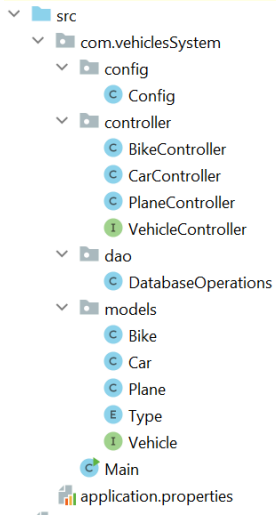
Use case - Vehicles System

We have a company that need a system to sell a vehicles like (car , plane , motorcycle)
Each one of them will be stored in the database so , we will have operations like saveVehicle.
The configurations of the database like url , username and password will be saved into a properties file.

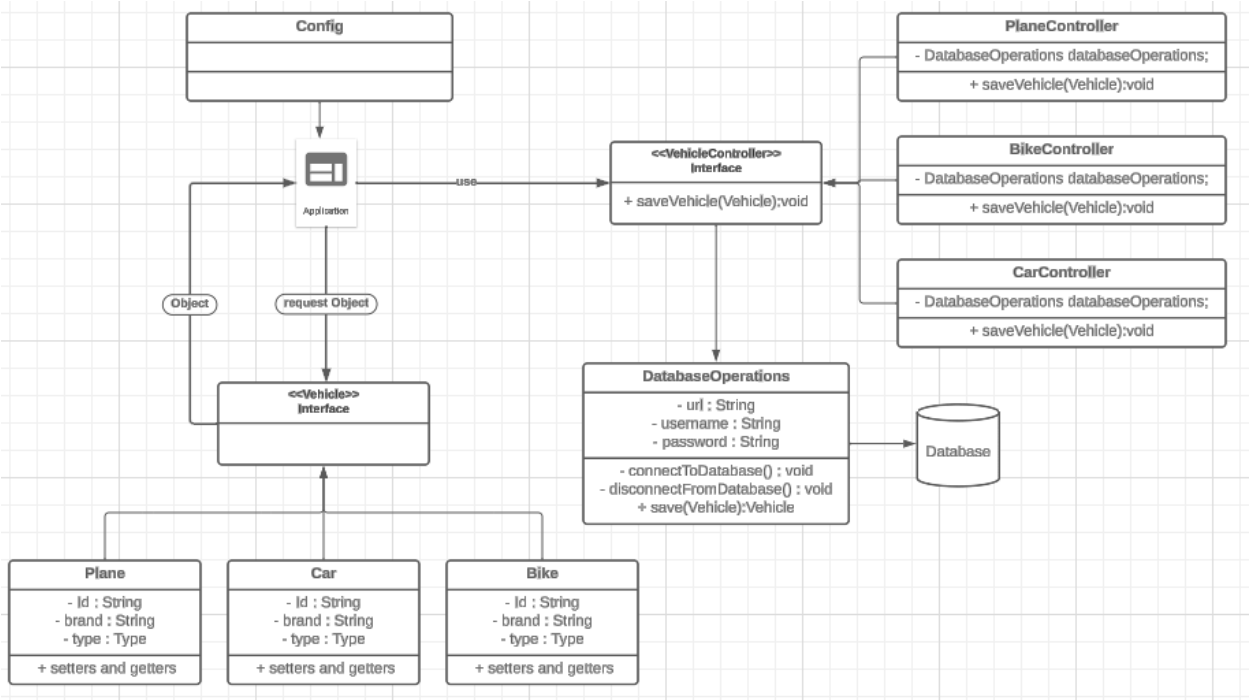
Requirements:

1. Make a class that will control the database operations and have **singleton** scope (as we only need one bean through the app)
2. Read the database properties from properties file and link them to fields in the database operations class.
3. Define your models (car , plane , Bike) classes
4. Define your VehicleDTO that will handle the operation of insertion into database.I
5. Inject the database operations bean into these models using constructor injection for car and setter injection for plane and field injection for Bike.
6. Make all the configurations inside the app in the java config class (no XML)
7. Make init method inside the database operations class that will connect to the database after the container created
8. Make destroy method inside the database operations class that will close the connection from the database after the container closed
9. Use MySQL to store your data and JDBC **[saveVehicle() method]**
10. Use System.out.print methods for debugging purposes (connection established , closed …etc)
11. Use @Getter , @Setter , @ToString from lombok instead of normal setters and getters. See the hints
12. Add methods save , delete , update , searchById (Optional)
13. Define the below tables attached in the hints.

Hint: please follow this structure



See lombok tutorial here and add this JAR into your project libraries and also install lombok plugin in intellij file->settings->plugins->search for lombok
This is the class diagram



DAO Pattern tutorial : https://www.baeldung.com/java-dao-pattern

Vehicle Table

ID	brand	type
1	BMW	Car
2	Honda	Bike

Orders Table

ID	Order Date	Price	vehicle_id
1	18/9/2020	1400$	1
2	18/10/2021	1500$	2
3	21/11/2021	1500$	2