Task No. 1: Write a program and create the objects of classes in class car to explain the concept of composition. Create several classes as engine, doors, capacity and wheel having their individual methods attributes. The object of these classes are created in a car class and they are set as public. The object of this car class is created in Main method and this with the help of this object we can call other classes as well and can use their functionalities and design UML class diagram.

Solution:

```
package Javalabtasks;
import java.util.*;

public class  main{

    public static void main(String[] args) {
        Car c1=new Car();

        c1.startCar();
        c1.openDoors();
        c1.printCapacity();
        c1.rotateWheels();
        c1.closeDoors();
        c1.stopCar();
    }
}
```

Car CLASS:

```
package Javalabtasks;
import java.util.*;
public class Car {
    public Engine engine;
    public Door door;
    public Capacity capacity;
    public Wheel wheel;

    public Car() {
        engine = new Engine();
        door =new Door(5);
        capacity =new Capacity(5);
        wheel =new Wheel(6);
    }
    public void startCar(){
        engine.start();
    }
    public void stopCar(){
        engine.stop();
    }
    public void openDoors(){
        door.open();
    }
    public void closeDoors(){
        door.close();
    }
    public void printCapacity(){
        System.out.println("Seating capacity : "+capacity.getCapacity());
    }
    public void rotateWheels(){
        wheel.rotate();
    }


}
```

Engine CLASS:

```
package Javalabtasks;
import java.util.*;
public class Engine {
    public void start(){
        System.out.println("Engine is Started");
    }
    public void stop(){
        System.out.println("Engine has Stopped");
    }


}
```

Wheel CLASS:

```
package Javalabtasks;

public class Wheel {
    private int wheel;

    public Wheel(int wheel) {
        this.wheel = wheel;
    }

    public int getWheel() {
        return wheel;
    }
    public void rotate(){
        System.out.println("Wheel is Rotating");
    }
}
```

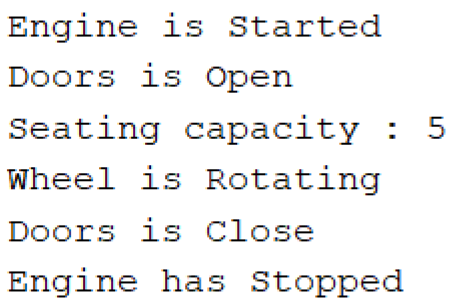
Door CLASS:

```
package Javalabtasks;

public class Door {
    private int no_doors;

    public Door(int no_doors) {
        this.no_doors = no_doors;
    }
    public int getNo_doors() {
        return no_doors;
    }

    public void open(){
        System.out.println("Doors is Open");
    }
    public void close(){
        System.out.println("Doors is Close");
    }

}
```

<u>Capacity CLASS:</u>

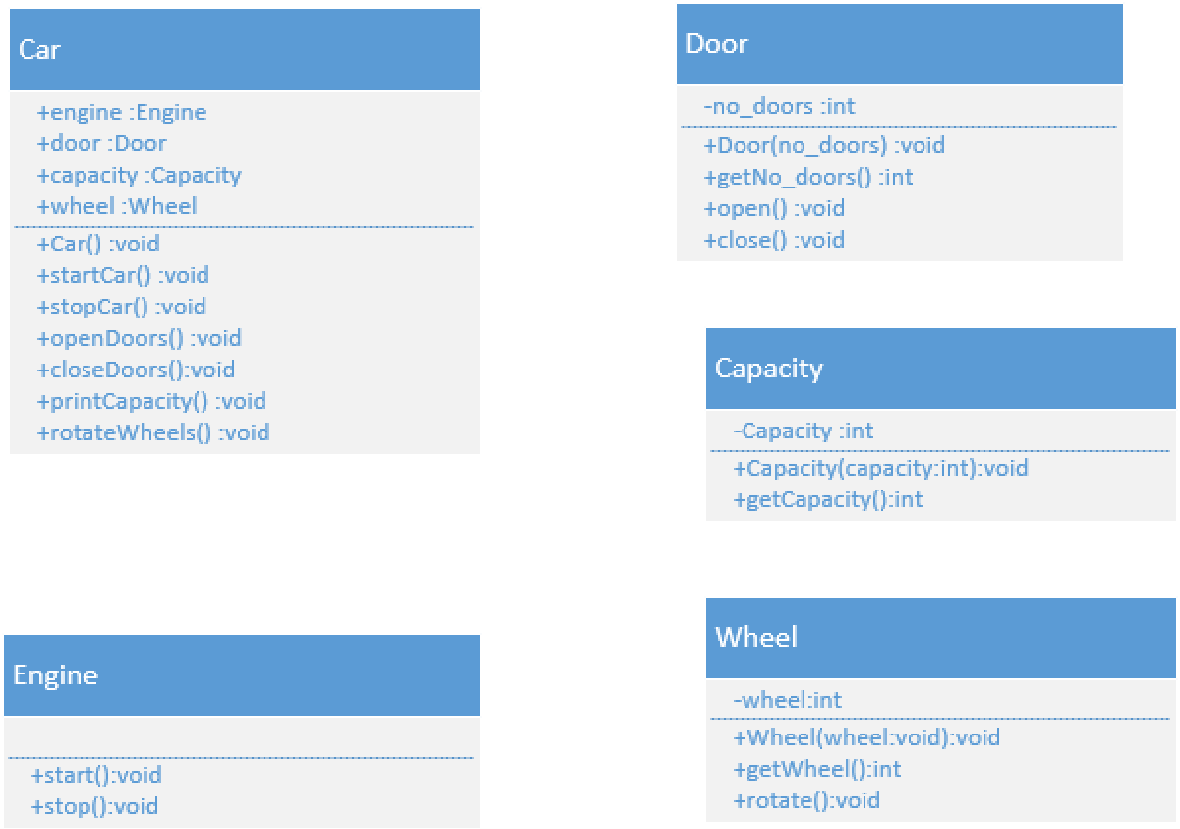
```
package Javalabtasks;

public class Capacity {
    private int capacity;

    public Capacity(int capacity) {
        this.capacity = capacity;
    }

    public int getCapacity() {
        return capacity;
    }

}
```

Output:



```
Engine is Started
Doors is Open
Seating capacity : 5
Wheel is Rotating
Doors is Close
Engine has Stopped
```

UML:



Car

+engine :Engine
+door :Door
+capacity :Capacity
+wheel :Wheel

+Car() :void
+startCar() :void
+stopCar() :void
+openDoors() :void
+closeDoors():void
+printCapacity() :void
+rotateWheels() :void

Door

-no_doors :int

+Door(no_doors) :void
+getNo_doors() :int
+open() :void
+close() :void

Capacity

-Capacity :int

+Capacity(capacity:int):void
+getCapacity():int

Engine

+start():void
+stop():void

Wheel

-wheel:int

+Wheel(wheel:void):void
+getWheel():int
+rotate():void

Task No. 2 : Write complete program for Flight's class, Time's class and Passenger's class with the concept of association and aggregation and design UML class diagram. Functions information also been given in the table below:

Method	Description
addPassenger(Passenger)	This method will add Passenger's object to vector passengerList.
printInfo()	This method will display all flight information namely ID (Flight number), destination, departure time, arrival time and number of passengers. For Example: Flight no : PK-303 Destination : Lahore Departure : 8:10 Arrival : 9:00 Number of passenger :2
getHour()	This method will return the value of attribute hour
getMinute()	This method will return the value of attribute minute

Solution:

```
package Javalabtasks;
import java.util.*;

public class  main{

    public static void main(String[] args) {
        Passenger p1=new Passenger("Omair",20);
        Passenger p2=new Passenger("Ali",30);
        Passenger p3=new Passenger("Usman",10);
        Passenger p4=new Passenger("Umer",25);
        Passenger p5=new Passenger("Muhammad",29);

        Flight f1=new Flight("PK-204","London",new Time(2,19),new Time(4,35));
        f1.addPassenger(p1);
        f1.addPassenger(p2);
        f1.addPassenger(p3);
        f1.addPassenger(p4);
        f1.addPassenger(p5);
         f1.printInfo();
    }
}
```

Flight CLASS:

```
package Javalabtasks;
import java.util.*;
public class Flight {
    String flight_no;
    String destination;
    Time departuretime;
    Time arrivaltime ;
    private List<Passenger> passengerList=new ArrayList<Passenger>();

    Flight(String flight_no, String destination, Time departuretime, Time arrivaltime)
{
        this.flight_no = flight_no;
        this.destination = destination;
        this.departuretime = departuretime;
        this.arrivaltime = arrivaltime;

    }
    public void addPassenger(Passenger p){
        this.passengerList.add(p);
    }

    public void printInfo(){
        System.out.println("Flight : "+flight_no);
        System.out.println("Destination : "+destination);
        System.out.println("Departure : "+timeconvert(departuretime));
        System.out.println("Arrival : "+timeconvert(arrivaltime));
        System.out.println("Number of passengers : "+passengerList.size());
    }

    public String timeconvert(Time t){
        return String.format("%d:%d",t.hours,t.minute);
    }

}
```

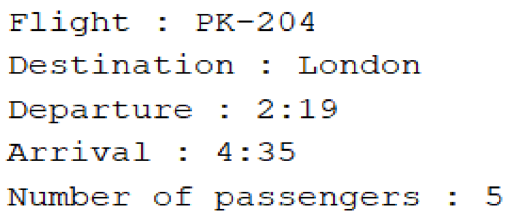
Time CLASS:

```
package Javalabtasks;

public class Time {
    int hours;
    int minute;

    public Time() {
    }

    Time(int hours, int minute) {
        this.hours = hours;
        this.minute = minute;
    }
    public int getHours() {
        return hours;
    }

    public int getMinute() {
        return minute;
    }

}
```

Passenger CLASS:

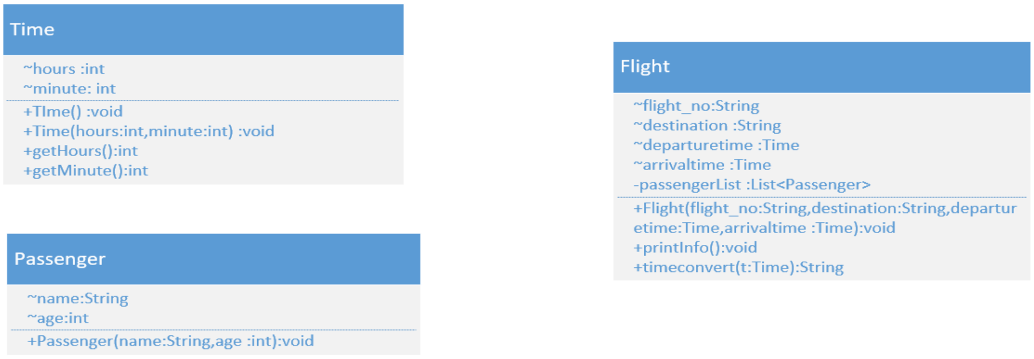
```
package Javalabtasks;
import java.util.*;
public class Passenger {
    String name;
    int age;

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


}
```

Output:



```
Flight : PK-204
Destination : London
Departure : 2:19
Arrival : 4:35
Number of passengers : 5
```

UML:


Time
~hours :int
~minute: int
+TIme() :void
+Time(hours:int,minute:int) :void
+getHours():int
+getMinute():int
Passenger
~name:String
~age:int
+Passenger(name:String,age :int):void
Flight
~flight_no:String
~destination :String
~departuretime :Time
~arrivaltime :Time
-passengerList :List<Passenger>
+Flight(flight_no:String,destination:String,departuretime:Time,arrivaltime :Time):void
+printInfo():void
+timeconvert(t:Time):String

Task No. 3: A company manages many stores. Each Store contains many Products. Implement Product, Store and Company classes using association and aggregation concepts and design UML class diagram.

Solution:

```
package Javalabtasks;
import java.util.*;

public class  main{

    public static void main(String[] args) {
        Product p1=new Product("Rice",200,80);
        Product p2=new Product("Sugar",900,90);
        Product p3=new Product("Shampoo",800,50);
        Product p4=new Product("Bread",500,40);
        Product p5=new Product("Soap",500,10);
        Product p6=new Product("Butter",1000,30);

        List<Product> branch1_product=new ArrayList<Product>();
        List<Product> branch2_product=new ArrayList<Product>();
        branch1_product.add(p1);
        branch1_product.add(p2);
        branch1_product.add(p5);
        branch2_product.add(p4);
        branch2_product.add(p3);
        branch2_product.add(p6);

        Store store1=new Store("Karachi Branch",branch1_product);
        Store store2=new Store("Punjab Branch",branch2_product);
        List<Store> Store=new ArrayList<Store>();
        Store.add(store1);
        Store.add(store2);
        Company C1=new Company("ABC",Store);
        C1.showallitems();
    }
}
```

<u>Company CLASS:</u>

```
package Javalabtasks;
import java.util.*;
public class Company {
    String name;
    private List<Store> store;

    public Company(String name, List<Store> store) {
        this.name = name;
        this.store = store;
    }

public void showallitems(){
        List<Product> product;
        for(Store s : store){
            System.out.println("\n"+s.name);
            product=s.getProducts();
            for(Product p : product){
                System.out.println(p.name);
            }
        }
    }
}
```

Store CLASS:

```
package Javalabtasks;
import java.util.*;
public class Store {
    String name;
    private List<Product> products;

    public Store(String name, List<Product> products) {
        this.name = name;
        this.products = products;
    }

    public List<Product> getProducts() {
        return products;
    }

}
```

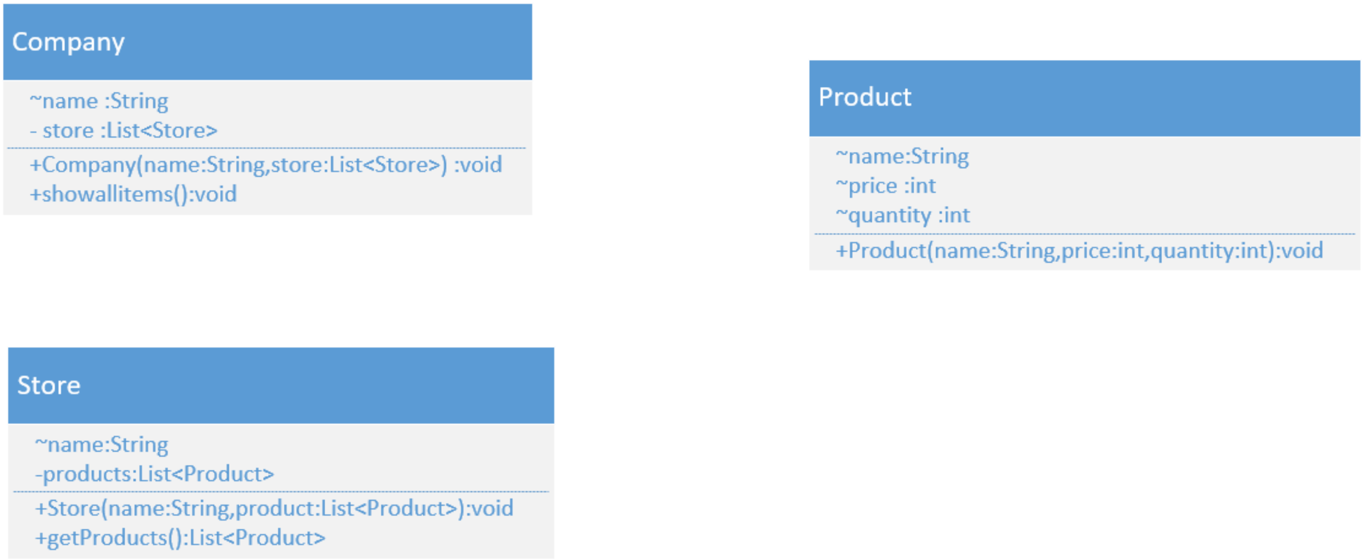
Products CLASS:

```
package Javalabtasks;

public class Product {
    String name;
    int price;
    int quantity;

    Product(String name, int price, int quantity) {
        this.name = name;
        this.price = price;
        this.quantity = quantity;
    }
}
```

Output:

```
Karachi Branch
Rice
Sugar
Soap

Punjab Branch
Bread
Shampoo
Butter
```

UML:



Company

~name :String
- store :List<Store>

+Company(name:String,store:List<Store>) :void
+showallitems():void

Product

~name:String
~price :int
~quantity :int

+Product(name:String,price:int,quantity:int):void

Store

~name:String
-products:List<Product>

+Store(name:String,product:List<Product>):void
+getProducts():List<Product>