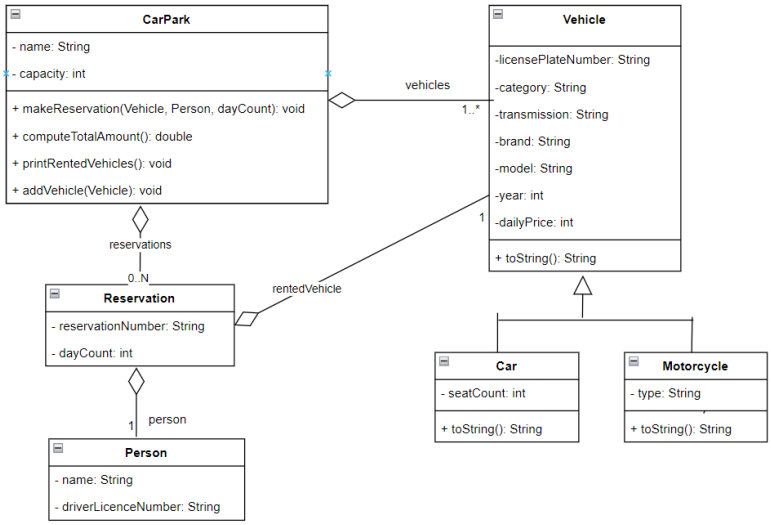
Homework-3 / April 6, 2023

1. **The Deadline for this homework is Wednesday, 14 April 11:59 PM**
2. **Late submission will not be accepted.**
3. **Homework should be completed individually, copying the work of others is not permitted.**
4. **The penalty for teamwork or copying source code from others is -100.**
5. **You must upload your project into LMS as an archive file. Sending via email will not be accepted.**

Homework Details:

1. A simple Car Rental/Park system's UML diagram is provided. Implement corresponding Java classes.
2. CarPark's constructor should take park's capacity and creates the *vehicles* array.
3. Add all listed fields and methods in your classes. Apply encapsulation principle. Add getter/setter methods and all required contructors.



4. Use simple arrays. Don't use any collection class from java.util package:like Array, ArrayList, Collection, Set, HashSet etc..
5. Implement toString methods int Vehicle, Car and Motorcycle. Don't repeat code, use method overriding feature
6. Implement following methods in the CarPark class:

Method name	**Implementation Details**
addVehicle **Input params:** Vehicle **return:** boolean (true/false)	1. Find the vehicle count in the *vehicles* array(don't count empty elements) 2. If parked vehicle count < carPark's capacity then add Vehicle into the ~~*parkedVehicles*~~ *vehicles* array and return true otherwise don't add vehicle into the array and return false
makeReservation **Input params:** Vehicle, Person, dayCount **return:** void	1. generate a random 4 digit reservation number 2. Create a reservation Object 3. Add newly created reservation object into the CarPark's *reservations* array
computeTotalAmount **Input params**: - **return**: double	1. Loop *reservations* array 2. Calculate *dailyPrice*dayCount* for each reservation. 3. Add previously calculated amount to totalAmount 4. Return totalAmount
printRentedVehicles **params**: -	1. create a loop for reservations array . 2. print following reservation data on the screen: • reservation number

return: void	• person name • day count • vehicle brand • vehicle model • seat count (if car) • type (if motorcycle)

SAMPLE TEST CODE

```
CarPark carPark = new CarPark("Star Park", 5);
Car car1 =
Car car2 =
Car car3 =
Car car4 =
Motorcycle motor1 =
Motorcycle motor2 =

Person person1=…
Person person2=…
Person person3=…

carPark.addVehicle(car1);
carPark.addVehicle (car2));
carPark.addVehicle (car3));
carPark.addVehicle (motor1)
carPark.addVehicle (motor2);

carPark.makeReservation(car1, person1, 5);
carPark.makeReservation(car2, person2, 10);
carPark.makeReservation(motor1, person3, 4);

System.out.println("Total Amount="+carPark.computeTotalAmount());
System.out.println("--------- RENTED VEHICLES ---------");
carPark.printRentedVehicles();
```

This part is hidden. Because you can initialize your objects in different ways.
I don't want you to use exactly my way

SAMPLE EXECUTION

```
Total Amount=3400.0
--------- RENTED VEHICLES ---------
Res:[6413,5 days], Driver:[Esma Meral], Vehicle:[Honda, Civic, 5 seats, Car]
Res:[3067,10 days], Driver:[Sema Demir], Vehicle:[Honda, Jazz, 5 seats, Car]
Res:[1824,4 days], Driver:[Ahmet Karaca], Vehicle:[Honda, CBF 150, Scooter]
```

Sample CARs

#	License Plate Nbr	category	transmission	brand	model	year	Seat Count
1	34 EYY 62	Medium	Automatic	Honda	Civic	2020	5
2	34 H 6287	Small	Manuel	Honda	Jazz	2019	5
3	06 AB 87	Medium	Automatic	Toyota	Corolla	2021	5
4	16 CK 28	Large	Automatic	Peugeot	301	2022	5

Sample MOTORCYCLEs

#	License Plate Number	category	transmission	brand	model	year	type
1	34 KK 71	Standard	Manual	Honda	CBF 150	2018	Scooter
2	34 ABC 51	Adventure	Manual	BMW	R120GS ADV	2022	Motorcycle