1. Mengubah method view dealer agar menampilkan daftar mobil beserta detail terurut berdasarkan harga mobil.
   - File DealerModel.java untuk mengimplementasikan compareTo

   ```java
   class SortByPrice implements Comparator<CarModel>
   {
       // Used for sorting in ascending order of
       // roll number
       public int compare(CarModel a, CarModel b)
       {
           return Math.toIntExact(a.getPrice() - b.getPrice());
       }
   }
   ```

   ```java
   public List<CarModel> getListCar() {
       Collections.sort(listCar, new SortByPrice());
       return listCar;
   }
   ```

   - File DeaerController.java

   ```java
   @RequestMapping(value="/dealer/view", method=RequestMethod.GET)
   private String viewDealer(@RequestParam ("dealerId") long id, Model model) {
       if(dealerService.getDealerDetailById(id).isPresent()) {

           DealerModel dealer = dealerService.getDealerDetailById(id).get();
           model.addAttribute("dealer", dealer);
           /*model.addAttribute("car", dealer.getListCar());*/
           System.out.println(dealer.getListCar().size());
           List<CarModel> car = dealer.getListCar();
           model.addAttribute("car", car);
           return "viewDealer";
       }
       else {
           model.addAttribute("exception", "Data tidak ditemukan");
           return "exception";
       }

   }
   ```

   - File ViewDealer.html

```html
<body>
<h1> Detail Dealer</h1>
    <h2 th:text=" 'Nama      : ' + ${dealer.nama}"></h2>
    <h2 th:text=" 'Alamat    : '+ ${dealer.alamat}"></h2>
    <h2 th:text=" 'Telepon   : '+ ${dealer.noTelp}"></h2>
    <a th:href="@{'/car/add/'+ ${dealer.id}}">Tambah Mobil</a>

    <a th:href="@{'/dealer/update/'+ ${dealer.id}}">Update Dealer</a>

    <h3> Daftar Mobil</h3>
    <table class="table table-striped">
        <tr>
        <th> Id</th>
        <th> Brand</th>
        <th> Type</th>
        <th> Price</th>
        <th> Amount</th>
        <th> Option</th>
        </tr>
        <tr th:each="car : ${car}">
            <td th:text="${car.id}"></td>
            <td th:text="${car.brand}"></td>
            <td th:text="${car.type}"></td>
            <td th:text="${car.price}"></td>
            <td th:text="${car.amount}"></td>
            <td><form th:action="@{'/car/delete/'+ ${car.id}}">
            <button type="submit" >Hapus</button></form>
```

2.1 Delete untuk menghapus Mobil
-    File Service mobil

```java
package com.apap.tutorial4.service;
import java.util.Optional;

public interface CarService {
    void addCar(CarModel car);

    Optional<CarModel> getCarDetailById(Long id);

    void deleteCar(CarModel car);

    void updateCar(long id, CarModel car);


}
```

-    File Service mobil impl

```java
@Override
public void deleteCar(CarModel car) {
    // TODO Auto-generated method stub
    carDb.delete(car);
}
```

- File mobil controller

```java
@RequestMapping(value="/car/delete/{id}", method = RequestMethod.GET)
private String deleteCar(@PathVariable String id, Model model){
    long carId = Long.parseLong(id);
    CarModel car = carService.getCarDetailById(carId).get();
    carService.deleteCar(car);
    return "delete";
}
```

2.2. Untuk menghapus dealer
- File DealerService

```java
public interface DealerService {
    Optional<DealerModel> getDealerDetailById(Long id);
    void addDealer(DealerModel dealer);
    void deleteDealer(DealerModel dealer);
    void updateDealer(DealerModel dealer, long id);
    List<DealerModel> viewAll();

}
```

- File Dealer Service impl

```java
public void deleteDealer(DealerModel dealer) {
    dealerDb.delete(dealer);
}
```

- File DealerCOntroller

```java
//Method untuk menghapus dealer
@RequestMapping(value="/dealer/delete", method=RequestMethod.GET)
private String deleteDealer(@RequestParam ("dealerId") long id, Model model){
    DealerModel dealer = dealerService.getDealerDetailById(id).get();
    dealerService.deleteDealer(dealer);
    return "delete";

}
```

3.1 Update untuk dealer
- File Dealer Service

```java
public interface DealerService {
    Optional<DealerModel> getDealerDetailById(Long id);
    void addDealer(DealerModel dealer);
    void deleteDealer(DealerModel dealer);
    void updateDealer(DealerModel dealer, long id);
    List<DealerModel> viewAll();

}
```

- File Dealer service impl

```
@Override
public void updateDealer(DealerModel dealer, long id) {
    // TODO Auto-generated method stub
    DealerModel old = dealerDb.getOne(id);
    old.setAlamat(dealer.getAlamat());
    old.setNama(dealer.getNama());
    old.setNoTelp(dealer.getNoTelp());
    dealerDb.save(old);

}
```

- File DealerController

```
@RequestMapping(value="/dealer/update/{id}", method= RequestMethod.GET)
private String updateDealer(@PathVariable String id, Model model) {
    long dealerId = Long.parseLong(id);
    DealerModel dealer = dealerService.getDealerDetailById(dealerId).get();
    model.addAttribute("dealer", dealer);
    return "updateDealer";
}

@RequestMapping(value="/dealer/update/{id}", method=RequestMethod.POST)
private String updateDealer(@PathVariable (value="id") long id, @ModelAttribute DealerModel dealer) {
    dealerService.updateDealer(dealer, id);
    return "update";
}
```

3.2 Untuk mengupdate Car

- File CarService

```
package com.apap.tutorial4.service;
import java.util.Optional;

public interface CarService {
    void addCar(CarModel car);

    Optional<CarModel> getCarDetailById(Long id);

    void deleteCar(CarModel car);

    void updateCar(long id, CarModel car);


}
```

- File CarServiceImp

```
@Override
public void updateCar(long id, CarModel car) {
    CarModel baru = carDb.getOne(id);
    baru.setBrand(car.getBrand());
    baru.setAmount(car.getAmount());
    baru.setPrice(car.getPrice());
    baru.setType(car.getType());
    carDb.save(baru);


}
```

- File CarController

```java
@RequestMapping(value="/car/update/{id}", method= RequestMethod.GET)
private String updateCar(@PathVariable String id, Model model) {
    long carId = Long.parseLong(id);
    CarModel car = carService.getCarDetailById(carId).get();
    model.addAttribute("car",car);
    return "updateCar";
}

@RequestMapping(value="/car/update/{id}", method=RequestMethod.POST)
private String updateCar(@PathVariable (value="id") long id, @ModelAttribute CarModel car) {
    carService.updateCar(id, car);
    return "update";
}
```

4. Melihat ViewAll dealer
   - File DealerService

```java
public interface DealerService {
    Optional<DealerModel> getDealerDetailById(Long id);
    void addDealer(DealerModel dealer);
    void deleteDealer(DealerModel dealer);
    void updateDealer(DealerModel dealer, long id);
    List<DealerModel> viewAll();

}
```

   - File DealerServiceImpl

```java
@Override
public List<DealerModel> viewAll() {
    // TODO Auto-generated method stub
    return dealerDb.findAll();

}
```

   - File DealerController

```java
@RequestMapping("/viewAll")
private String viewAll(Model model) {
    List<DealerModel> dealerAll =  dealerService.viewAll();
    model.addAttribute("listDealer", dealerAll);
    return "viewAll";
}
```