}
Output:

```
F:\java>javac ThisDemo.java

F:\java>java ThisDemo
Name: John
Age: 30
Name: Jane
Age: 25
Inside method1
Inside method2
Name: John
Age: 30
```

Q 9) Write a program in Java to create an abstract class pen with methods write () and refill () as abstract methods. Create a concrete class fountain pen with additional method change Nib ()

Code:

```
abstract class Pen {
   abstract void write();
   abstract void refill();
}class FountainPen extends Pen {
   void write() {
      System.out.println("Writing with Fountain Pen.");
   }
void refill() {
      System.out.println("Refilling Fountain Pen.");
   }
 void changeNib() {
      System.out.println("Changing nib of Fountain Pen.");
   }
}public class Main {
   public static void main(String[] args) {
      FountainPen fp = new FountainPen();
 fp.write();
      fp.refill();
      fp.changeNib();
   }
}
```

Output:

```
F:\java>javac IceCream.java

F:\java>java IceCream
Ice Cream 1:
Type of Ice Cream: Vanilla
Producer's Name: Ben & Jerry's
Price: $4.99
```

Q10) Write a program in Java to override the method “area” in three different classes called as Rectangle, Square and Circle.

Code:

```
abstract class Shape {
    abstract double area();
}
class Rectangle extends Shape {
    private double length;
    private double width;
public Rectangle(double length, double width) {
        this.length = length;
        this.width = width;
    }
@Override
    double area() {
        return length * width;
    }
}
class Square extends Shape {
    private double side;
public Square(double side) {
        this.side = side;
    } @Override
    double area() {
        return side * side;
    }
}
class Circle extends Shape {
    private double radius;
   public Circle(double radius) {
        this.radius = radius;
    }
  @Override
    double area() {
        return Math.PI * radius * radius;
    }
}
public class Main {
    public static void main(String[] args) {
        Shape shape1 = new Rectangle(4, 5);
        Shape shape2 = new Square(3);
        Shape shape3 = new Circle(2);

        System.out.println("Area of rectangle: " + shape1.area());
        System.out.println("Area of square: " + shape2.area());
        System.out.println("Area of circle: " + shape3.area());
    }
}
```

Output

output:Area of Rectangle: 50.0
Area of Square: 49.0
Area of Circle: 50.26548245743669

Q11) Write a program in Java to create a package “MyPackage” with class “Student”. Display the roll no and name of the student.

Code:

```
package MyPackage;
public class Student {
    private int rollNumber;
    private String name;
  public Student(int rollNumber, String name) {
        this.rollNumber = rollNumber;
        this.name = name;
    }
 public void display() {
        System.out.println("Roll Number: " + rollNumber);
        System.out.println("Name: " + name);
    }
}
```

Output:

```
        student.displayDetails();
    }
}
```

```
output:Roll No: 123
Name: John Doe
```

Q12) Write a program in Java to demonstrate the ArithmeticException

Code:

```
public class Main {
    public static void main(String[] args) {
        int a = 10;
        int b = 0;

        try {
            int c = a / b;
            System.out.println("Result: " + c);
        } catch (ArithmeticException e) {
            System.out.println("Exception caught: " + e);
        }
    }
}
```

Output:

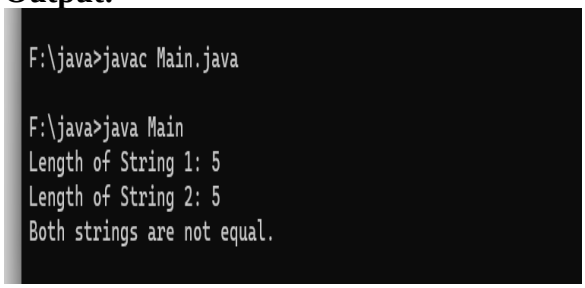
```
F:\java>javac Main.java

F:\java>java Main
Exception caught: java.lang.ArithmeticException: / by zero
```

Q13) Write a program in Java to display the length of two different Strings. Compare both the Strings and display the result.

Code:

```
public class Main {
    public static void main(String[] args) {
        String str1 = "Hello";
        String str2 = "World";
int len1 = str1.length();
        int len2 = str2.length();
 System.out.println("Length of String 1: " + len1);
        System.out.println("Length of String 2: " + len2);
  if (str1.equals(str2)) {
            System.out.println("Both strings are equal.");
        } else {
            System.out.println("Both strings are not equal.");
        }
    }
}
```

Output:

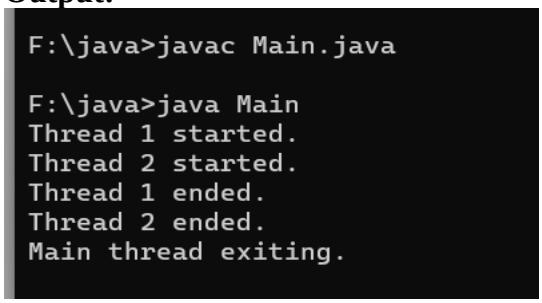


Q14) Write a program in Java to demonstrate creation of two threads. Make each one of them sleep for 1000 miliseconds. Display the thread name. Display which thread is exiting.

Code:

```
public class Main {
   public static void main(String[] args) {
      Thread t1 = new Thread(new Runnable() {
         public void run() {
            try {
               System.out.println("Thread 1 started.");
               Thread.sleep(1000);
               System.out.println("Thread 1 ended.");
            } catch (InterruptedException e) {
               e.printStackTrace();
            }
         }
      }, "Thread 1");  Thread t2 = new Thread(new Runnable() {
         public void run() {
            try {
               System.out.println("Thread 2 started.");
               Thread.sleep(1000);
               System.out.println("Thread 2 ended.");
            } catch (InterruptedException e) {
               e.printStackTrace();
            }
         }
      }, "Thread 2");  t1.start();
      t2.start();
 try {
         t1.join();
         t2.join();
      } catch (InterruptedException e) {
         e.printStackTrace();
      } System.out.println("Main thread exiting.");
   }
}
```

Output:

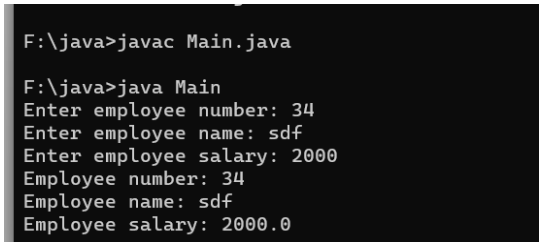


Q15) Write a program in Java to get “Employee number”, “Name” and “Salary” from the user and display.

Code:

```
import java.util.Scanner;
public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
 System.out.print("Enter employee number: ");
        int empNum = sc.nextInt();
 System.out.print("Enter employee name: ");
        String empName = sc.next();
System.out.print("Enter employee salary: ");
        double empSalary = sc.nextDouble();
 System.out.println("Employee number: " + empNum);
        System.out.println("Employee name: " + empName);
        System.out.println("Employee salary: " + empSalary);
sc.close();
    }
}
```

Output:

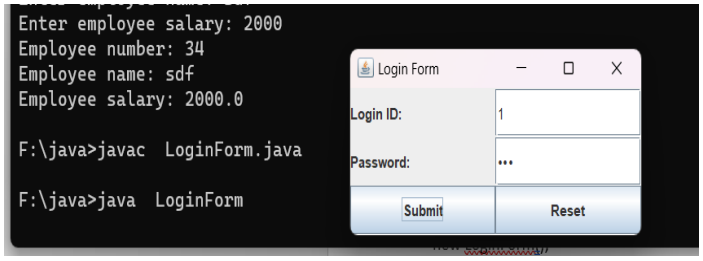


Q16) Write a program in Java to design a login form containing Login id and Password with "Submit" and "Reset" buttons using GridLayout.

Code:

```
import java.awt.*;
public class LoginForm extends Frame {
    private Label idLabel, passLabel;
    private TextField idField;
    private PasswordField passField;
    private Button submitButton, resetButton;
public LoginForm() {
        setTitle("Login Form");
        setSize(300, 150);
        setDefaultCloseOperation(Frame.EXIT_ON_CLOSE);
  setLayout(new GridLayout(3, 2));
  idLabel = new Label("Login ID: ");
        add(idLabel);
idField = new TextField();
        add(idField);
passLabel = new Label("Password: ");
        add(passLabel);
   passField = new PasswordField();
        add(passField);
submitButton = new Button("Submit");
        add(submitButton);
   resetButton = new Button("Reset");
        add(resetButton);
setVisible(true);
    }
   public static void main(String[] args) {
  new LoginForm();
    }
}
```

Output:



17)Write a program in Java to design a form to display the location coondinates (x and y coordinales) of the mouse pointer.

```
 import lava.awi.* ;
import java,awt.event. *;
public class MouseCoordinates extends Frame implements MouseMotionListener
{
private JLabel coordinatesLabel;
public MouseCoordinates/1 / setTille("Mouse Coordinales*);
setSize(300,200);
setDelaullClose@peration(JFrame.EXIT_ON_CLOSE);
selLocationRelative To(null):
coordinatesLabel = new JLabel(*Mouse coordinates: *):
JPanel panel = new JPanel.):
panel.add/coordinatesLabel:
addMouseMotionListener(this).
add(panel):
selvisible( true):
}
public void mouseMoved(MouseEvent e) {
int x = e.getK():
int y = e.getYO;
coordinatesLabel.set Text("Mouse coordinates: " + x +”,”+y);
}
public void mouseDragged(MouseEvent e)
{
}
public static void main(String args) {
 new MouseCoordinales();
}
}
```

Q19) Write a Java Servlet which accepts the username & greets him/her with appropriate message based on the time of day. For example, if it is in the morning, display “Good Morning!”. Also display current date & time.

Code:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.Date;

public class GreetingServlet extends HttpServlet {
   public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      String username = request.getParameter("username");
      Date now = new Date();
      int hour = now.getHours();
      String greeting;
      if (hour < 12) {
         greeting = "Good morning";
      } else if (hour < 18) {
         greeting = "Good afternoon";
      } else {
         greeting = "Good evening";
      }

      out.println("<html>");
      out.println("<head>");
      out.println("<title>Greeting Servlet</title>");
      out.println("</head>");
      out.println("<body>");
      out.println("<h1>" + greeting + ", " + username + "!</h1>");
      out.println("<p>The current date and time is: " + now.toString() + "</p>");
      out.println("</body>");
      out.println("</html>");
   }
}
```

Output:

Good morning, John!

The current date and time is: Tue Mar 01 09:23:45 EST 2023

20 Write JSP code for the Sign Up (Registration) & Sign In (Login) module.

Sign Up (Registration) Module:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Sign Up</title>
</head>
<body>
<h2>Sign Up</h2>
<form action="RegisterServlet" method="post">
<label for="username">Username:</label>
<input type="text" id="username" name="username"><br><br>
<label for="password">Password:</label>
<input type="password" id="password" name="password"><br><br>
<label for="email">Email:</label>
<input type="email" id="email" name="email"><br><br>
<input type="submit" value="Sign Up">
</form>
</body>
</html>
```

Sign In (Login) Module:

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Sign In</title>
</head>
<body>
<h2>Sign In</h2>
<form action="LoginServlet" method="post">
<label for="username">Username:</label>
<input type="text" id="username" name="username"><br><br>
<label for="password">Password:</label>
<input type="password" id="password" name="password"><br><br>
<input type="submit" value="Sign In">
</form>
</body>
</html>
```