CU
CHANDIGARH UNIVERSITY

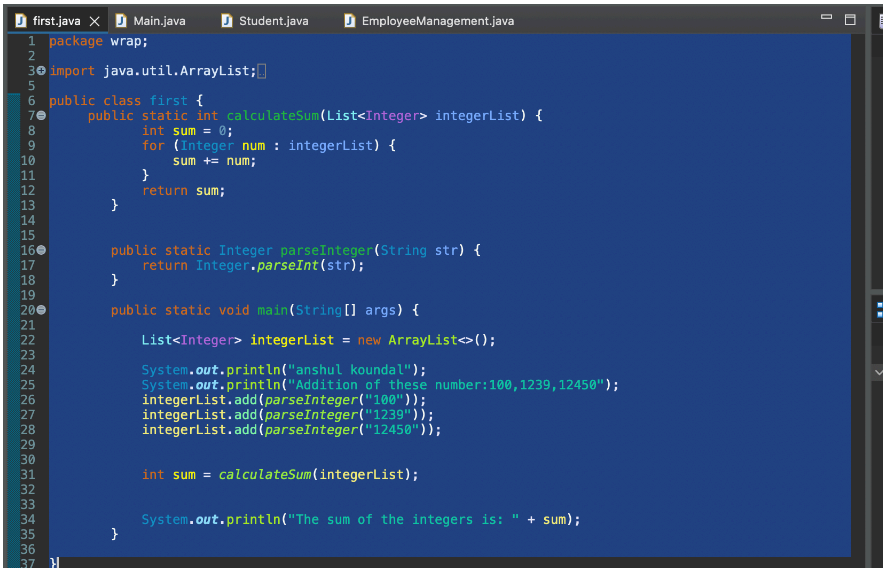
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

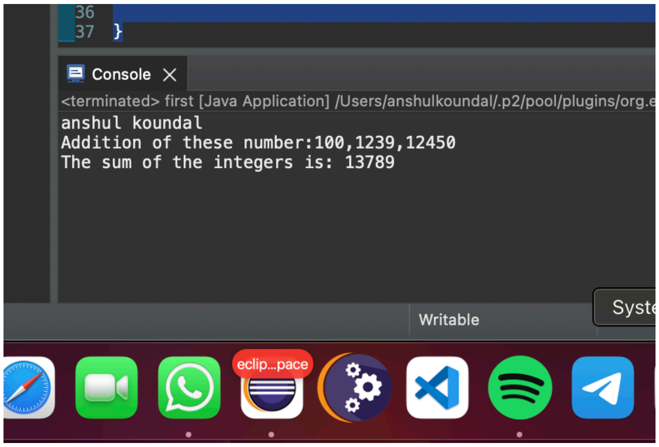
Name :-Anshul koundal
UID:-22BCS16818
CLASS:-22BCS-IOT-619 B

#CODE

```
package wrap;
import java.util.ArrayList;
import java.util.List;
public class first {
   public static int calculateSum(List<Integer> integerList) {
        int sum = 0;
        for (Integer num : integerList) {
           sum += num;
        }
        return sum;
     }
     public static Integer parseInteger(String str) {
        return Integer.parseInt(str);
     }
     public static void main(String[] args) {
        List<Integer> integerList = new ArrayList<>();
        System.out.println("anshul koundal");
        System.out.println("Addition of these number:100,1239,12450");
        integerList.add(parseInteger("100"));
        integerList.add(parseInteger("1239"));
        integerList.add(parseInteger("12450"));
        int sum = calculateSum(integerList);
        System.out.println("The sum of the integers is: " + sum);
}}
```



```
package wrap;

import java.util.ArrayList;

public class first {
     public static int calculateSum(List<Integer> integerList) {
            int sum = 0;
            for (Integer num : integerList) {
                sum += num;
            }
            return sum;
        }


        public static Integer parseInteger(String str) {
            return Integer.parseInt(str);
        }

        public static void main(String[] args) {

            List<Integer> integerList = new ArrayList<>();

            System.out.println("anshul koundal");
            System.out.println("Addition of these number:100,1239,12450");
            integerList.add(parseInteger("100"));
            integerList.add(parseInteger("1239"));
            integerList.add(parseInteger("12450"));


            int sum = calculateSum(integerList);


            System.out.println("The sum of the integers is: " + sum);
        }

}
```



```
}
```

```
<terminated> first [Java Application] /Users/anshulkoundal/.p2/pool/plugins/org.e
anshul koundal
Addition of these number:100,1239,12450
The sum of the integers is: 13789
```

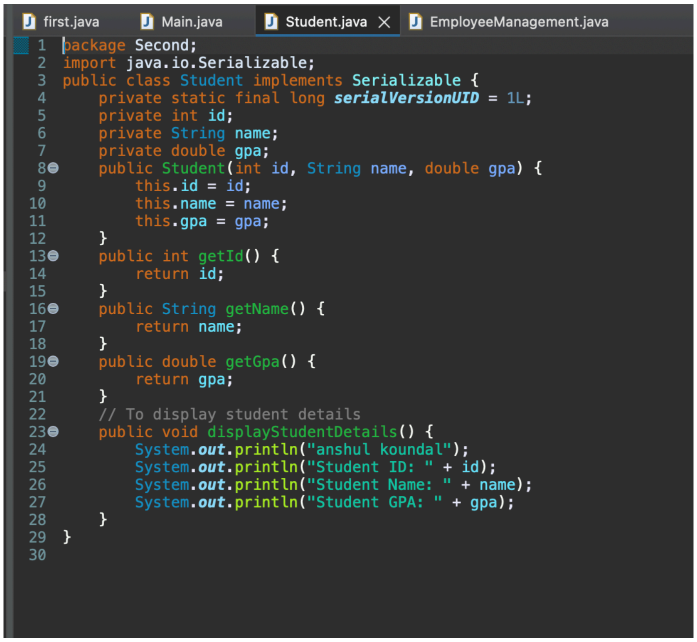
#code 2

———————————-main.java——————————

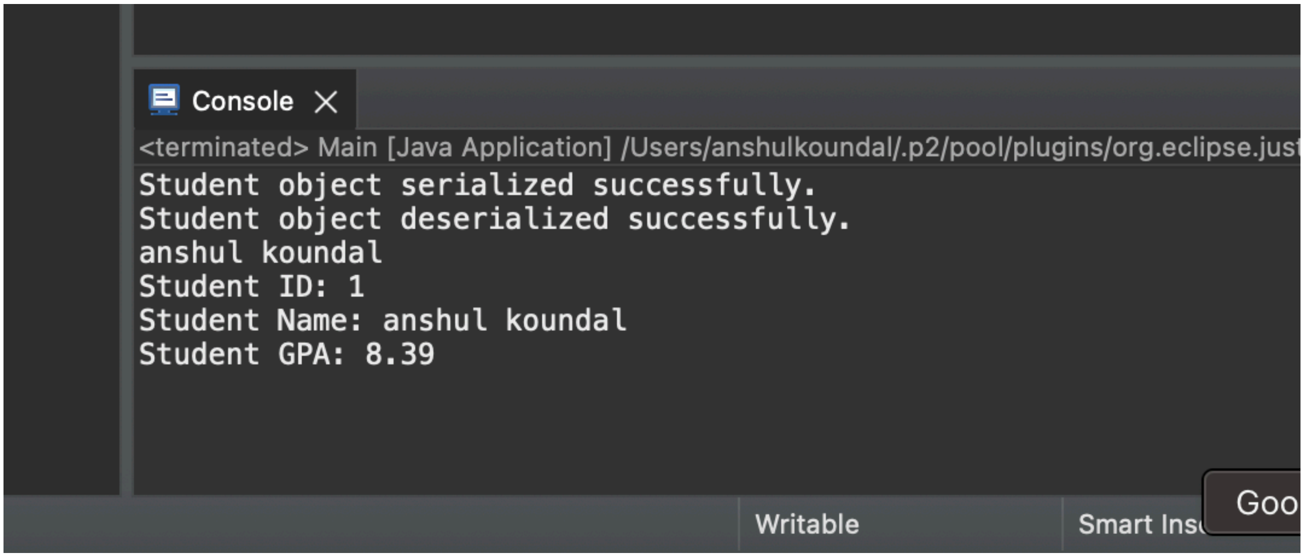
```
package Second;
import java.io.*;
public class Main {
    public static void serializeStudent(Student student, String filename) {
        try (ObjectOutputStream out = new ObjectOutputStream(new FileOutputStream(filename))) {
            out.writeObject(student);
            System.out.println("Student object serialized successfully.");
        } catch (FileNotFoundException e) {
            System.out.println("File not found: " + e.getMessage());
        } catch (IOException e) {
            System.out.println("IOException occurred: " + e.getMessage());
        }
    }
    public static Student deserializeStudent(String filename) {
        Student student = null;
        try (ObjectInputStream in = new ObjectInputStream(new FileInputStream(filename))) {
            student = (Student) in.readObject();  // Deserializes the student object
            System.out.println("Student object deserialized successfully.");
        } catch (FileNotFoundException e) {
            System.out.println("File not found: " + e.getMessage());
        } catch (IOException e) {
            System.out.println("IOException occurred: " + e.getMessage());
        } catch (ClassNotFoundException e) {
            System.out.println("Class not found: " + e.getMessage());
        }
        return student;
    }
    public static void main(String[] args) {

        Student student = new Student(1, "anshul koundal ", 8.39);
        String filename = "student.ser";
        serializeStudent(student, filename);
        Student deserializedStudent = deserializeStudent(filename);
        if (deserializedStudent != null) {
            deserializedStudent.displayStudentDetails();
        }}}
```

——————————Student.java——————————-

```
package Second;
import java.io.Serializable;
public class Student implements Serializable {
    private static final long serialVersionUID = 1L;
    private int id;
    private String name;
    private double gpa;
    public Student(int id, String name, double gpa) {
        this.id = id;
        this.name = name;
        this.gpa = gpa;
    }
    public int getId() {
        return id;
    }
    public String getName() {
        return name;
    }
    public double getGpa() {
        return gpa;
    }
    // To display student details
    public void displayStudentDetails() {
     System.out.println("anshul koundal");
        System.out.println("Student ID: " + id);
        System.out.println("Student Name: " + name);
        System.out.println("Student GPA: " + gpa);
    }
}
```



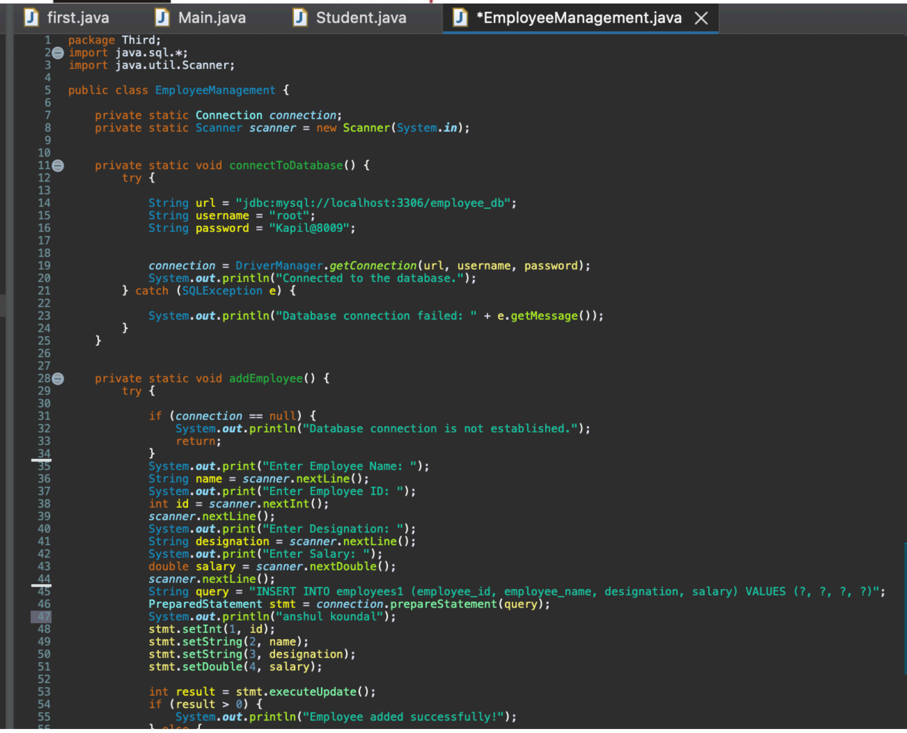
```
package Second;
import java.io.Serializable;
public class Student implements Serializable {
    private static final long serialVersionUID = 1L;
    private int id;
    private String name;
    private double gpa;
    public Student(int id, String name, double gpa) {
        this.id = id;
        this.name = name;
        this.gpa = gpa;
    }
    public int getId() {
        return id;
    }
    public String getName() {
        return name;
    }
    public double getGpa() {
        return gpa;
    }
    // To display student details
    public void displayStudentDetails() {
        System.out.println("anshul koundal");
        System.out.println("Student ID: " + id);
        System.out.println("Student Name: " + name);
        System.out.println("Student GPA: " + gpa);
    }
}
```



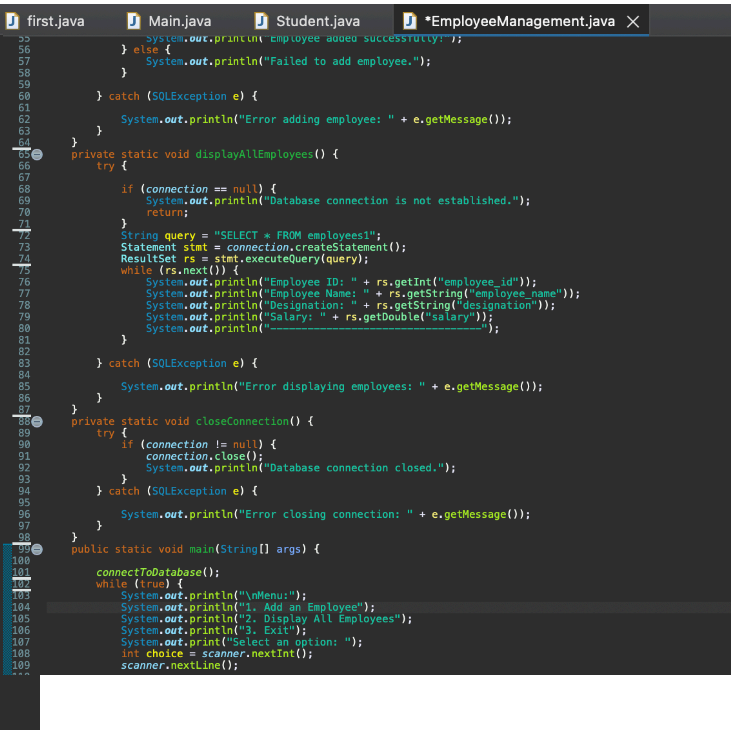
```
<terminated> Main [Java Application] /Users/anshulkoundal/.p2/pool/plugins/org.eclipse.just
Student object serialized successfully.
Student object deserialized successfully.
anshul koundal
Student ID: 1
Student Name: anshul koundal
Student GPA: 8.39
```

#code 3

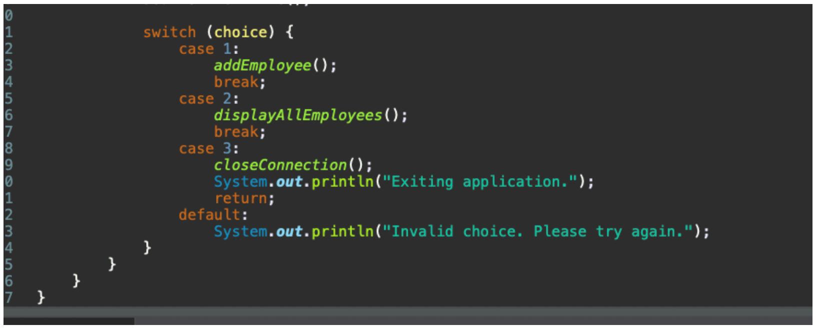
```
package Third;
import java.sql.*;
import java.util.Scanner;

public class EmployeeManagement {

    private static Connection connection;
    private static Scanner scanner = new Scanner(System.in);


    private static void connectToDatabase() {
        try {

            String url = "jdbc:mysql://localhost:3306/employee_db";
            String username = "root";
            String password = "Kapil@8009";


            connection = DriverManager.getConnection(url, username, password);
            System.out.println("Connected to the database.");
        } catch (SQLException e) {

            System.out.println("Database connection failed: " + e.getMessage());
        }
    }


    private static void addEmployee() {
        try {

            if (connection == null) {
                System.out.println("Database connection is not established.");
                return;
            }
            System.out.print("Enter Employee Name: ");
            String name = scanner.nextLine();
            System.out.print("Enter Employee ID: ");
            int id = scanner.nextInt();
            scanner.nextLine();
            System.out.print("Enter Designation: ");
            String designation = scanner.nextLine();
            System.out.print("Enter Salary: ");
            double salary = scanner.nextDouble();
            scanner.nextLine();
            String query = "INSERT INTO employees1 (employee_id, employee_name, designation, salary) VALUES
(?, ?, ?, ?)";
            PreparedStatement stmt = connection.prepareStatement(query);
            System.out.println("anshul koundal");
            stmt.setInt(1, id);
            stmt.setString(2, name);
            stmt.setString(3, designation);
            stmt.setDouble(4, salary);

            int result = stmt.executeUpdate();
            if (result > 0) {
                System.out.println("Employee added successfully!");
            } else {
                System.out.println("Failed to add employee.");
            }

        } catch (SQLException e) {

            System.out.println("Error adding employee: " + e.getMessage());
        }
    }
    private static void displayAllEmployees() {
        try {

            if (connection == null) {
                System.out.println("Database connection is not established.");
                return;
            }
            String query = "SELECT * FROM employees1";
            Statement stmt = connection.createStatement();
            ResultSet rs = stmt.executeQuery(query);
            while (rs.next()) {
                System.out.println("Employee ID: " + rs.getInt("employee_id"));
                System.out.println("Employee Name: " + rs.getString("employee_name"));
                System.out.println("Designation: " + rs.getString("designation"));
                System.out.println("Salary: " + rs.getDouble("salary"));
                System.out.println("---------------------------------");
            }

        } catch (SQLException e) {

            System.out.println("Error displaying employees: " + e.getMessage());
        }
    }
    private static void closeConnection() {
        try {
            if (connection != null) {
                connection.close();
                System.out.println("Database connection closed.");
            }
        } catch (SQLException e) {

            System.out.println("Error closing connection: " + e.getMessage());
        }
    }
    public static void main(String[] args) {

        connectToDatabase();
        while (true) {
            System.out.println("\nMenu:");
            System.out.println("1. Add an Employee");
            System.out.println("2. Display All Employees");
            System.out.println("3. Exit");
            System.out.print("Select an option: ");
            int choice = scanner.nextInt();
            scanner.nextLine();

            switch (choice) {
                case 1:
                    addEmployee();
                    break;
                case 2:
                    displayAllEmployees();
                    break;
                case 3:
                    closeConnection();
                    System.out.println("Exiting application.");
                    return;
                default:
                    System.out.println("Invalid choice. Please try again.");
            }
        }
    }
}
```



```java
package Third;
import java.sql.*;
import java.util.Scanner;

public class EmployeeManagement {

    private static Connection connection;
    private static Scanner scanner = new Scanner(System.in);


    private static void connectToDatabase() {
        try {

            String url = "jdbc:mysql://localhost:3306/employee_db";
            String username = "root";
            String password = "Kapil@8009";


            connection = DriverManager.getConnection(url, username, password);
            System.out.println("Connected to the database.");
        } catch (SQLException e) {

            System.out.println("Database connection failed: " + e.getMessage());
        }
    }


    private static void addEmployee() {
        try {

            if (connection == null) {
                System.out.println("Database connection is not established.");
                return;
            }
            System.out.print("Enter Employee Name: ");
            String name = scanner.nextLine();
            System.out.print("Enter Employee ID: ");
            int id = scanner.nextInt();
            scanner.nextLine();
            System.out.print("Enter Designation: ");
            String designation = scanner.nextLine();
            System.out.print("Enter Salary: ");
            double salary = scanner.nextDouble();
            scanner.nextLine();
            String query = "INSERT INTO employees1 (employee_id, employee_name, designation, salary) VALUES (?, ?, ?, ?)";
            PreparedStatement stmt = connection.prepareStatement(query);
            System.out.println("anshul koundal");
            stmt.setInt(1, id);
            stmt.setString(2, name);
            stmt.setString(3, designation);
            stmt.setDouble(4, salary);

            int result = stmt.executeUpdate();
            if (result > 0) {
                System.out.println("Employee added successfully!");
```

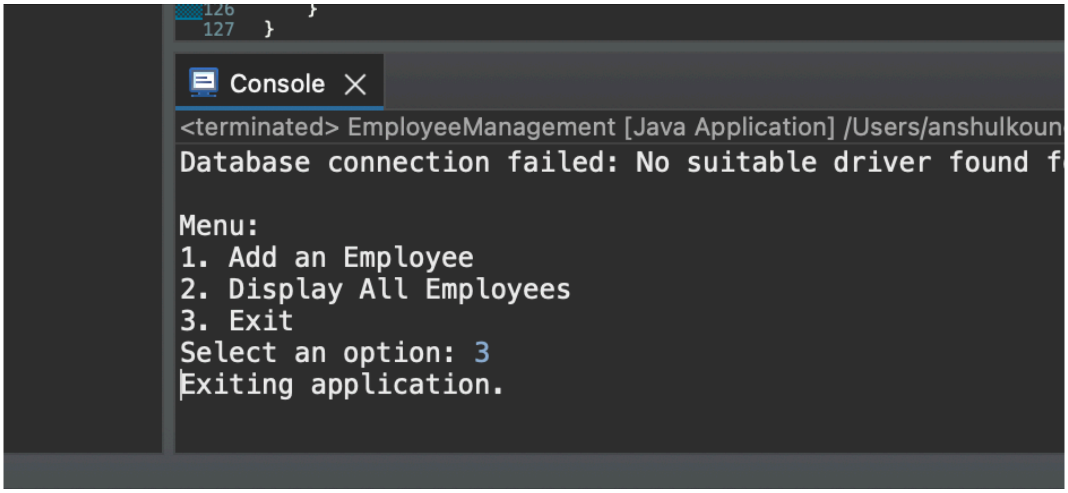


```java
                System.out.println("Employee added successfully!");
            } else {
                System.out.println("Failed to add employee.");
            }

        } catch (SQLException e) {

            System.out.println("Error adding employee: " + e.getMessage());
        }
    }
    private static void displayAllEmployees() {
        try {

            if (connection == null) {
                System.out.println("Database connection is not established.");
                return;
            }
            String query = "SELECT * FROM employees1";
            Statement stmt = connection.createStatement();
            ResultSet rs = stmt.executeQuery(query);
            while (rs.next()) {
                System.out.println("Employee ID: " + rs.getInt("employee_id"));
                System.out.println("Employee Name: " + rs.getString("employee_name"));
                System.out.println("Designation: " + rs.getString("designation"));
                System.out.println("Salary: " + rs.getDouble("salary"));
                System.out.println("--------------------------------");
            }

        } catch (SQLException e) {

            System.out.println("Error displaying employees: " + e.getMessage());
        }
    }
    private static void closeConnection() {
        try {
            if (connection != null) {
                connection.close();
                System.out.println("Database connection closed.");
            }
        } catch (SQLException e) {

            System.out.println("Error closing connection: " + e.getMessage());
        }
    }
    public static void main(String[] args) {

        connectToDatabase();
        while (true) {
            System.out.println("\nMenu:");
            System.out.println("1. Add an Employee");
            System.out.println("2. Display All Employees");
            System.out.println("3. Exit");
            System.out.print("Select an option: ");
            int choice = scanner.nextInt();
            scanner.nextLine();
```



```java

            switch (choice) {
                case 1:
                    addEmployee();
                    break;
                case 2:
                    displayAllEmployees();
                    break;
                case 3:
                    closeConnection();
                    System.out.println("Exiting application.");
                    return;
                default:
                    System.out.println("Invalid choice. Please try again.");
            }
        }
    }
}
```

Output



```java
        }
    }
```

```
<terminated> EmployeeManagement [Java Application] /Users/anshulkoun
Database connection failed: No suitable driver found f

Menu:
1. Add an Employee
2. Display All Employees
3. Exit
Select an option: 3
Exiting application.
```