

..... "Shahzad Malik"

..... "Shehz Muhammad"

Topic: "OOP Final Project"

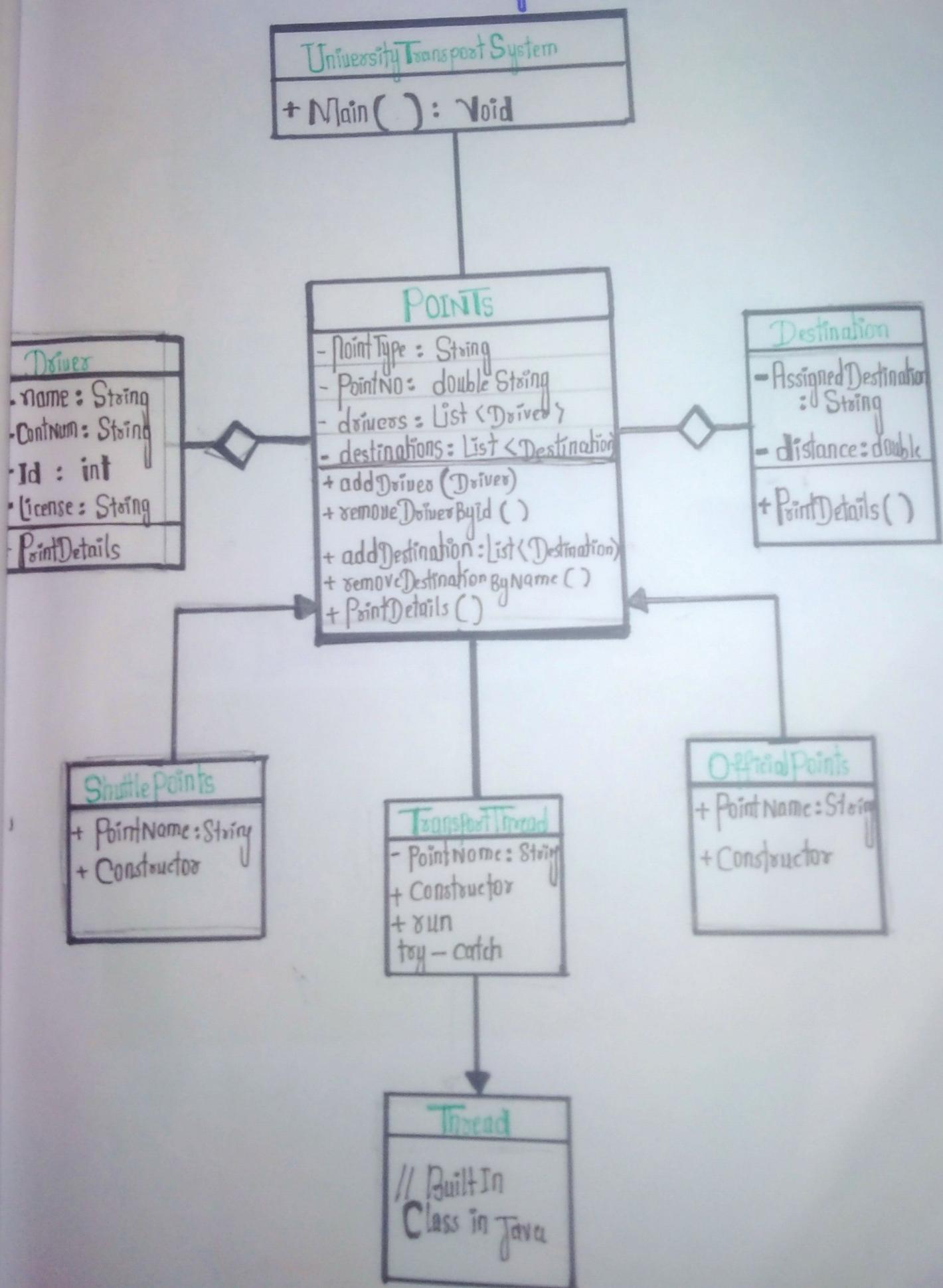
Teacher: "University Transport Management System"

Roll No.: 24MSW071

..... "Dr. Mohsin Memon"

.....“University Transport Management System”.....

.....“Class Diagram”.....



Class Destination:

→ First Class of Project, Name → Destination
Attributes:
• String AssignedDestination
• double distance
Methods
• Getters()
• PrintDetails()

```
officialPoints.java    shuttlesPoints.java    Destination.java    Driver.java    UniversityTransportSystem.java    TransportThread.java
package PROJECT;

public class Destination { 1 usages
    private String AssignedDestination; 3 usages
    private double distance; 2 usages

    public Destination(String AssignedDestination, double distance) { 4 usages
        this.AssignedDestination = AssignedDestination;
        this.distance = distance;
    }
    public String getAssignedDestination() { no usages
        return AssignedDestination;
    }

    public void printDetails() { 1 usage
        System.out.println(" - AssignedDestination: " + AssignedDestination + ", Distance from University: " + distance + " km");
    }
}
```

Class Driver:

→ This class Shows Details of Drivers including their Properties
Attributes:
• String name
• int id
• String contactNumber
• String license
Methods
• Getters
• PrintDetails

```
officialPoints.java    shuttlesPoints.java    Destination.java    Driver.java    UniversityTransportSystem.java    TransportThread.java
package PROJECT;

public class Driver { 0 usages
    private String name; 2 usages
    private String contactNumber; 2 usages
    private int id; 3 usages
    private String license; 2 usages

    public Driver(String name, String contactNumber, int id, String license) { 4 usages
        this.name = name;
        this.contactNumber = contactNumber;
        this.id = id;
        this.license = license;
    }

    public int getId() { 0 usages
        return id;
    }

    public void printDetails() { 0 usages
        System.out.println("Driver name: " + name + " - Driver ID: " + id + " - DRIVER Contact number: " + contactNumber + " - License: " + license);
    }
}
```

→ Both Classes Destination and Driver classes are associated with Points by Aggregation because if we destroy Point then Destination and Driver can exist independently.



Class POINTS:

```
POINTS.java    officialPoints.java    shuttlesPoints.java    Destination.java    Driver.java    UniversityTransportSystem.java    Transport.java
package PROJECT;
import java.util.*;
public class POINTS {
    protected String pointName;
    protected List<Driver> drivers = new ArrayList<>();
    protected List<Destination> destinations = new ArrayList<>();

    public POINTS(String pointName) {
        this.pointName = pointName;
    }

    public void addDriver(Driver driver) {
        drivers.add(driver);
    }

    public void removeDriverById(int id) {
        drivers.removeIf( Driver driver -> driver.getId() == id);
    }

    public void addDestination(Destination dest) {
        destinations.add(dest);
    }

    public void removeDestinationByName(String name) {
        destinations.removeIf( Destination destination -> destination.getAssignedDestination().equalsIgnoreCase(name));
    }
}
```

ORIGINAL PROJECT > src > PROJECT > @POINTS

251 CDRF UTF-8 4 spaces

POINTS... Continue

```
public void addDestination(Destination dest) {
    destinations.add(dest);
}

public void removeDestinationByName(String name) {
    destinations.removeIf( Destination destination -> destination.getAssignedDestination().equalsIgnoreCase(name));
}

public void printDetails() {
    System.out.println("Drivers:");
    for (Driver d : drivers) d.printDetails();
    System.out.println("Destinations:");
    for (Destination d : destinations) d.printDetails();
}
```

Methods In POINT Class

- addDriver()
- removeDriverById()
- addDestination()
- removeDestinationByName()

Class OfficialPoints extends POINTS:

```
POINTS.java officialPoints.java shuttlesPoints.java Destination.java Driver.java UniversityTransportSystem.java
package PROJECT;

public class officialPoints extends POINTS{ 2 usages
    public officialPoints(String pointName) { 2 usages
        super(pointName);
    }
}
```

Class ShuttlesPoints extends POINTS:

```
POINTS.java officialPoints.java shuttlesPoints.java Destination.java Driver.java UniversityTransportSystem.java TransportThread.java
package PROJECT;

public class shuttlesPoints extends POINTS{ 2 usages
    public shuttlesPoints(String pointName) { 2 usages
        super(pointName);
    }
}
```

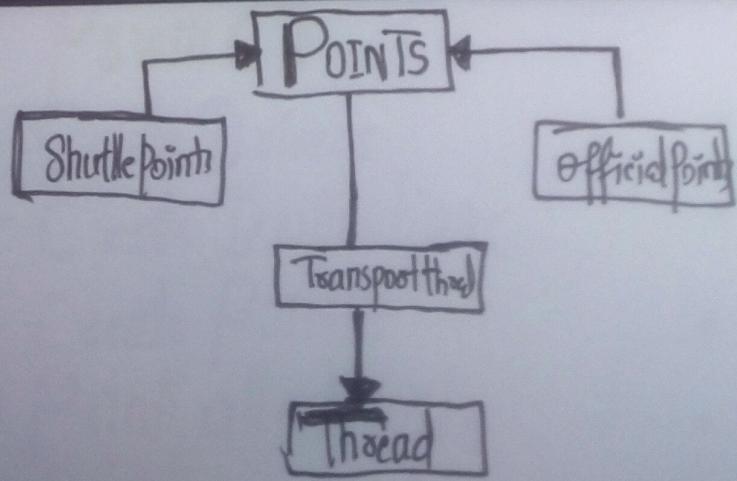
Class TransportThread extends Thread:

```
officialPoints.java shuttlesPoints.java Destination.java Driver.java UniversityTransportSystem.java TransportThread.java
package PROJECT;

public class TransportThread extends Thread{ 2 usages
    private String pointName; 2 usages

    public TransportThread(String pointName) { 4 usages
        this.pointName = pointName;
    }

    public void run() {
        System.out.println("Transport from " + pointName + " is now running ...");
        try {
            Thread.sleep( millis: 1000); // simulate time
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("Transport from " + pointName + " has completed the route.");
    }
}
```



Main Class (University Transport System)

```
officialPoints.java shuttlesPoints.java Destination.java Driver.java UniversityTransportSystem.java TransportThread.java

public class UniversityTransportSystem {
    public static void main(String[] args) {
        // Create drivers
        Driver d1 = new Driver("Suresh Patel", "98912345678", "100", "Mobile: 9876543210");
        Driver d2 = new Driver("Armen Berashvili", "98123456789", "100", "Mobile: 9876543210");
        Driver d3 = new Driver("Shubham", "9898989898", "100", "Mobile: 9876543210");
        Driver d4 = new Driver("Shivam", "98322999945", "100", "Mobile: 9876543210");

        // Create destinations
        Destination dest1 = new Destination("Bhopal-district", "Bhopal", "100");
        Destination dest2 = new Destination("Agra-district", "Agra", "100");
        Destination dest3 = new Destination("Raipur-district", "Raipur", "100");
        Destination dest4 = new Destination("Heldar-chock", "Delhi", "400");

        // Create points
        POINTS p1 = new officialPoints("Official point1");
        POINTS p2 = new officialPoints("Official point2");
        POINTS p3 = new shuttlesPoints("Shuttle point1");
        POINTS p4 = new shuttlesPoints("Shuttle point2");

        // Assign drivers
        p1.addDriver(d1);
        p2.addDriver(d2);
        p3.addDriver(d3);
        p4.addDriver(d4);

        // Assign destinations
        p1.addDestination(dest1);
        p2.addDestination(dest2);
        p3.addDestination(dest3);
        p4.addDestination(dest4);
    }

    System.out.println("*****");
    System.out.println("*****");
    System.out.println("UNIVERSITY TRANSPORT MANAGEMENT SYSTEM");
    System.out.println("*****");
    System.out.println("*****");
    System.out.println("the details of point 1:");
    System.out.println("*****");
}
```

Continue...

```
officialPoints.java shuttlesPoints.java Destination.java Driver.java UniversityTransportSystem.java TransportThread.java

public class UniversityTransportSystem {
    public static void main(String[] args) {
        // Assign drivers
        p1.addDriver(d1);
        p2.addDriver(d2);
        p3.addDriver(d3);
        p4.addDriver(d4);

        // Assign destinations
        p1.addDestination(dest1);
        p2.addDestination(dest2);
        p3.addDestination(dest3);
        p4.addDestination(dest4);
    }

    System.out.println("*****");
    System.out.println("*****");
    System.out.println("UNIVERSITY TRANSPORT MANAGEMENT SYSTEM");
    System.out.println("*****");
    System.out.println("*****");
    System.out.println("the details of point 1:");
    System.out.println("*****");
}
```

Main Class

PSVM()

- Create Drivers
- Create Destinations
- Create Points
- Print all details

- Assign Drivers
- Assign Destinations
- removeDriverById
- removeDestinationByName

Continue...

```
OfficePoints.java    shuttlePoints.java    Destination.java    Driver.java    UniversityTransportSystem.java    TransportThread.java
public class UniversityTransportSystem {
    public static void main(String[] args) {
        p2.printDetails();
        System.out.println("*****");
        System.out.println("the details of point 3:");
        System.out.println("*****");
        p3.printDetails();
        System.out.println("*****");
        System.out.println("the details of point 4:");
        System.out.println("*****");
        System.out.println();
        p4.printDetails();
    }
    // Simulate transport running with threads
    TransportThread t1 = new TransportThread(p1.pointName);
    TransportThread t2 = new TransportThread(p2.pointName);
    TransportThread t3 = new TransportThread(p3.pointName);
    TransportThread t4 = new TransportThread(p4.pointName);
    t1.start();
    t2.start();
    t3.start();
    t4.start();
}
```

..... "Output" Compiling.....
"Screen" Running.....

* * * University Transport Management System * * *

* * * Details of Point 1: * *

Drivers:

Driver Name: Shahzad Malik -- Driver Id: 101 --- Driver Contact: 03032894532

Destinations:

Assigned Destination: Qasimbad, Distance from University 3.0 km

* * * * * Details of Point 2: * *

Point 3:

Point 4: