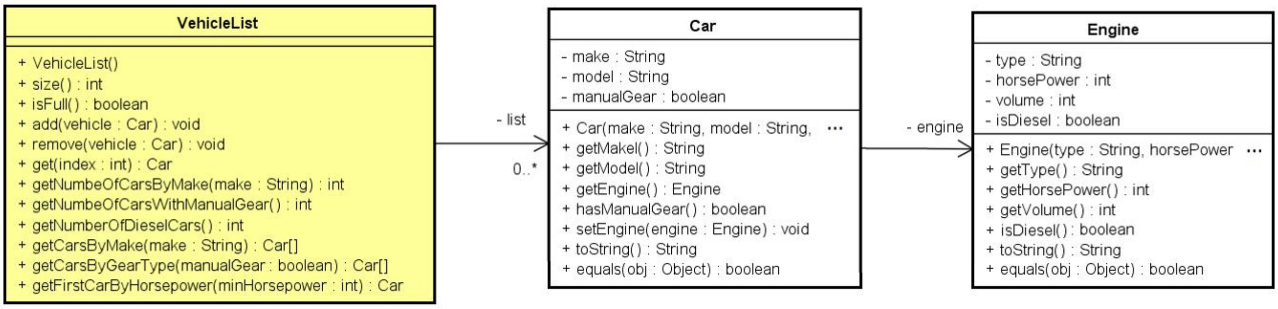
Exercise: VehicleList (a list of Car's), version 2

Create a module named VehicleList_v2 with VehicleList, Car and Engine into this module. Copy Car and Engine from a previous exercise or use from the end of this document.

Study the class diagram below and note that the VehicleList has is no size variable and the constructor takes no capacity, I.e. there is no limit.


VehicleList
+ VehicleList()
+ size() : int
+ isFull() : boolean
+ add(vehicle : Car) : void
+ remove(vehicle : Car) : void
+ get(index : int) : Car
+ getNumbeOfCarsByMake(make : String) : int
+ getNumbeOfCarsWithManualGear() : int
+ getNumberOfDieselCars() : int
+ getCarsByMake(make : String) : Car[]
+ getCarsByGearType(manualGear : boolean) : Car[]
+ getFirstCarByHorsepower(minHorsepower : int) : Car
- list
0..*
Car
- make : String
- model : String
- manualGear : boolean
+ Car(make : String, model : String, …
+ getMakel() : String
+ getModel() : String
+ getEngine() : Engine
+ hasManualGear() : boolean
+ setEngine(engine : Engine) : void
+ toString() : String
+ equals(obj : Object) : boolean
- engine
Engine
- type : String
- horsePower : int
- volume : int
- isDiesel : boolean
+ Engine(type : String, horsePower …
+ getType() : String
+ getHorsePower() : int
+ getVolume() : int
+ isDiesel() : boolean
+ toString() : String
+ equals(obj : Object) : boolean

Do not change any of the classes Car and Engine.

Notes to class VehicleList:

a) Use an ArrayList of Car objects as the only instance variable, i.e. of type ArrayList<Car>.
b) Delete the int instance variable, i.e. only one instance variable.
c) The constructor takes no arguments. In the body of the constructor, initialize the ArrayList.
d) The method size returning the actual size of the list, i.e. the size of the ArrayList.
e) Method isFull always return false (the list is never full)
f) A method add, adding a car to end of the list.
g) A method remove, removing the Car specified. Note that you need an equals method in class Car (why?)
h) A method get returning the element at the index specified.
i) Modify the remaining 6 methods to use an ArrayList instead of an array.

Run the test (the class with the main method) again.

Classes Engine and Car on the next pages…

Class Engine

```
public class Engine
{
  private String type;
  private int horsePower;
  private int volume;
  private boolean isDiesel;

  public Engine(String type, int horsePower, int volume, boolean isDiesel)
  {
    if (type == null)
    {
      type = "";
    }
    this.type = type;
    this.horsePower = horsePower;
    this.volume = volume;
    this.isDiesel = isDiesel;
  }

  public String getType()
  {
    return type;
  }

  public int getHorsePower()
  {
    return horsePower;
  }

  public int getVolume()
  {
    return volume;
  }

  public boolean isDiesel()
  {
    return isDiesel;
  }

  @Override public String toString()
  {
    String s = type + ", " + volume + " cm3, " + horsePower + " hp, ";
    if (!isDiesel)
    {
      s += "not a";
    }
    s += " diesel";
    return s;
  }

  @Override public boolean equals(Object obj)
  {
    if (!(obj instanceof Engine))
    {
      return false;
    }
    Engine other = (Engine) obj;
    return horsePower == other.horsePower && volume == other.volume
        && isDiesel == other.isDiesel && type.equals(other.type);
  }
}
```

Class Car

```
public class Car
{
  private String make;
  private String model;
  private boolean manualGear;
  private Engine engine;

  public Car(String make, String model, Engine engine, boolean manualGear)
  {
    if (make == null)
    {
      make = "NO make";
    }
    if (model == null)
    {
      model = "No model";
    }
    this.make = make;
    this.model = model;
    this.manualGear = manualGear;
    setEngine(engine);
  }

  public String getMake()
  {
    return make;
  }

  public String getModel()
  {
    return model;
  }

  public boolean hasManualGear()
  {
    return manualGear;
  }

  public Engine getEngine()
  {
    return engine;
  }

  public void setEngine(Engine engine)
  {
    this.engine = engine;
  }

  @Override public String toString()
  {
    String s = make + " " + model + ", ";
    if (manualGear) {s += " manual gear";    }
    else {s += "automatic gear";}

    s += ", " + engine;
    return s;
  }

  @Override public boolean equals(Object obj)
  {
    if (!(obj instanceof Car))
    {
      return false;
    }
    Car other = (Car) obj;
    return make.equals(other.make) && model.equals(other.model) && engine
        .equals(other.engine) && manualGear == other.manualGear;
  }
}
```