

DATABASE SYSTEM

Project 1

E-R design and Relational Schema design

Automobile Company

컴퓨터공학과

20181668

이예진

목차

1. 프로젝트 개요

2. 구현 결과물

2.1. E-R Diagram

2.1.1. Entity Sets

2.1.2. Relation Sets

2.2. Relational Schema Diagram

2.2.1. 결과물 설명

2.2.2. 릴레이션 스키마로의 축소 과정 설명

3. Queries 적용

1. 프로젝트 개요

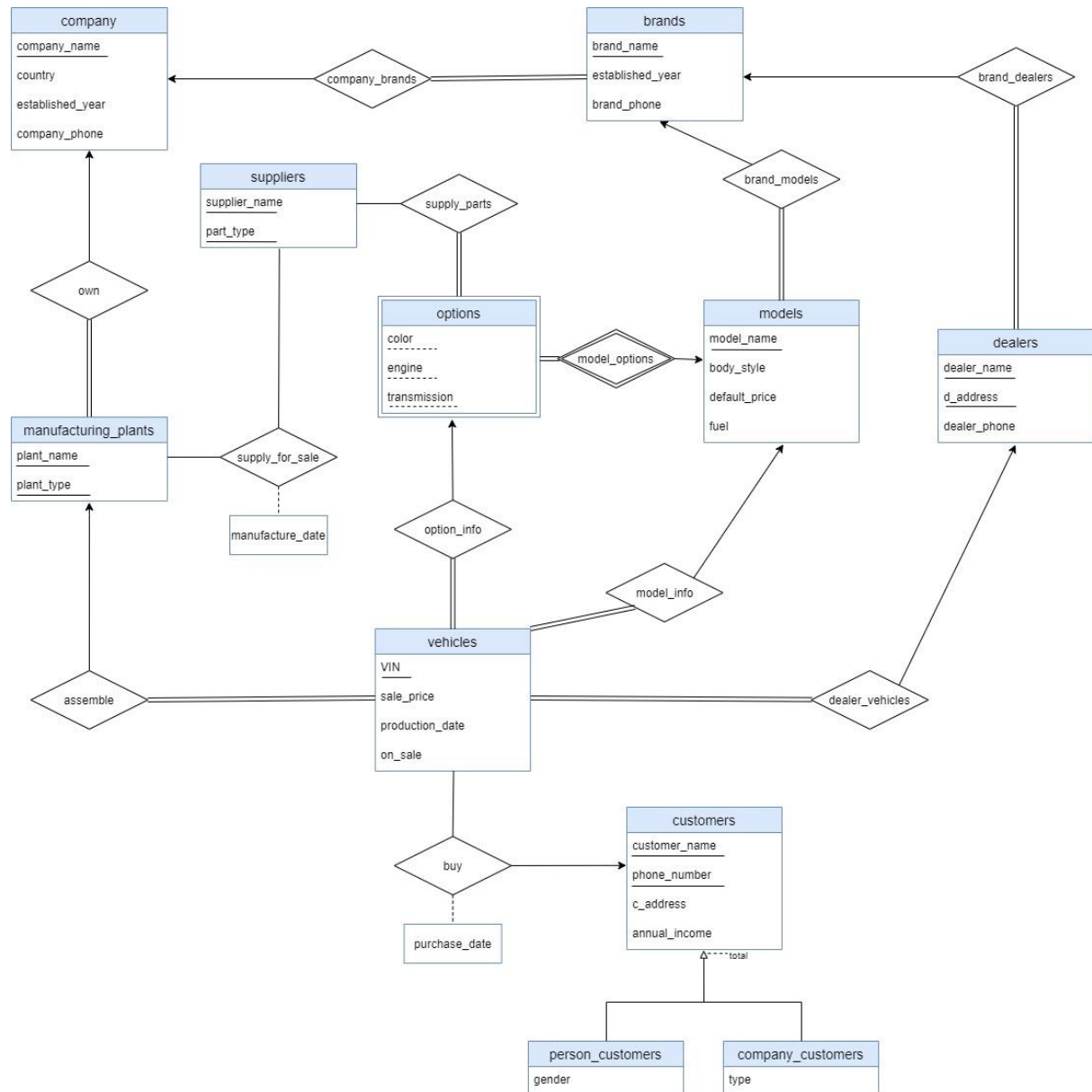
본 프로젝트의 목표는 관계형 데이터베이스와 관련 애플리케이션의 개념적 디자인, 논리적 디자인, 구현, 운영 및 유지 보수에 대한 경험이다.

본인을 가상의 자동차 회사의 DBA라고 가정한다. 이 가상의 회사에서 회사 운영의 기반이 되는 데이터베이스의 주요 부분을 재설계하기로 결정한 상황이다. 데이터베이스 설계 제안서를 요청하기 위해 할당된 관리자는 컴퓨터에 능통하지 않기 때문에 상세한 사양을 제공할 수 없다. 그렇기 때문에 다음에 이어질 분석 부분에서 제시한 측면에 초점을 맞춰 데이터베이스 설계 제안서를 작성해야 한다.

E-R Diagram에는 명세서에 존재하는 모든 개체와 관계 집합들을 포함해야 한다. 필요하다면 더 추가할 수 있다. 또한 추후 클라이언트가 원하는 정보를 데이터베이스에서 가져올 수 있도록 주어진 질의문들을 참고해야 한다.

2. 구현 결과물

2.1. E-R Diagram



2.1.1. Entity Sets

(1) company

자동차 회사의 정보를 저장한다. 속성으로 회사 이름, 회사의 국적, 설립

연도, 회사 대표 전화번호가 있다. 같은 이름을 갖는 회사는 없기 때문에 PK는 회사 이름이다. 본 프로젝트에서는 하나의 회사에 대한 정보만 저장해도 되지만 다른 회사들의 정보도 같이 저장할 수 있는 확장성을 위해 해당 개체 집합을 추가하였다.

(2) brands

자동차 회사가 갖는 브랜드의 정보를 저장한다. 속성으로 브랜드 이름, 설립 연도, 대표 전화번호가 있다. 같은 이름을 갖는 브랜드는 없기 때문에 PK는 브랜드 이름이다.

(3) models

자동차 모델의 정보를 저장한다. 속성으로는 모델 이름, 모델의 차제 외형, 옵션을 추가하지 않았을 때 기본 가격, 연료 정보가 있다. 보통 실제 자동차는 모델 이름이 고유하기 때문에 PK는 모델 이름이다. 모델이름에 자동차의 외형이 표기되지 않은 경우도 있기 때문에 자동차의 외형의 정보를 저장할 body_style을 따로 지정하였다. 예를 들어 실제 미니라는 브랜드의 해치백 모델 중에는 미니 해치백 5도어라는 모델명을 가진 자동차도 있지만, 미니 해치백이라는 모델명을 가진 자동차도 있다. 미니 해치백은 3도어이지만 모델명에 따로 표기되지는 않았다. 이런 경우를 위해 미니 해치백은 모델명으로 저장하고 body_style에 3도어를 저장하여 외형의 정보도 포함하도록 설계하였다.

(4) options

선택할 수 있는 옵션의 정보를 저장한다. 약성 개체 집합으로 options에 대한 식별 개체 집합은 models이다. 약성 개체 집합은 PK를 갖지는 않지만 구별자라는 속성들의 집합을 갖는다. 속성으로는 색깔 정보, 엔진 종류, 변속기 종류가 있다. options의 PK는 구별자와 models의 PK를 합친 {model_name, color, engine, transmission} 이다. 하나의 모델에 대해 여러 가지 옵션이 가능한 경우가 있다. 편의를 위해 본 프로젝트에서는 엔진과 변속기의 각 부품을 세세하게 나누지 않고 하나의 값만 갖도록 했다. 예를 들어 engine에는 엔진 형식 값이 저장되는데 14, 16 등의 값이 들어갈 수 있다. 그리고 transmissstion에는 변속기 종류가 저장되는데 자동 8단, 자동 7동 등의 값이 들어갈 수 있다.

(5) vehicles

판매되는 하나의 상품인 자동차의 정보를 저장한다. 속성에는 차량 식별 번호, 최종 판매 가격, 생산 날짜, 판매 중인지 이미 팔렸는지에 대한 정보가 있다. 차량 식별 고유 번호는 고유한 값이므로 PK는 VIN이다. on_sale 속성의 값으로는 이미 판매되었음을 나타내는 no와 아직 판매 중이라는 것을 나타내는 yes가 들어갈 수 있다. 하나의 모델은 옵션에 따라 가격이 달라지므로 최종 판매 가격을 따로 저장한다.

(6) dealers

최종 상품인 vehicles를 고객들에게 판매하는 딜러점의 정보를 저장한다. 속성에는 딜러점의 이름, 주소, 지점 대표 전화번호가 있다. 딜러점의 이름과 주소를 합친 정보는 유일하게 하나의 개체를 식별할 수 있게 하므로 PK는 {dealer_name, d_address} 이다. 예를 들어 실제로 폭스바겐 브랜드의 딜러점의 종류는 클라세오토, 아우토플라츠가 있다. 그리고 한국에는 각 딜러점이 각 지역에 위치한다. 예를 들어 클라세오토는 수원지점, 강남 신사점 등 여러 곳에 있다. 그러므로 클라세오토 수원점, 클라세오토 강남 신사점은 각각 dealers의 하나의 개체가 될 수 있다.

(7) manufacturing_plants

각 회사가 가진 제조 공장의 정보를 저장한다. 속성에는 공장의 이름과 공장의 종류가 있다. 공장 이름과 공장 종류를 합친 정보는 유일하게 하나의 개체를 식별할 수 있게 하므로 PK는 {plant_name, plant_type} 이다. plant_type은 조립과 부품 생산 두 개 중 하나의 값을 저장한다. 예를 들어 A라는 이름을 가진 제조 공장이 있는데 이 공장은 자동차 조립과 부품 생산 두 가지를 모두 한다. 이 경우 {A, assemble}, {A, part} 2개의 튜플 값을 갖을 수 있을 것이다.

(8) suppliers

제조 공장에서 생성한 부품을 공급해주는 공급점의 정보를 저장한다. 속성으로는 공급점 이름, 부품 종류가 있다. 공급점 이름, 부품 종류를 합친 정보는 유일하게 하나의 개체를 식별할 수 있게 하므로 PK는 {supplier_name, part_type} 이다. part_type은 이 공급점이 공급하는 부품의 종류가 값으로 저장되는데 본 프

로젝트에서는 편의를 위해 부품을 엔진과 변속기 2개로 한정했으므로 가능한 값은 엔진 또는 변속기이다. 이때 편의를 위해 part_type이 engine이면 엔진에 필요한 모든 부품을 공급하는 것으로 하고 transmission이라면 변속기에 필요한 모든 부품을 공급하는 것으로 한다. 추가 조건으로 part_type에는 공급점 이름과 공급 날짜에 대한 정보도 포함되어 있다. 예를 들어 e_ge_210414는 ge사가 21년 4월 14일에 공급한 엔진 부품이고 t_vw_210414는 vw사가 21년 4월 14일에 공급한 변속기 부품이다.

(9) customers

차량을 구매하는 고객의 정보를 저장한다. 속성으로는 고객 이름, 고객 전화 번호, 주소, 연 수익이 있다. 고객 이름과 전화번호를 합친 정보는 유일하게 하나의 개체를 식별해줄 수 있으므로 PK는 {customer_name, phone_number} 이다. 이 개체 집합은 세분화를 통해 person_customer, company_customer이라는 하위 그룹을 갖는다. 어떤 개체는 개인이면서 기업일 수 없으므로 중복되지 않는 세분화를 적용하였다. 또한 개체가 세분화에서 반드시 하나 이상의 하위 개체 집합에 속해야 하므로 total 일반화를 적용하였다.

(10) person_customers

customers의 하위 그룹이다. 속성은 customers의 모든 속성에 gender라는 속성을 추가로 갖는다. 기업은 성별이 없지만 개인 고객은 성별을 갖으므로 gender를 속성으로 추가해서 구분되도록 했다.

(11) company_customers

customers의 하위 그룹이다. 속성은 customers의 모든 속성에 type이라는 속성을 추가로 갖는다. 개인 고객은 종류가 없지만 기업은 어떤 기업인지 기업 종류가 있으므로 type을 속성으로 추가해서 구분되도록 했다.

2.1.2. Relation Sets

(1) company_brands

각 회사는 여러 개의 브랜드를 소유할 수 있으므로 일대다 관계이다. 브랜

드는 하나의 회사에 반드시 소속되어야 하므로 브랜드 개체 집합의 참가는 전체적이다.

(2) brand_models

각 브랜드는 여러 개의 자동차 모델을 소유할 수 있으므로 일대다 관계이다. 자동차 모델은 하나의 브랜드에 반드시 소속되어야 하므로 모델 개체 집합의 참가는 전체적이다.

(3) model_info

각 자동차 상품은 반드시 어떤 모델에 속하므로 models와 vehicles는 일대다 관계이다. vehicles 개체 집합의 참가는 전체적이다.

(4) option_info

각 자동차 상품은 반드시 어떤 옵션값을 갖으므로 options와 vehicles는 일대다 관계이다. vehicles 개체 집합의 참가는 전체적이다.

(5) model_options

options라는 약성 개체 집합을 상응하는 식별 개체 집합인 models에 연관 짓는 역할을 하는 식별 관계이다. options으로부터 models으로의 다대일이고 그 관계에서 options의 참가는 전체적이다. 그리고 이 관계 집합은 어떠한 설명 속성도 갖지 않는다.

(6) brand_dealers

각 브랜드는 여러 개의 딜러점을 소유할 수 있으므로 일대다 관계이다. 딜러점은 하나의 브랜드에 반드시 소속되어야 하므로 dealers 개체 집합의 참가는 전체적이다.

(7) dealer_vehicles

각 딜러점은 여러 개의 자동차 상품을 소유할 수 있으므로 일대다 관계이다. 자동차 상품은 하나의 딜러점에 반드시 소속되어야 하므로 vehicles 개체 집합의 참가는 전체적이다.

(8) own

각 회사는 여러 개의 제조 공장을 소유할 수 있으므로 일대다 관계이다. 제조공장은 하나의 회사에 반드시 소속되어야 하므로 manufacturing plants 개체 집합의 참가는 전체적이다.

(9) assemble

각 제조 공장은 여러 개의 자동차 상품을 조립할 수 있으므로 일대다 관계이다. 자동차 상품은 반드시 하나의 제조 공장에서 조립되어야 하므로 vehicles 개체 집합의 참가는 전체적이다.

(10) supply_for_sale

판매를 위해 제조 공장에서 공급점으로 부품을 공급하는 관계를 나타낸다. 각 제조 공장은 여러 개의 공급점에 부품을 공급할 수 있다. 또한 각 공급점은 여러 제조 공장으로부터 부품을 공급받을 수 있다. 예를 들어 A라는 supplier가 있다면 이 공급점에 엔진 부품은 B라는 제조 공장으로부터 공급받고, 변속기 부품은 C라는 제조 공장으로부터 공급 받을 수 있다. 또한 반대의 경우를 보면, B라는 제조 공장은 엔진 부품을 제조하는 공장인데 이 엔진 부품을 E와 F 공급점에 공급할 수 있다. 따라서 다대다 관계라고 할 수 있다. 또한 반드시 제조 공장 개체 집합의 모든 개체, 공급점 개체 집합의 모든 개체가 관계에 참여할 필요는 없다. 따라서 두 개체 집합의 참여는 부분적이다. 이 관계 집합은 manufacture_date 라는 설명 속성을 갖는다. 편의를 위하여 공장에서 생산한 날짜에 공급점에 부품을 공급했다고 정의한다.

(11) supply_parts

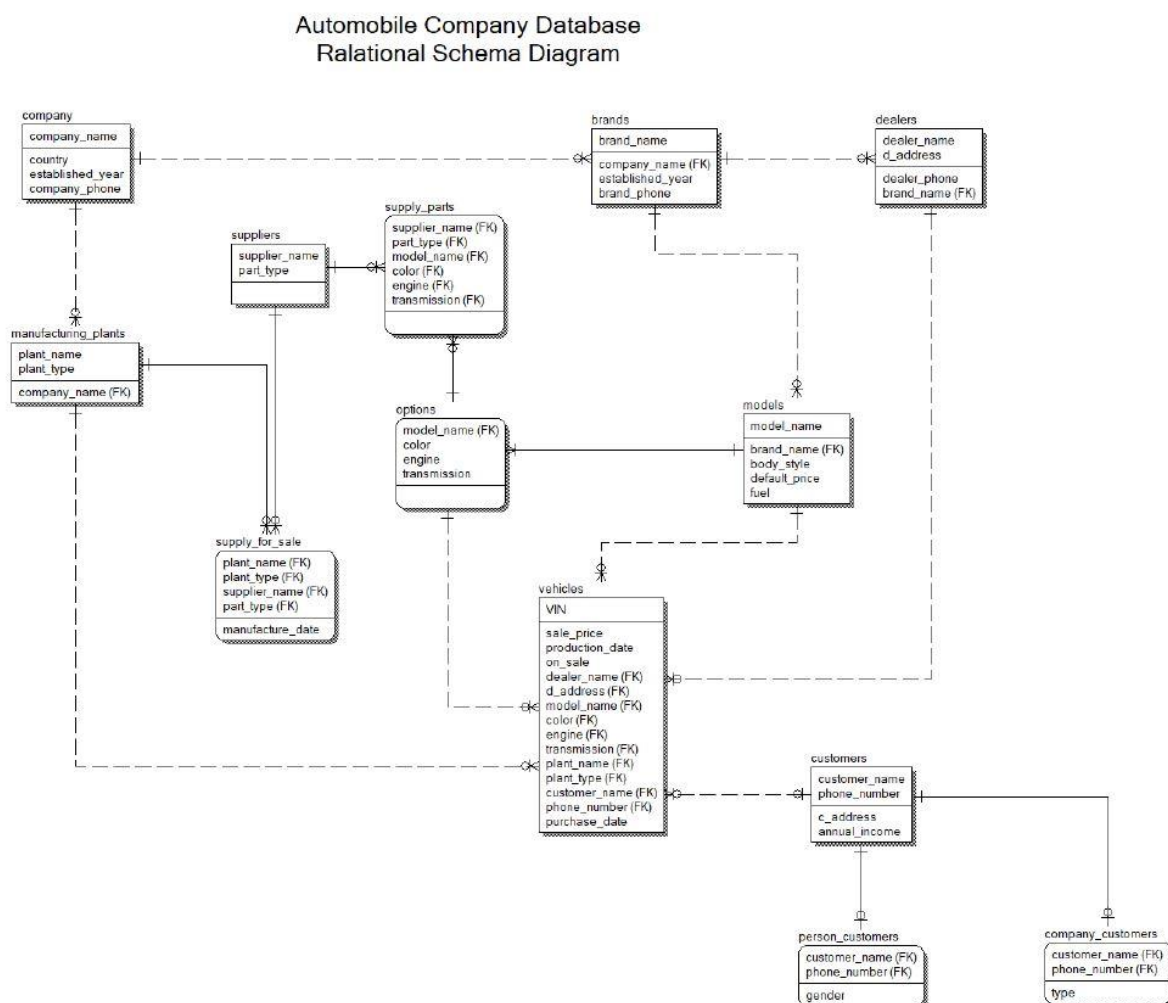
공급점은 각 옵션에 맞게 부품을 공급해줄 수 있다. 편의를 위해 어떤 종류의 엔진에 대해 필요한 부품을 모두 공급할 수 있다고 가정한다. 예를 들어 옵션에서 I4 엔진이라고 하면 하나의 공급점에서 이 엔진에 필요한 모든 부품을 공급한다. 변속기의 경우도 마찬가지이다. 하나의 공급점은 여러 개의 옵션에 대해 부품을 공급할 수 있고 반대로 하나의 옵션은 여러 개의 공급점으로부터 공급받을 수 있으므로 다대다 관계이다. 예를 들어 A라는 공급점은 part_type이 engine 이어서 엔진에 관한 부품을 모두 생산할 수 있는 기술이 있다고 가정한다. 그러면 engine 속성에 대한 값이 I4인 옵션 B와 C에게 공급할 수 있다. 반대의 경우

를 보면 어떤 옵션 개체에 대해 엔진과 변속기를 각각 다른 공급점으로부터 공급 받을 수 있다. options는 반드시 하나의 공급점으로부터 공급을 받아야 하므로 참가는 전체적이다. 편의를 위해 엔진과 변속기를 같은 날짜에 공급한다고 정의한다.

(12) buy

각 고객은 여러 개의 자동차 상품을 구입할 수 있으므로 일대다 관계이다. 자동차 상품은 반드시 구입되지 않아도 되므로 vehicles 개체 집합의 참가는 부분적이다. 이 관계 집합은 purchase_date라는 설명 속성을 갖는다. 고객이 자동차 상품을 구입한 날짜를 저장한다.

2.2. Relational Schema Diagram



2.2.1. 결과물 설명

- Entities

Entity	속성	Key	설명
company	company_name	PK	회사 이름
	country		회사 국적
	established_year		회사 설립연도
	company_phone		회사 대표 번호
brands	brand_name	PK	브랜드 이름
	company_name	FK	company 참조
	established_year		브랜드 설립연도
	brand_phone		브랜드 대표 번호
dealers	dealer_name	PK	딜러 이름
	d_address		딜러 주소
	dealer_phone		딜러 대표 번호
	brand_name	FK	brands 참조
models	model_name	PK	자동차 모델 이름
	brand_name	FK	brands 참조
	body_style		자동차 외형 종류
	default_price		자동차 기본 가격
	fuel		자동차 연료 종류
options	model_name	PK, FK	models 참조
	color	PK	자동차 색상
	engine	PK	자동차 엔진 부품
	transmission	PK	자동차 변속기 부품
vehicles	VIN	PK	자동차 식별 번호
	sale_price		자동차 판매 가격
	production_date		자동차 생산 날짜
	on_sale		자동차 판매중('Y'), 이미 판매됨('N')
	dealer_name	FK	dealers 참조
	d_address		
	model_name	FK	models, options 참조
	color	FK	model_name과 합쳐서 options 참조
	engine		
	transmission		
	plant_name	FK	manufacturing_plants 참조
	plant_type		

	customer_name	FK	customers 참조
	phone_number		
	purchase_date		자동차 판매 날짜
supply_parts	supplier_name	PK, FK	supplier 참조
	part_type		options 참조
	model_name		
	color		
	engine		
	transmission		
suppliers	supplier_name	PK	공급점 이름
	part_type	PK	공급하는 부품 종류와 이름
supply_for_sale	plant_name	PK, FK	manufacturing_plants 참조
	plant_type		suppliers 참조
	supplier_name		
	part_type		
	manufacture_date		부품 제조 날짜
manufacturing_plants	plant_name	PK	제조 공장 이름
	plant_type		제조 공장 타입 (조립 또는 부품 공급)
	company_name	FK	company 참조
customers	customer_name	PK	고객 이름
	phone_number		고객 번호
	c_address		고객 주소
	annual_income		고객 연간 수입
person_customers	customer_name	PK, FK	customers 참조
	phone_number		
	gender		사람 고객 성별
company_customers	customer_name	PK, FK	customers 참조
	phone_number		
	type		기업 고객의 기업 타입

- Relationships

관계 이름	설명
brands_to_dealers	one to zero one or more – 브랜드에 대응되는 딜러가 없을 수도 있고 여러 개일수도 있다. Non-Identifying – 비식별 관계이다. No Nulls – 브랜드에 Null을 허용하지 않는다.
brands_to_models	one to zero one or more – 브랜드에 대응되는 모델이

	<p>없을 수도 있고 여러 개일수도 있다.</p> <p>Non-Identifying – 비식별 관계이다.</p> <p>No Nulls – 브랜드에 Null을 허용하지 않는다.</p>
company_to_brands	<p>one to zero one or more – 회사에 대응되는 브랜드가 없을 수도 있고 여러 개일수도 있다.</p> <p>Non-Identifying – 비식별 관계이다.</p> <p>No Nulls – 회사에 Null을 허용하지 않는다.</p>
company_to_manufacturing_plants	<p>one to zero one or more – 회사에 대응되는 제조 공장이 없을 수도 있고 여러 개일수도 있다.</p> <p>Non-Identifying – 비식별 관계이다.</p> <p>No Nulls – 회사에 Null을 허용하지 않는다.</p>
customers_to_company_customers	<p>one to zero or one – 고객에 대응되는 회사 고객이 없을 수도 있고 한 개일수도 있다.</p> <p>Identifying – 식별 관계이다.</p>
customers_to_person_customers	<p>one to zero or one – 고객에 대응되는 사람 고객이 없을 수도 있고 한 개일수도 있다.</p> <p>Identifying – 식별 관계이다.</p>
customers_to_vehicles	<p>zero or one to zero one or more – 고객이 없을 수도 있는데 있는 경우에는 대응되는 자동차가 없을 수도 있고 여러 개일수도 있다.</p> <p>Non-Identifying – 비식별 관계이다.</p> <p>Nulls Allowed – 고객에 Null을 허용한다.</p>
dealers_to_vehicles	<p>one to zero one or more – 딜러에 대응되는 자동차가 없을 수도 있고 여러 개일수도 있다.</p> <p>Non-Identifying – 비식별 관계이다.</p> <p>No Nulls – 딜러에 Null을 허용하지 않는다.</p>
manufacturing_plants_to_supply_for_sale	<p>one to zero one or more – 제조 공장에 대응되는 공급점에 판매 이력이 없을 수도 있고 여러 개일수도 있다.</p> <p>Identifying – 식별 관계이다.</p>
manufacturing_plants_to_vehicles	<p>one to zero one or more – 제조 공장에 대응되는 자동차가 없을 수도 있고 여러 개일수도 있다.</p> <p>Non-Identifying – 비식별 관계이다.</p> <p>No Nulls – 제조 공장에 Null을 허용하지 않는다.</p>
models_to_options	<p>one to one or more – 모델에 대응되는 옵션이 한 개일수도 있고 여러 개일수도 있다.</p> <p>Identifying – 식별 관계이다.</p>
models_to_vehicles	<p>one to zero one or more – 모델에 대응되는 자동차가 없을 수도 있고 여러 개일수도 있다.</p>

	Non-Identifying – 비식별 관계이다. No Nulls – 모델에 Null을 허용하지 않는다.
options_to_supply_parts	one to zero one or more – 옵션에 대응되는 부품 공급이 없을 수도 있고 여러 개일수도 있다. Identifying – 식별 관계이다.
options_to_vehicles	one to zero one or more – 옵션에 대응되는 자동차 상품이 없을 수도 있고 여러 개일수도 있다. Non-Identifying – 비식별 관계이다. No Nulls – 옵션에 Null을 허용하지 않는다.
suppliers_to_supply_for_sale	one to zero one or more – 공급점에 제조 공장으로부터의 공급이 없을 수도 있고 여러 개일수도 있다. Identifying – 식별 관계이다.
suppliers_to_supply_parts	one to zero one or more – 공급점에 대응되는 판매 상품을 위한 공급이 없을 수도 있고 여러 개일수도 있다. Identifying – 식별 관계이다.

2.2.2. 릴레이션 스키마로의 축소 과정 설명

- 단순 속성들을 지닌 강성 개체 집합의 표현 : 개체 집합을 스키마로 표현한 다음 관계 집합 스키마와의 조합을 통해 필요한 속성은 추가하였다.

company(company_name, country, established_year, company_phone)

brands(brand_name, established_year, brand_phone, company_name)

models(model_name, body_style, default_price, fuel, brand_name)

dealers(dealer_name, d_address, dealer_phone, brand_name)

vehicles(VIN, sale_price, production_date, on_sale, dealer_name, d_address, model_name, color, engine, transmission, plant_name, plant_type, customer_name, phone_number, purchase_date)

suppliers(supplier_name, part_type)

manufacturing_plants(plant_name, plant_type, company_name)

- 약성 개체 집합의 표현 : 약성 개체 집합이 의존하는 강성 개체 집합의 PK와 식별자를 합성하여 스키마의 PK로 활용하였다.

options(model_name, color, engine, transmission)

- 일반화의 표현 : 상위 개체 집합의 스키마를 생성하고 하위 개체 집합을 위해 각 개체 집합의 속성들과 상위 개체 집합의 PK인 속성들에 대한 속성을 가지는 스키마를 생성했다.

customers(customer_name, phone_number, c_address, annual_income)

person_customers(customer_name, phone_number, gender)

company_customers(customer_name, phone_number, type)

- 관계 집합의 표현 : 관계 대응수에 따라 주키를 설정하여 스키마를 만든 후 스키마의 조합을 통해 중복성을 제거하였다. 지워진 스키마들은 제거된 스키마들이다.

company_brands(brand_name, company_name)

brand_models(model_name, brand_name)

brand_dealers(dealer_name, d_address, brand_name)

dealer_vehicles(VIN, dealer_name, d_address)

option_info(VIN, model_name, color, engine, transmission)

model_info(VIN, model_name)

model_options(model_name, color, engine, transmission)

own(plant_name, plant_type, company_name)

assemble(VIN, plant_name, plant_type)

buy(VIN, customer_name, phone_number, purchase_date)

supply_for_sale(plant_name, plant_type, supplier_name, part_type, manufacture_date)

supply_parts(supplier_name, part_type, model_name, color, engine, transmission, supply_date)

3. Queries 적용

- 질의문들이 작성되어질 수 있는 형태를 대략적으로 나타내보았다.
- 성별에 따른 지난 3년간 브랜드들의 판매 트렌드

```

SELECT c.gender, b.brand_name, v.purchase_date

FROM vehicles AS v

JOIN models AS m ON v.model_name = m.model_name

JOIN brands AS b ON m.brand_name = b.brand_name

JOIN person_customers AS c ON v.customer_name = c.customer_name

                        AND v.phone_number = c.phone_number

WHERE v.on_sale = 'N'

GROUP BY c.gender, b.brand_name

HAVING YEAR(v.purchase_date) >= 2019

```

- 소득 범위에 따른 지난 3년간 브랜드들의 판매 트렌드

```

SELECT c.annual_income, b.brand_name, v.purchase_date

FROM vehicles AS v

JOIN models AS m ON v.model_name = m.model_name

JOIN brands AS b ON m.brand_name = b.brand_name

JOIN customers AS c ON v.customer_name = c.customer_name

                        AND v.phone_number = c.phone_number

WHERE v.on_sale = 'N'

GROUP BY b.brand_name

having c.annual_income < 10000000 AND YEAR(v.purchase_date) >= 2019

```

- Getrag의 공장 중 하나에서 문제가 생겨 공급업체 Getrag가 지정된 기간동안 변속기 결함을 발생시켰다. 이 변속기가 포함된 VIN과 구매 고객

```

SELECT VIN, customer_name, phone_number

FROM vehicles

```


WHERE transmission IN ('t_gt_210403', 't_gt_210404') AND on_sale = 'N'

- 달러 금액 기준 1년간 판매 상위 2개 브랜드

```
SELECT b.brand_name
FROM vehicles AS v
JOIN models AS m ON v.model_name = m.model_name
JOIN brands AS b ON m.brand_name = b.brand_name
WHERE v.on_sale = 'N'
GROUP BY b.brand_name
ORDER BY SUM(v.sale_price) DESC
LIMIT 2
```

- 유닛 기준 1년간 판매 상위 2개 브랜드

```
SELECT b.brand_name
FROM vehicles AS v
JOIN models AS m ON v.model_name = m.model_name
JOIN brands AS b ON m.brand_name = b.brand_name
WHERE v.on_sale = 'N'
GROUP BY b.brand_name
ORDER BY COUNT(v.VIN) DESC
LIMIT 2
```

- 컨버터블이 잘 팔리는 달

```
SELECT MONTH(v.purchase_date) as pc_month, COUNT(VIN)
FROM vehicles AS v
```

```
JOIN models AS m ON v.model_name = m.model_name
```

```
WHERE m.body_style = 'Convertible' AND v.on_sale = 'N'
```

```
GROUP BY pc_month
```

각 월별 컨버터블의 판매량 수가 테이블로 나오게 된다.

- 차량을 가장 오랜 시간동안 재고로 보관하는 딜러점

```
SELECT dealer_name, AVG(production_date) as avg_date
```

```
FROM vehicles
```

```
WHERE on_sale = 'Y' /*판매되지 않은 상품들*/
```

```
GROUP BY dealer_name
```

```
ORDER BY avg_date ASC
```

각 딜러점별 평균 생산 날짜가 오름차순으로 나오는데 production_date는 정수형으로 210414 이런 식으로 저장되어 있었다. 결과가 180401, 190407,... 같은 값이 있는 avg_date 열을 가진 테이블로 나온다.