

Project:

Now that you know about objects. You always think of real life objects:

You have seen car Dealership in street. What if you are ask to write java Program to do design Dealership class.

Think what dealership has, Cars, SalePersons, Manager and more objects.

We are only concern for Car, SalesPerson, Manager and Dealership.

Write a class for Car and have the following private fields or members:

make , price, year and vinNumber.

Write a class for SalesPerson and have the following private fields or members

First name, last name, Bonus, address, date of hiring.

Write a class for Manager and have the following private fields or members:

First name, last name, address, date of hiring.

Write a class for Dealership and have the following private fields or members:

Array of SalesPerson, Array of Car, and one Manager.

Provide constructor and methods for all of above classes. You should have setters and getters, and method toString() that returns states of objects that is what it knows about object (example: First name, last...).

Dealership class should have methods addSalesPerson, removeSalesPerson, carSold , removeCar, addCar, and toString().

toString() should print state of Dealership: manager , SalesPersons, Cars, and number of car sold.

The Following show possible layout of Dearship class:

```
/**
 *
 * @author adehkhoda
 */
public class Dealership {
    private ArrayList <SalesPerson> sp = new ArrayList <SalesPerson> ();
    private ArrayList <Car> cars= new ArrayList <Car> ();
    private Manager mg;

    public Dealership()
    {
    }

    public void addCar(Car another)
    {
    }

    public void addSalesPerson(SalesPerson s)
    {
    }
}
```

```

public void removeSalesPerson(SalesPerson s)
{
}

public void carSold (SalesPerson s)
{
}

public void removeCar( Car aCar)
{
}

```

```

public SalesPerson  getPersonOfTheMonth()
{
}

public String toString()
{
}

```

Note: that why we use ArrayList class because we have no clue of how many cars, or

SalesPerson Dealership has. ArrayList has very useful methods that make life easy.

Dealership must only created when associate a Manager with it no other way.

Method getPersonOfTheMonth()

Should return SalesPerson that makes highest bonus. Feel free to add any other

Methods that you think is good to have for Dearship.

Your Diver class that you must use:

```
public class TestDealership
{
    public static void main(String [] args)
    {
        // Create one manager
        // Create at least 5 SalesPerson
        // Create at Least 10 Cars
        // Create one Dealership and associates manager with it.
        // Use Dealership to add SalesPerson and Cars
        // now test your program for all the method that was given in Dealership
        // Example carSold, reamoveCar, reamoveSalesPerson....
        // Note if the Car was sold you should not sell again

        System.exit(0);
    }
}
```