The purpose of this lab is to get hands on experience of inheritance in OO programming.

Create a new package: `com.lab.week3`

Part 1 – write a Car class

A `Car` class has four attributes :
brand (String), year (int), speed (int), doors (int);
And two methods:

- accelerate – which takes in a parameter `increase` (int) and adds the increase to the speed; It then prints out "The speed is now X ";
- brake – which takes in a `decrease` (int) and reduces the speed by this decrease amount. Then prints out "The speed is now X ";

  Add a toString() method that returns a String containing the car object characteristics: "This car is a volvo… etc

Create the Car class (Car.java).
Make sure the attributes are encapsulated.
Add constructor that sets all the Car attributes.

Create a separate Control class with a `main` method in it.
Instantiate a car object.
Call the accelerate and brake methods for the object to make sure they work.

Part 2 –Motorcycle class

A Motorcycle class has four attributes:
brand (String), year (int), speed (int), twoSeater(boolean);
And two methods:

- accelerate – which takes in an `increase` (int) and add the increase to the speed;
- brake – which takes in an `decrease` (int) and reduces the speed by this decrease amount.

Create the Motorcycle class (Motorcycle.java).

Make sure the attributes are encapsulated.

Add constructor that sets all the Motorcar attributes.

Add a toString() method that returns a String containing the Motorcycle object characteristics: “this motorbike is a Suzuki, with … etc

From your Control class in part 1, instantiate a motorcycle object, with different attribute values.
Call the accelerate and brake methods for the motorcycle object to make sure they work.

Part 3 – Implement inheritance (super class and sub classes)

Note: keep the original car/motorcycle code as comments so the lab grader can see it.

Create a third class Vehicle.

This will be the superclass for Car and Motorcycle, to remove the redundancy of repeating code in car and motorcycle classes.

Edit the Car and Motorcycle classes so that they are inheriting from Vehicle :
Move the common attributes (3) and methods (2) from Car and Motorcycle into the Vehicle class. Adjust the constructors and the toString() methods of the Car and Motorcycle classes to use the super class.

Re-run your Control class to make sure the Car and Motorcycle objects creation code still works okay (now that you have edited the code to use inheritance.
Also create a vehicle object in your Control class.

Part 4 – Add class Electric Car

Create a fourth class ElectricCar. This is a type of Car (i.e. sub class of Car).
It has an additional attribute: range (int).
Implement this class, add the constructor.
Instantiate an object of it from the Control class to test it works.

Part 5 – Overriding methods

Add a honk() method to *each* of the four classes. The method should just print out a sound to the console.
A vehicle honk should be “honk honk!”;
A motorbike honk should be “Vroooom!”
A car honk should be “beep beep!”;
An electric car should be “prrrrrrr”;

From the main method, call the honk() method for each of your vehicle, car, motorcycle and electric objects.