**Q. 1 Write a simple Java Program to print factorial of a given number using recursion.**

public class FactorialRecursive {

public static int factorial(int number) {

if (number == 0) {

return 1; } else {

return number \* factorial(number - 1);

} }

public static void main(String[] args) {

int number = 5;

int factorialResult = factorial(number);

System.out.println("Factorial of " + number + " is: " + factorialResult);

} }

**Q. 1. Write a simple Java program to generate 5 random numbers.**

import java.util.Random;

public class RandomNumbers {

public static void main(String[] args) {

Random random = new Random();

for (int i = 0; i < 5; i++) {

int randomNumber = random.nextInt(); System.out.println("Random number " + (i + 1) + ": " + randomNumber); } }

}

**Q. Write a program to design Registration process form using Applet and AWT components**

import java.applet.Applet;

import java.awt.Button;

import java.awt.Label;

import java.awt.TextField;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class RegistrationForm extends Applet implements ActionListener {private TextField nameTextField, emailTextField;

private Button registerButton;

public void init() {

Label nameLabel = new Label("Name:");

Label emailLabel = new Label("Email:");

nameTextField = new TextField(20);

emailTextField = new TextField(20);

registerButton = new Button("Register");

registerButton.addActionListener(this);

add(nameLabel);

add(nameTextField);

add(emailLabel);

add(emailTextField);

add(registerButton);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == registerButton) {

String name = nameTextField.getText();

String email = emailTextField.getText();

showStatus("Registration successful for " + name + " with email " + email);}

} }

**Q. 1. Write a Java Program to implement stack using Queue interface**

import java.util.LinkedList;

import java.util.Queue;

public class StackUsingQueue {

private Queue<Integer> queue1;

private Queue<Integer> queue2;

public StackUsingQueue() {

queue1 = new LinkedList<>();

queue2 = new LinkedList<>();

} public void push(int value) {

if (queue1.isEmpty()) {

queue1.add(value);

} else {

while (!queue1.isEmpty()) {

queue2.add(queue1.remove());

} while (!queue2.isEmpty()) {

queue1.add(queue2.remove());

}}} public int pop() {

if (queue1.isEmpty()) {

throw new IllegalStateException("Stack is empty"); }return queue1.remove();

} public int top() {

if (queue1.isEmpty()) {

throw new IllegalStateException("Stack is empty"); }

return queue1.peek();

} public boolean isEmpty() {

return queue1.isEmpty();

} public int size() {

return queue1.size();

}public static void main(String[] args) {

StackUsingQueue stack = new StackUsingQueue();

stack.push(10);

stack.push(20);

stack.push(30);

System.out.println("Stack size: " + stack.size());

System.out.println("Top element: " + stack.top());

System.out.println("Popped element: " + stack.pop());

System.out.println("Popped element: " + stack.pop());System.out.println("Is stack empty? " + stack.isEmpty());

} }

**Q. 2 Write a JAVA Servlet Program to implement and demonstrate get () and Post () methods (Using HTTP Servlet Class).**

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class MyServlet extends HttpServlet {

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

String name = request.getParameter("name");

String htmlResponse = "<html><body>";

htmlResponse += "<h2>Hello, " + name + " (GET)</h2>";

htmlResponse += "</body></html>";

response.getWriter().print(htmlResponse);

} protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

String name = request.getParameter("name");

String htmlResponse = "<html><body>";

htmlResponse += "<h2>Hello, " + name + " (POST)</h2>";

htmlResponse += "</body></html>"; response.getWriter().print(htmlResponse);

}

}

**Q. 1 Write a Java Program to Implement stack using Stack class**

import java.util.Stack;

public class StackExample {

public static void main(String[] args) {

Stack<Integer> stack = new Stack<>();

stack.push(10);

stack.push(20);

stack.push(30);

System.out.println("Stack elements: " + stack);

int poppedElement = stack.pop();

System.out.println("Popped element: " + poppedElement);

System.out.println("Stack after popping: " + stack); int topElement = stack.peek();

System.out.println("Top element: " + topElement);

boolean isEmpty = stack.isEmpty();

System.out.println("Is stack empty? " + isEmpty);

int size = stack.size();

System.out.println("Size of the stack: " + size);

}

}

**Q. 2 Write JSP Program to validate username and password**

<%@ page language="java" %>

<!DOCTYPE html>

<html>

<head>

<title>Login Page</title>

</head>

<body>

<h1>Login Page</h1>

<%-- Retrieve the submitted username and password --%>

<%

String submittedUsername = request.getParameter("username");

String submittedPassword = request.getParameter("password");

String validUsername = "admin";

String validPassword = "password";

if (submittedUsername.equals(validUsername) && submittedPassword.equals(validPassword)) {

out.println("<p>Login successful!</p>");

} else {

out.println("<p>Login failed. Invalid username or password.</p>");}%>

<%-- Display the login form --%>

<form method="post" action="login.jsp">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required><br><br>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required><br><br>

<input type="submit" value="Login">

</form></body></html>

**Q. 2 Write JSP program to print current date & time**

<%@ page import="java.util.Date" %>

<!DOCTYPE html>

<html>

<head>

<title>Print Current Date and Time</title>

</head>

<body>

<h1>Current Date and Time:</h1>

<%

Date currentDate = new Date();

out.println("<p>" + currentDate.toString() + "</p>");

%>

</body>

</html>

**Q. 1. Write a java Program to implement thread using runnable interface**

public class MyRunnable implements Runnable {

public void run() {

System.out.println("Thread is running.");

} public static void main(String[] args) {

MyRunnable myRunnable = new MyRunnable();

Thread thread = new Thread(myRunnable);

thread.start();

} }

**Q. 2. Write a Java program to implement student information in a file and perform the operations on it**

import java.io.\*;

import java.util.\*;

public class StudentInformation {

private static final String FILE\_NAME = "student.txt";

public static void main(String[] args) {

// Write student information to a file

writeStudentInfoToFile();

// Read student information from the file

List<String> studentInfo = readStudentInfoFromFile();

// Display student information

displayStudentInfo(studentInfo);

}

public static void writeStudentInfoToFile() {

try {

FileWriter fileWriter = new FileWriter(FILE\_NAME);

BufferedWriter bufferedWriter = new BufferedWriter(fileWriter);

// Write student information to the file

bufferedWriter.write("John Doe, 20, A+");

bufferedWriter.newLine();

bufferedWriter.write("Jane Smith, 22, B");

bufferedWriter.newLine();

bufferedWriter.close();

} catch (IOException e) {

e.printStackTrace();

}

}

public static List<String> readStudentInfoFromFile() {

List<String> studentInfo = new ArrayList<>();

try {

FileReader fileReader = new FileReader(FILE\_NAME);

BufferedReader bufferedReader = new BufferedReader(fileReader);

String line;

while ((line = bufferedReader.readLine()) != null) {

// Add each line of student information to the list

studentInfo.add(line);

}

bufferedReader.close();

} catch (IOException e) {

e.printStackTrace();

}

return studentInfo;

} public static void displayStudentInfo(List<String> studentInfo) {

System.out.println("Student Information:");

for (String info : studentInfo) {

System.out.println(info);

}}}