



DB-JAVA 연동 중고차 검색

조현태

CONTENTS



01 DB, JAVA

- DB(Oracle)
- JAVA



02 실행

- 실행화면



03 마무리

- 느낀점



Part 1.

DB, JAVA



2.1 DB

Oracle

```
create table H_car(  
  origin varchar2(10) not null,  
  make varchar2(50) not null,  
  carnum varchar2(50) primary key,  
  name varchar2(50) not null,  
  accident varchar2(10) not null,  
  cartype varchar2(30) not null,  
  fuel number(4),  
  p_year date,  
  mileage number(10),  
  price int,  
  color varchar(30)  
);
```

테이블 생성

데이터 삽입

```
insert into H_car values ('국산','현대','327어9577','아반떼(CN7)','무사고','준중형','가솔린','2020-05-01','19445','25635000','화이트');  
insert into H_car values ('국산','쉐보레','264루9845','더뉴스파크프리미어','단순교환','경차','가솔린','2020-07-01','3240','10579500','진주색');  
insert into H_car values ('국산','기아','223구4695','K8','무사고','대형차','가솔린3.5','2021-04-01','4819','52935000','검정색');  
insert into H_car values ('국산','현대','167어7240','투싼(4세대)','무사고','준중형SUV','하이브리드','2020-11-01','9306','42035000','흰색');  
insert into H_car values ('국산','현대','39러4915','팰리세이드 2WD','무사고','준대형SUV','디젤','2019-01-01','11495','43335000','검정색');  
insert into H_car values ('해외','BMW','10더9454','520dxDriveM스포츠플러스','무사고','중형','디젤','2017-12-01','51947','47645000','쥐색');  
insert into H_car values ('해외','벤츠','04머0135','S-클래스W222마이바흐S500','무사고','대형','가솔린','2016-02-01','134750','91885000','검정색');  
insert into H_car values ('해외','포르쉐','03수6338','718박스터S2.5','무사고','스포츠카','가솔린','2017-09-01','35618','101705000','노란색');  
insert into H_car values ('해외','테슬라','64누4321','모델3롱레인지','무사고','중형','전기','2020-09-01','24148','54015000','청색');
```

2.2 Java

DAO .1



DAO(Data Access Object)는 데이터 베이스의 **data**에 접근하기 위한 객체 데이터베이스 접근을 하기 위한 로직과 비즈니스 로직을 분리하기 위해 사용

```
public CarDAO() {  
    dtos = new ArrayList<CarVO>();  
  
    try {  
        String user="system";  
        String pw="1234";  
        String url="jdbc:oracle:thin:@localhost:1521:XE";  
  
        Class.forName("oracle.jdbc.driver.OracleDriver");  
        con=DriverManager.getConnection(url,user,pw);  
        st=con.createStatement();  
  
    }catch(Exception e) {  
        System.out.println("데이터베이스 연결 오류:"+e.getMessage());  
    }  
}
```

DB-Java 연결

2.2 Java

DAO . 2

```
public ArrayList<CarVO> getAllCar() {
    String SQL="select * from h_Car";
    try {
        rs=st.executeQuery(SQL);
    }

public ArrayList<CarVO> SearchCar(String content, String input_search) {
    String SQL="select * from h_car where "+content+" like ?";
    String Like_search="%" +input_search+"%";
    PreparedStatement pstmt=null;

public ArrayList<CarVO> SearchMileage(int input_search8, int input_mileage) {
    String SQL="select * from h_car where mileage BETWEEN ? AND ?";

public ArrayList<CarVO> SearchPrice(int input_search, int input_price) {
    String SQL="select * from h_car where price BETWEEN ? AND ?";

public ArrayList<CarVO> InsertCar(String origin, String make, String carnum
    String name, String accident, String cartype, String fuel,
    String p_year, int mileage, int price, String color) {

    String SQL="Insert into h_car values(?,?,?,?,?,?,?,?,?,?,?)";

public ArrayList<CarVO> UpdateCar(String content, String input_content, String input_carnum
    String SQL="update h_car set "+content+"=? where carnum=? ";

public ArrayList<CarVO> DeleteCar(String carnum) {
    String SQL="delete from h_car where carnum='"+carnum+"'";

    try {
        rs=st.executeQuery(SQL);
    }
```

DB 모두 조회

DB 조건 조회

DB 주행거리 조건조회

DB 가격 조건조회

DB 추가

DB 수정

DB 삭제

2.2 Java

DTO



DTO(Data Transfer Object)는 계층 간 데이터 교환을 하기 위해 사용하는 객체로 로직을 가지지 않고 getter, setter 만 가집니다.

```
public CarVO(String origin, String make, String carnum, String name, String accident,
String cartype,String fuel,String p_year,int mileage,int price,String color) {

    this.origin=origin;
    this.make=make;
    this.carnum=carnum;
    this.name=name;
    this.accident=accident;
    this.cartype=cartype;
    this.fuel=fuel;
    this.p_year=p_year;
    this.mileage=mileage;
    this.price=price;
    this.color=color;
}
```

```
public void setOrigin(String origin) {
    this.origin=origin;
}
public String getOrigin() {
    return origin;
}
public void setCarnum(String carnum) {
    this.carnum=carnum;
}
public String getCarnum() {
    return carnum;
}
public void setMake(String make) {
    this.make=make;
}
public String getMake() {
    return make;
}
```



2.2 Java

Controller . 1



Controller는 사용자에게 받은 요청을 수행할 로직을 제어하는 객체입니다.

```
public static void main(String[] args) {
    ArrayList<CarVO> dtos = null;
    CarService service = new CarService();
    CarDAO db = new CarDAO();
    Scanner sc = new Scanner(System.in);
    int choice=0;

    while(true)
    {
        menu();
        choice=sc.nextInt();
        if(choice==1) {
            dtos=service.getAllCars() ; //모든 차량 조회
            for(int i=0; i<dtos.size(); i++) {
                System.out.printf("원산지: %s\n",dtos.get(i).getOrigin());
                System.out.printf("제조사: %s\n",dtos.get(i).getMake());
                System.out.printf("차량번호: %s\n",dtos.get(i).getCarnum());
                System.out.printf("차이름: %s\n",dtos.get(i).getName());
                System.out.printf("사고유무: %s\n",dtos.get(i).getAccident());
                System.out.printf("차종: %s\n",dtos.get(i).getCartype());
                System.out.printf("연료: %s\n",dtos.get(i).getFuel());
                System.out.printf("연식: %s\n",dtos.get(i).getP_year());
                System.out.printf("주행거리: %dKm\n",dtos.get(i).getMileage());
                System.out.printf("차량가격: %d원\n",dtos.get(i).getPrice());
                System.out.printf("차량색상: %s\n",dtos.get(i).getColor());
                System.out.printf("-----");
                System.out.printf("\n");
            }
        }
    }
}
```

```
.....
}else if(choice==2) { // 선택 차량조회
    Serch_menu();
    int num=sc.nextInt();
    switch(num) {
        case 1:
            System.out.print("원산지 입력: ");
            String input_search = sc.next();
            dtos = service.SearchCar("origin", input_search);
            break;
        case 2:
            System.out.print("제조사 입력: ");
            String input_search2 = sc.next();
            dtos = service.SearchCar("make", input_search2);
            break;
        case 3:
            System.out.print("차량이름 입력: ");
            String input_search3 = sc.next();
            dtos = service.SearchCar("name", input_search3);
            break;
        case 4:
            System.out.print("사고유무 입력: ");
            String input_search4 = sc.next();
            dtos = service.SearchCar("accident", input_search4);
            break;
        case 5:
            System.out.print("차종 입력: ");
            .....
```

컨트롤러 조회 부분

2.2 Java

Controller . 2

```
    }else if(choice==3) { //차량정보 등록
        System.out.print("원산지 입력: ");
        String origin=sc.next();
        System.out.print("제조사 입력: ");
        String make=sc.next();
        System.out.print("차량번호 입력: ");
        String carnum=sc.next();
        System.out.print("차량이름 입력: ");
        String name=sc.next();
        System.out.print("차량 사고유무 입력: ");
        String accident=sc.next();
        System.out.print("차종 입력: ");
        String cartype=sc.next();
        System.out.print("연료 입력: ");
        String fuel=sc.next();
        System.out.print("연식 입력: ");
        String p_year=sc.next();
```

컨트롤러 차량 등록

```
    }else if(choice==4) { //차량정보 수정
        Update_menu();
        int choice3 = sc.nextInt();
        switch(choice3) {
            case 1:
                System.out.print("원산지 입력: ");
                String input_content = sc.next();
                System.out.print("변경할 차량번호 입력: ");
                String input_carnum = sc.next();
                dtos=service.UpdateCar("origin", input_content, input_carnum);
                System.out.print("변경 되었습니다.\n");
                break;
            case 2:
                System.out.print("제조사 입력: ");
                String input_content2 = sc.next();
                System.out.print("변경할 차량번호 입력: ");
                String input_carnum2 = sc.next();
                dtos=service.UpdateCar("make", input_content2, input_carnum2);
                System.out.print("변경 되었습니다.\n");
                break;
            case 3:
                System.out.print("차량번호 입력: ");
                String input_carnum3 = sc.next();
                dtos=service.DeleteCar(input_carnum3);
                System.out.print("삭제되었습니다.\n");
                break;
            default:
                System.out.print("잘못된 입력입니다.\n");
                break;
        }
        sc.close();
    }
}

}else if(choice==5) { //차량정보 삭제
    System.out.printf("삭제할 차량 번호 입력: ");
    String carnum = sc.next();
    dtos=service.DeleteCar(carnum);
}

}else if(choice==6) { // 종료
    break;
}
}
```

컨트롤러 차량 정보 수정

컨트롤러 차량 정보 삭제

2.2 Java

Controller . 3

```
public static void menu() {
    System.out.println("1.모든차량 조회");
    System.out.println("2.선택차량 조회");
    System.out.println("3.차량정보 등록");
    System.out.println("4.차량정보 수정");
    System.out.println("5.차량정보 삭제");
    System.out.println("6.종료");
}

public static void Serch_menu() {
    System.out.println("1. 원산지 조회");
    System.out.println("2. 제조사 조회");
    System.out.println("3. 차이름으로 조회");
    System.out.println("4. 사고유무 조회");
    System.out.println("5. 차종 조회");
    System.out.println("6. 연료 조회");
    System.out.println("7. 연식 조회");
    System.out.println("8. 주행거리 조회");
    System.out.println("9. 차량가격 조회");
    System.out.println("10. 차량색상 조회");
}

public static void Update_menu() {
    System.out.println("1. 원산지 변경");
    System.out.println("2. 제조사 변경");
    System.out.println("3. 차이름 변경");
    System.out.println("4. 사고유무 변경");
    System.out.println("5. 차종 변경");
    System.out.println("6. 연료 변경");
    System.out.println("7. 연식 변경");
    System.out.println("8. 주행거리 변경");
    System.out.println("9. 차량가격 변경");
    System.out.println("10. 차량색상 변경");
}
```

메뉴 화면

2.2 Java

Service



Service는 사용자 요청에 대해 어떤 처리를 할지 결정하는 파트입니다.

```
private CarDAO dao;

public CarService() {
    dao=new CarDAO();
}

public ArrayList<CarVO> getAllCars(){
    return dao.getAllCar();
}

public ArrayList<CarVO> InsertCar(String origin, String make, String carnum,
    String cartype, String fuel, String p_year, int mileage, int price, S

    return dao.InsertCar(origin, make, carnum, name, accident, cartype, fuel,
}

public ArrayList<CarVO> DeleteCar(String carnum) {

    return dao.DeleteCar(carnum);
}

public ArrayList<CarVO> UpdateCar(String content, String input_content, Strin

    return dao.UpdateCar(content, input_content, input_carnum);
}
```

Part 2.

실행



3.1 실행화면

전체조회

1. 모든차량 조회
2. 선택차량 조회
3. 차량정보 등록
4. 차량정보 수정
5. 차량정보 삭제
6. 종료

원산지: 국산
제조사: 현대
차량번호: 39러4915
차이름: 팰리세이드 2WD
사고유무: 무사고
차종: 준대형SUV
연료: 디젤
연식: 2019-01-01 00:00:00.0
주행거리: 11495Km
차량가격: 43335000원
차량색상: 검정색

원산지: 국산
제조사: 현대
차량번호: 327어9577
차이름: 아반떼(CN7)
사고유무: 무사고
차종: 준중형
연료: 가솔린
연식: 2020-05-01 00:00:00.0
주행거리: 19445Km
차량가격: 25635000원
차량색상: 화이트

원산지: 국산
제조사: 쉐보레
차량번호: 264루9845
차이름: 더뉴스파크프리미어
사고유무: 단순교환
차종: 경차
연료: 가솔린
연식: 2020-07-01 00:00:00.0
주행거리: 3240Km
차량가격: 10579500원
차량색상: 진주색

원산지: 국산
제조사: 현대
차량번호: 167어7240
차이름: 투싼(4세대)
사고유무: 무사고
차종: 준중형SUV
연료: 하이브리드
연식: 2020-11-01 00:00:00.0
주행거리: 9306Km
차량가격: 42035000원
차량색상: 흰색

원산지: 해외
제조사: BMW
차량번호: 10더9454
차이름: 520dxDriveM스포츠플러스
사고유무: 무사고
차종: 중형
연료: 디젤
연식: 2017-12-01 00:00:00.0
주행거리: 51947Km
차량가격: 47645000원
차량색상: 쥐색

3.1 실행화면

조건 조회

1. 모든차량 조회
2. 선택차량 조회
3. 차량정보 등록
4. 차량정보 수정
5. 차량정보 삭제
6. 종료

2

1. 원산지 조회
2. 제조사 조회
3. 차이름으로 조회
4. 사고유무 조회
5. 차종 조회
6. 연료 조회
7. 연식 조회
8. 주행거리 조회
9. 차량가격 조회
10. 차량색상 조회

3

차이름 입력: 아반떼

원산지: 국산

제조사: 현대

차량번호: 327어9577

차이름: 아반떼(CN7)

사고유무: 무사고

차종: 준중형

연료: 가솔린

연식: 2020-05-01 00:00:00.0

주행거리: 19445Km

차량가격: 25635000원

차량색상: 화이트

1. 모든차량 조회

2. 선택차량 조회

6. 연료 조회

7. 연식 조회

8. 주행거리 조회

9. 차량가격 조회

10. 차량색상 조회

8

시작 주행거리 입력: 5000

끝 주행거리 입력: 35000

원산지: 국산

제조사: 현대

차량번호: 39러4915

차이름: 팰리세이드 2WD

사고유무: 무사고

차종: 준대형SUV

연료: 디젤

연식: 2019-01-01 00:00:00.0

주행거리: 11495Km

차량가격: 43335000원

차량색상: 검정색

원산지: 국산

제조사: 현대

차량번호: 327어9577

차이름: 아반떼(CN7)

사고유무: 무사고

차종: 준중형

연료: 가솔린

연식: 2020-05-01 00:00:00.0

주행거리: 19445Km

차량가격: 25635000원

차량색상: 화이트

3.1 실행화면

등록 및 변경, 삭제

0.공포

3

원산지 입력: 해외

제조사 입력: 쌍용

차량번호 입력: 11거1111

차량이름 입력: 티볼리

차량 사고유무 입력: 무사고

차종 입력: 소형SUV

연료 입력: 디젤

연식 입력: 2018-10-01

주행거리 입력: 56000

차량가격 입력: 24000000

차량색상 입력: 아이보리

등록 되었습니다.

1. 모든차량 조회

2. 선택차량 조회

3. 차량정보 등록

차량 정보 등록

6.종료

4

1. 원산지 변경

2. 제조사 변경

3. 차이름 변경

4. 사고유무 변경

5. 차종 변경

6. 연료 변경

7. 연식 변경

8. 주행거리 변경

9. 차량가격 변경

10. 차량색상 변경

1

원산지 입력 : 국내

변경할 차량번호 입력 : 11거1111

변경 되었습니다.

1. 모든차량 조회

2. 선택차량 조회

3. 차량정보 등록

4. 차량정보 수정

차량 정보 변경

4. 차량정보 수정

5. 차량정보 삭제

6. 종료

5

삭제할 차량 번호 입력: 11거1111

삭제되었습니다

차량 정보 삭제

Part 4.

마무리



감사합니다