

PROGRAM CODE:

```
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

class Classroom {
    private List<String> students;
    private Map<String, String> attendance;

    public Classroom() {
        students = new ArrayList<>();
        attendance = new HashMap<>();
    }

    public void addStudent(String studentName) {
        if (studentName != null && !studentName.isEmpty()) {
            students.add(studentName);
            attendance.put(studentName, "Absent");
            System.out.println("Student " + studentName + " has been added.");
        }
    }

    public void markAttendance(String studentName, String status) {
        if (students.contains(studentName) && (status.equals("Present") || status.equals("Absent"))) {
            attendance.put(studentName, status);
            System.out.println("Attendance for " + studentName + " has been updated to " + status + ".");
        }
    }

    public void displayAttendance() {
```

```
        for (String student : students) {  
            String status = attendance.get(student);  
            System.out.println(student + ": " + status);  
        }  
    }  
  
    public static void main(String[] args) {  
        Classroom classroom = new Classroom();  
        classroom.addStudent("Alice");  
        classroom.addStudent("Bob");  
        classroom.markAttendance("Alice", "Present");  
        classroom.markAttendance("Bob", "Absent");  
        classroom.displayAttendance();  
    }  
}
```

Sample Input:

Alice

Bob

Present

Absent

Sample Output:

Student Alice has been added.

Student Bob has been added.

Attendance for Alice has been updated to Present.

Attendance for Bob has been updated to Absent.

Alice: Present

Bob: Absent

PROGRAM CODE:

```
import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

class DeviceManager {

    private List<Integer> evenDeviceData;

    private List<Integer> oddDeviceData;

    private int[] devices;

    private int[] data;

    public DeviceManager(int[] devices, int[] data) {

        this.devices = devices;

        this.data = data;

        evenDeviceData = new ArrayList<>();

        oddDeviceData = new ArrayList<>();

    }

    public void processEvenDevices() {

        for (int i = 0; i < devices.length; i++) {

            if (devices[i] % 2 == 0) {

                evenDeviceData.add(data[i]);

            }

        }

        System.out.println("Even Devices Data: " + evenDeviceData);

    }

    public void processOddDevices() {

        for (int i = 0; i < devices.length; i++) {

            if (devices[i] % 2 != 0) {

                oddDeviceData.add(data[i]);

            }

        }

    }

}
```

```
        }  
    }  
    System.out.println("Odd Devices Data: " + oddDeviceData);  
}  
}
```

```
class EvenDeviceThread extends Thread {  
    private DeviceManager manager;  
  
    public EvenDeviceThread(DeviceManager manager) {  
        this.manager = manager;  
    }  
  
    @Override  
    public void run() {  
        manager.processEvenDevices();  
    }  
}
```

```
class OddDeviceThread extends Thread {  
    private DeviceManager manager;  
  
    public OddDeviceThread(DeviceManager manager) {  
        this.manager = manager;  
    }  
  
    @Override  
    public void run() {  
        manager.processOddDevices();  
    }  
}
```

```
public class DeviceManagementSystem {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.println("Enter device data in the format: deviceID1 data1 deviceID2 data2 ...  
deviceID10 data10");  
  
        String input = scanner.nextLine();  
        String[] inputs = input.split(" ");  
  
        int[] devices = new int[10];  
        int[] data = new int[10];  
  
        for (int i = 0; i < 10; i++) {  
            devices[i] = Integer.parseInt(inputs[2 * i]);  
            data[i] = Integer.parseInt(inputs[2 * i + 1]);  
        }  
  
        DeviceManager manager = new DeviceManager(devices, data);  
  
        EvenDeviceThread evenThread = new EvenDeviceThread(manager);  
        OddDeviceThread oddThread = new OddDeviceThread(manager);  
  
        evenThread.start();  
        oddThread.start();  
  
        try {  
            evenThread.join();  
            oddThread.join();  
        } catch (InterruptedException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
}
```

```
scanner.close();
```

```
}
```

```
}
```

Sample Input:

1 100 2 200 3 300 4 400 5 500 6 600 7 700 8 800 9 900 10 1000

Sample Output:

Even Devices Data: [200, 400, 600, 800, 1000]

Odd Devices Data: [100, 300, 500, 700, 900]

PROGRAM CODE:

```
import java.util.Scanner;
```

```
class Calculator {
```

```
    private int[] array;
```

```
    public Calculator(int[] array) {
```

```
        this.array = array;
```

```
    }
```

```
    public void divideArrayElements(int divisor) {
```

```
        try {
```

```
            for (int i = 0; i < array.length; i++) {
```

```
                int result = array[i] / divisor;
```

```
                System.out.println("Element " + i + " divided by " + divisor + " is: " + result);
```

```
            }
```

```
        } catch (ArithmeticException e) {
```

```
            System.out.println("Error: Division by zero is not allowed.");
```

```
        }
```

```
    }
```

```
    public void accessElement(int index) {
```

```
        try {
```

```
            System.out.println("Element at index " + index + " is: " + array[index]);
```

```
        } catch (ArrayIndexOutOfBoundsException e) {
```

```
            System.out.println("Error: Index " + index + " is out of bounds.");
```

```
        }
```

```
    }
```

```
public void calculateAverage() {  
    try {  
        int sum = 0;  
        for (int num : array) {  
            sum += num;  
        }  
        double average = (double) sum / array.length;  
        System.out.println("Average of array elements is: " + average);  
    } catch (Exception e) {  
        System.out.println("Error calculating the average.");  
    }  
}  
}
```

```
public class ArrayComputationTool {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        try {  
  
            String[] input = scanner.nextLine().split(" ");  
            int[] array = new int[input.length];  
  
            for (int i = 0; i < input.length; i++) {  
                array[i] = Integer.parseInt(input[i]);  
            }  
  
            Calculator calculator = new Calculator(array);  
  
            int divisor = scanner.nextInt();
```

```
        int index = scanner.nextInt();
        calculator.divideArrayElements(divisor);
        calculator.accessElement(index);
        calculator.calculateAverage();

    } catch (NumberFormatException e) {
        System.out.println("Error: Invalid number format.");
    } catch (Exception e) {
        System.out.println("An unexpected error occurred.");
    } finally {
        scanner.close();
        System.out.println("Computation finished.");
    }
}
}
```

Sample Input:

Enter array elements (space-separated integers): 10 20 30 40

Enter divisor: 5

Enter access index: 2

Sample Output:

Element 0 divided by 5 is: 2

Element 1 divided by 5 is: 4

Element 2 divided by 5 is: 6

Element 3 divided by 5 is: 8

Element at index 2 is: 30

Average of array elements is: 25.0

Computation finished.