

---

# 데이터베이스시스템(CSE4110)

## Project 1 보고서

---

전공: 미국문화      학년: 4      학번: 20170175      이름: 김태안

## 1. 프로젝트 목표

GM 이나 Ford 와 같은 자동차 회사의 데이터베이스를 설계해본다. 이를 통해 관계형 데이터베이스의 개념적 설계와 논리적 설계, 적용, 작동, 유지보수를 경험한다. 자동차 회사는 미국의 전기 자동차 회사 테슬라 주식회사 (Tesla, Inc.)를 참고해 데이터베이스를 설계했다.

## 2. E-R Model

### 2.1 Entities

고객이 제시한 고려사항과 쿼리 예시를 통해 다음의 엔티티와 속성, Primary Key 를 도출했다.

company	회사. 테슬라 주식회사의 정보를 저장한다.
<u>company_name</u>	회사명. 법적으로 회사명은 중복될 수 없고, 이 데이터베이스에는 테슬라만 저장될 것이기 때문에 Primary Key 로 설정 가능하다.

brand	자동차의 브랜드. 테슬라 주식회사는 아직까지는 Tesla 브랜드만을 가진다.
<u>brand_name</u>	브랜드명. 브랜드명은 중복되게 설정하지 않으므로 Primary Key 로 사용 가능하다.

model	테슬라는 자동차의 크기와 세대에 따라 다른 모델을 가지고 있다.
<u>model_ID</u>	ID 는 모델명과 세대 등에 따라 발급된 코드이다.
model_type	테슬라는 차의 크기에 따라 Model S, Model 3, Model X, Model Y, Roadster, Cybertruck, Semi 로 모델을 분류하고 있다.
model_name	각 모델은 모터와 기타 기능에 따라 세부 모델로 분류된다. 예를 들어, 2021 Tesla Model Y 는 Single Motor Standard Range, Dual Motor AWD, Dual Motor Performance 의 세부 모델이 있다.

model_year	모델 출시 연도로 세대를 구분한다.
is_aval	모델의 단종 여부를 표시한다.

option	자동차의 세부 옵션. 실제로는 배터리 용량, 충전기 등 더 많은 옵션이 있지만, 이번 프로젝트에서는 모델에 따라 정해진 기본 옵션은 제외하고 사용자가 구매 시 선택 가능한 옵션 5 개만을 사용하였다.
<u>option_ID</u>	옵션에 따른 코드가 존재하며, 옵션 별로 4 자리 알파벳으로 이루어져 있다.
paint	자동차의 색
wheels	자동차의 휠
interior	내부 인테리어
seat	시트 구성
self_driving	오토파일럿 기능

vehicle	회사의 고객이 구매한 자동차 한 대.
<u>VIN</u>	모든 차는 고유한 Vehicle Number(VIN)을 가지고 있다.

customer	회사의 회원으로 등록된 고객
<u>customer_ID</u>	회원마다 고유한 ID 를 부여해 구분한다.
type	회사 회원과 개인 회원을 구분한다.
name	이름
address	주소
phone	연락처
gender	성별. 회사는 비워 둔다.
income	수입

dealer	회사의 자동차를 판매하는 판매처
<u>dealer_name</u>	판매처의 이름. 지점명으로 중복되지 않는다.
address	주소
phone	연락처

<b>supply_plant</b>	자동차가 필요한 부품을 생산하는 회사의 공장
<u>plant_name</u>	공장의 이름

<b>supplier</b>	자동차가 필요한 부품을 생산해 납품하는 외부 회사
<u>supplier_name</u>	납품업체의 이름

<b>part</b>	자동차 조립에 필요한 부품. supply_plant 와 supplier 가 납품한 부품이다.
<u>part_ID</u>	부품별로 ID 를 부여해 구분한다.
part_type	Transmission 과 같은 부품의 종류

<b>assembly_plant</b>	부품을 조립해 자동차를 생산하는 회사의 공장
<u>plant_name</u>	공장의 이름

## 2.2 Relationships

위 엔티티는 아래와 같은 관계를 가진다.

- *owns\_brand*

company 와 brand 의 관계이다. 한 회사는 여러 개의 브랜드를 가질 수 있고, 브랜드는 하나의 회사와 연결되는 일대다 관계이다. 회사는 브랜드를 소유하고 있지 않을 수 있지만, 브랜드는 반드시 하나의 회사에 소속되어야 한다.

- *has\_model*

brand 와 model-option 의 관계이다. 브랜드 별로 여러 종류의 모델-옵션의 조합을 가지며, 모델-옵션은 하나의 브랜드에 속하는 일대다 관계이다.

- *option\_aval*

model 과 option 의 관계이다. 모델 별로 특정 조합의 옵션을 선택할 수 있는 다대다 관계이다. 이 관계는 하나의 엔티티로 aggregate 되어 다른 엔티티와 관계를 가진다. 모든 모델은 옵션을 가진다.

- *has\_model\_opt*

vehicle 과 model-option 의 관계이다. 모든 차량은 어떤 모델과 옵션을 가지는 다대일 관계이다. 어떤 모델-옵션 조합은 해당하는 차량이 없을 수 있다.

- *owns\_vehicle*

vehicle 과 customer 의 관계이다. 고객은 0 개 이상의 차량을 가질 수 있고, 차량은 한 명의 고객의 소유인 다대일 관계이다.

- *sells\_vehicle*

dealer 와 vehicle 의 관계이다. 모든 차량은 브랜드 소속 판매처에서 판매된다고 가정한다. 이때, 판매처와 차량의 관계는 일대다 관계이다. 판매 가격과 날짜를 기록해 관련 질의를 처리할 수 있게 한다.

- *sells\_brand*

brand와 dealer의 관계이다. 브랜드는 0개 이상의 판매처를 보유하고 있고, 판매처는 여러 브랜드를 판매하는 경우를 포함해 1 개 이상의 브랜드를 판매하는 다대다 관계이다.

- *pnt\_makes\_part*

supply\_plant 와 part 의 관계이다. 회사의 부품공장에서 부품을 납품한 날짜를 기록한다. 공장은 0 개 이상의 부품을 생산하고, 부품은 하나의 공장에서 생산된 일대다 관계이다.

- *sup\_makes\_part*

supplier 와 part 의 관계이다. 부품 납품 업체에서 부품을 납품한 날짜를 기록한다. 업체는 0 개 이상의 부품을 납품하고, 부품은 하나의 납품 업체가 납품한 일대다 관계이다.

- *owns\_sup\_plant*

company 와 supply\_plant 의 관계이다. 회사는 0 개 이상의 공장을 보유하고, 공장은 하나의 회사에 소속된 일대다 관계이다.

- *owns\_asm\_plant*

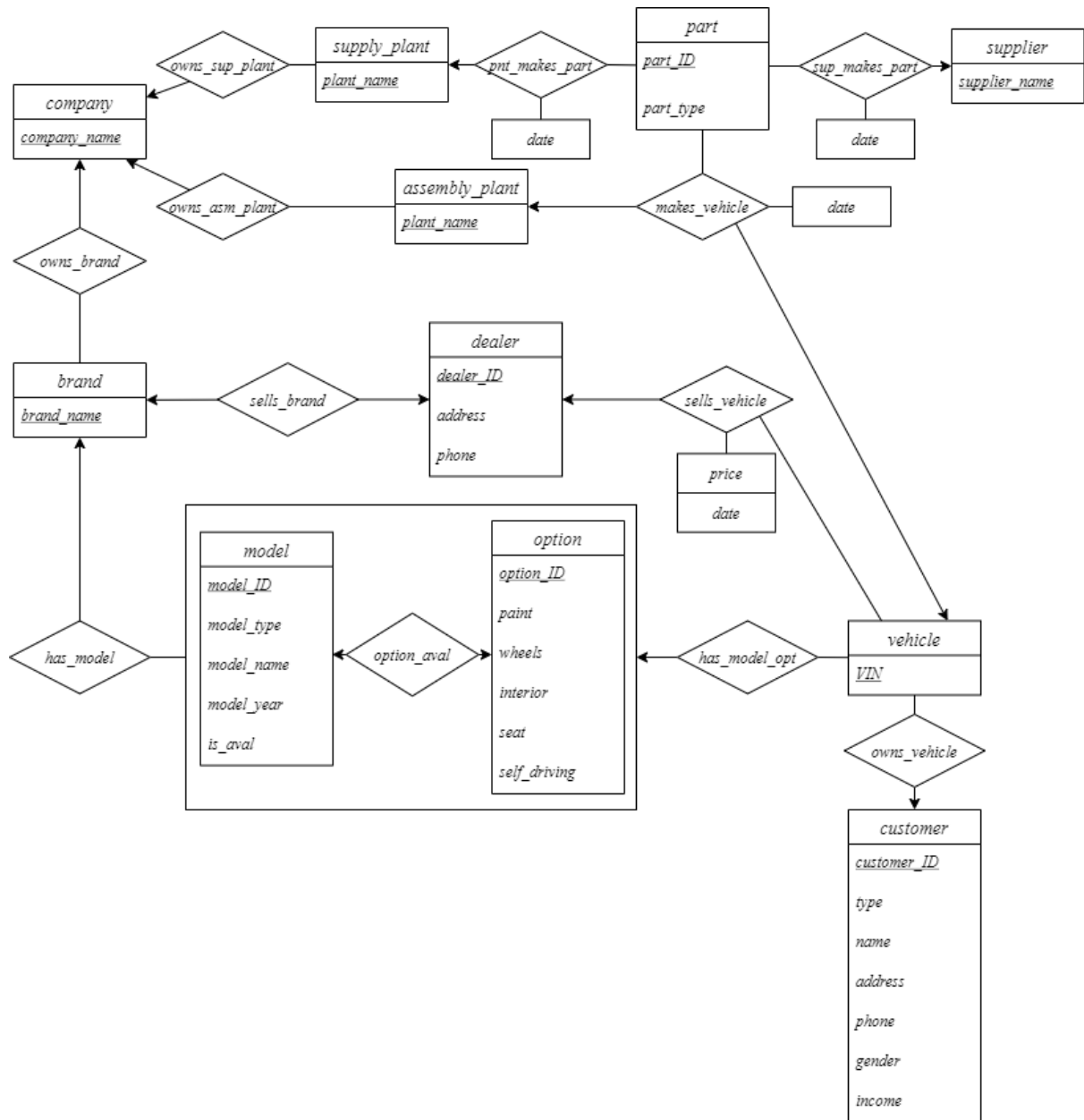
company 와 assembly\_plant 의 관계이다. 회사는 0 개 이상의 공장을 보유하고, 공장은 하나의 회사에 소속된 일대다 관계이다.

- *makes\_vehicle*

part 와 assembly\_plant, vehicle 의 관계이다. 부품이 하나의 자동차에 사용되고, 하나의 자동차는 한 공장에서 조립되며, 조립 공장에서 여러 자동차를 생산하는 일대다대다 관계이다. 조립 날짜를 저장한다.

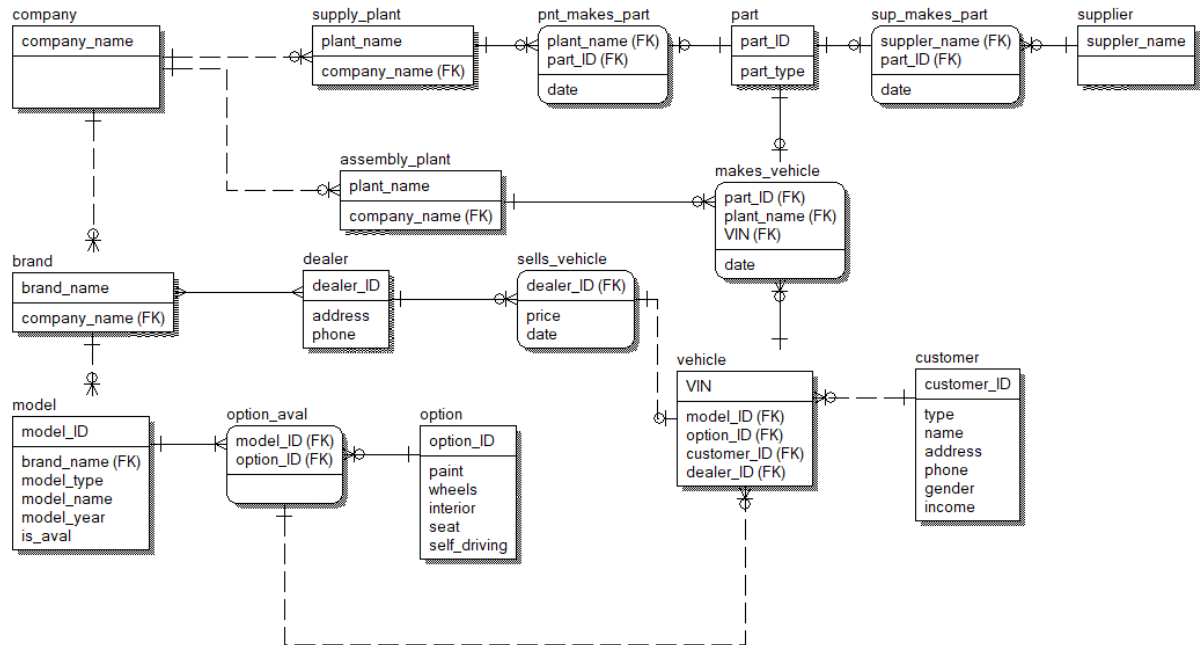
## 2.3 E-R Diagram

위의 내용을 바탕으로 구현한 E-R Model의 다이어그램은 다음과 같다.



### 3. Schema Diagram

구현한 ER Model 을 ERwin 을 통해 Schema Diagram 으로 옮기면 다음과 같다.



### 4. 참고 자료

Tesla 대한민국, [www.tesla.com/ko\\_kr](http://www.tesla.com/ko_kr).

네이버 자동차, [auto.naver.com/](http://auto.naver.com/).

“Option Codes.” *Tesla JSON API (Unofficial)*, [tesla-api.timdorr.com/vehicle/optioncodes](http://tesla-api.timdorr.com/vehicle/optioncodes).