COOL CARS COMPANY

Navid Hariri, Summeeyah Haq, Juyoung Suh, Junjie Xiao, Shonda O'Conner

INFORMATION SYSTEMS 441 Professor Davoudian

Contents

Introduction

Case Scenario

Cool Car Company is a company that has been around for many years specializing in selling vehicles. However, as the years went on, competition had risen, and other companies were responding faster than Cool Car Company in adopting more recent technologies, which resulted in less errors and cost and higher sales for them. Much of this issue seemed to have been caused by the company's usage of the outdated traditional file processing system. The traditional file processing system has its disadvantages of program data dependence, duplication of data, limited sharing, lengthy development time, and high program maintenance. Having identified this as their primary issue, Cool Car Company went forward to implement a database approach for the company. The database approach has its many advantages to help the company such as program data independence, planned redundancy of data, improved data consistency, better data sharing and more.

Business Needs

With the product line being the place where most companies compete with one another, Cool Car Company finds it crucial to be able to keep track of their revenue and expenses correctly and effortlessly. They are requesting a system to be able to identify the types of customers they have, the salespeople who sell these vehicles, the types of vehicle they sell, and the services provided for those vehicles by mechanics.

Requirements

The management system should deploy a centralized database as a core to store, manage, and operate data. The database itself should be independent which only the database

management system (DBMS) has the accessibility to the database. The DBMS connects with users' applications as well. The entities in the database all should relate to the business. Each entity should have a unique name. The entities in the business should be logically related. The relationship between the entities should represent the exact business activities.

Functions

The system will function to show what types of customers Cool Car Company has and what types of employees are selling or working on their vehicle. Also, the system will give detailed descriptions about the vehicles make, model, year, and cost as well as the orders made by customers. It will also function to show the services provided to each vehicle by the, mechanic

Activities

The database administrator has access to data definition language in order to create, alter, and drop tables. These privileges are granted to the database administrator as he or she would be most knowledgeable about constructing and organizing the database. He and the other developers also have access to the data manipulation commands to insert, update, modify, and query the database. The other developers need to be able to take user data from the frontend and insert, update, or delete records in the database.

Entities

The entities that are needed to create the system are Customer supertype with subtypes of International and Domestic subtypes. Employee supertype with Mechanic and Salesperson subtypes. Vehicle, Order, and Order Line, which will serve as an associative entity between Vehicle and Orders. Service entity will serve as an associative entity of Vehicle and Mechanic, Part, and Supplier.

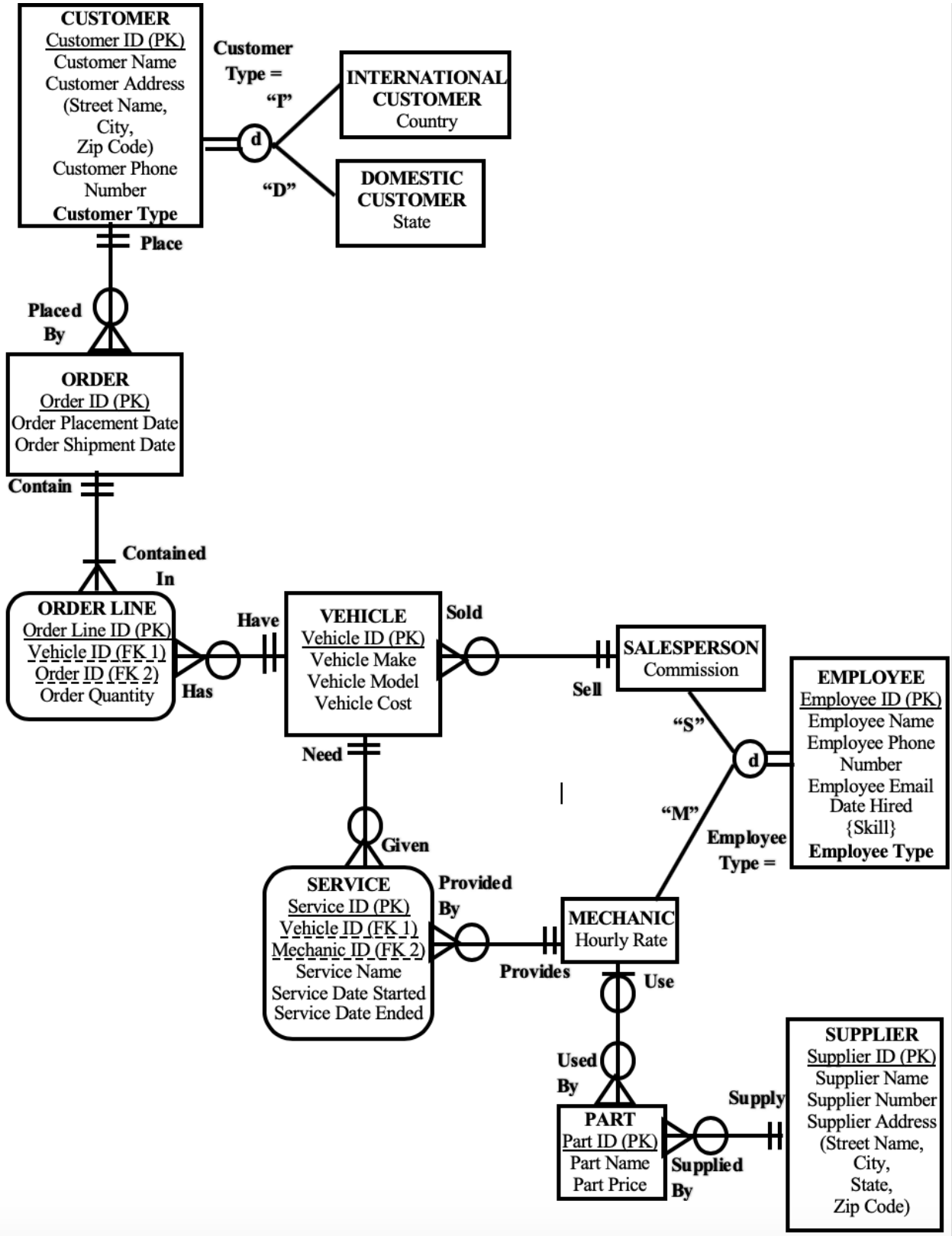
Purpose

The purpose of this technological shift to the database approach is to keep track of revenue and expenses accurately and effortlessly, which will then help Cool Cars Company continue to be competitive in the market and continue to have highly satisfied and happy customers.

Relationship Matrix

	Customer	Order	Order Line	Vehicle	Service	Mechanic	SP	Part	Supplier
Customer	---	**Places**	---	---	---	---	---	---	---
Order	**Placed By**	---	**Contain**	---	---	---	---	---	---
Order Line	---	**Contained In**	---	**Has**	---	---	---	---	---
Vehicle	---	---	**Have**	---	**Need**	---	**Sold**	---	---
Service	---	---	---	**Given**	---	**Provided By**	---	---	---
Mechanic	---	---	---	---	**Provide**	---	---	**Use**	---
SP	---	---	---	**Sell**	---	---	---		---
Part	---	---	---	---	---	**Used By**	---	---	**Supplied By**
Supplier	---	---	---	---	---	---	---	**Supply**	---

Enhanced Entity Relationship Diagram



Business Rule

a. Each CUSTOMER must be a DOMESTIC CUSTOMER or an INTERNATIONAL CUSTOMER but never both at the same time.

b. Each CUSTOMER may place one or more ORDERS**.** Each ORDER is placed by one and only one CUSTOMER.

c. Each ORDER must contain one or more ORDER LINES. Each ORDER LINE must contain in one and only one ORDER.

d. Each ORDER must include one or more VEHICLES**.** Each VEHICLE may be included in one or more orders.

e. Each ORDER LINE has one and only one VEHICLE**.** Each VEHICLE may have one or more ORDER LINES.

f. Each EMPLOYEE must be a SALESPERSON or a MECHANIC but never both at the same time.

g. Each VEHICLE may be worked on by one or more MECHANICS. Each MECHANIC may work on one or more VEHICLE.

h. Each VEHICLE may need many types of SERVICE**.** Each SERVICE must be given to one and only one VEHICLE.

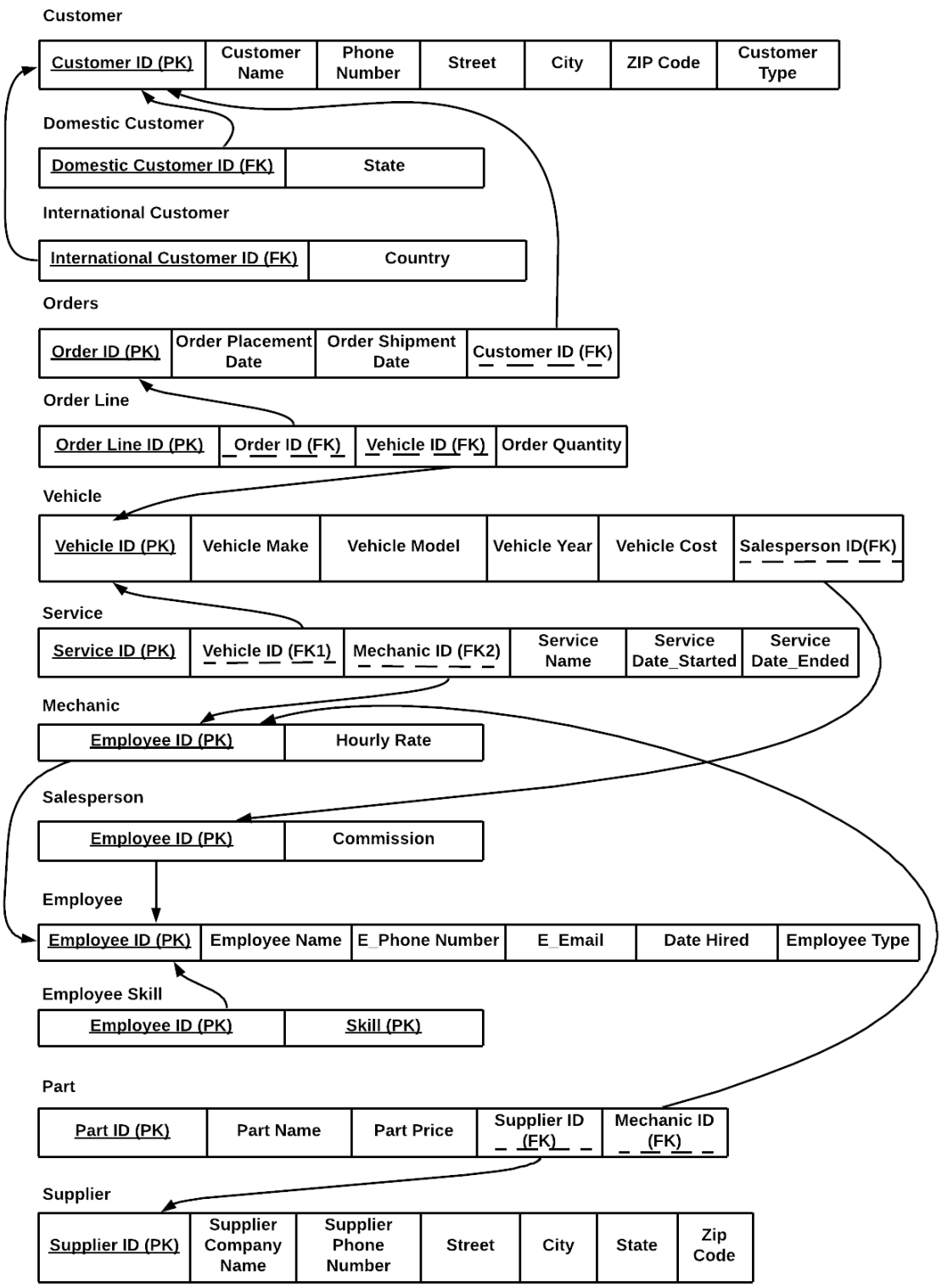
i. Each SERVICE must be provided by one and only one MECHANIC**.** Each MECHANIC may provide many SERVICES.

j. Each SALESPERSON may sell one or more VEHICLE**.** Each VEHICLE must be sold by one SALESPERSON**.**

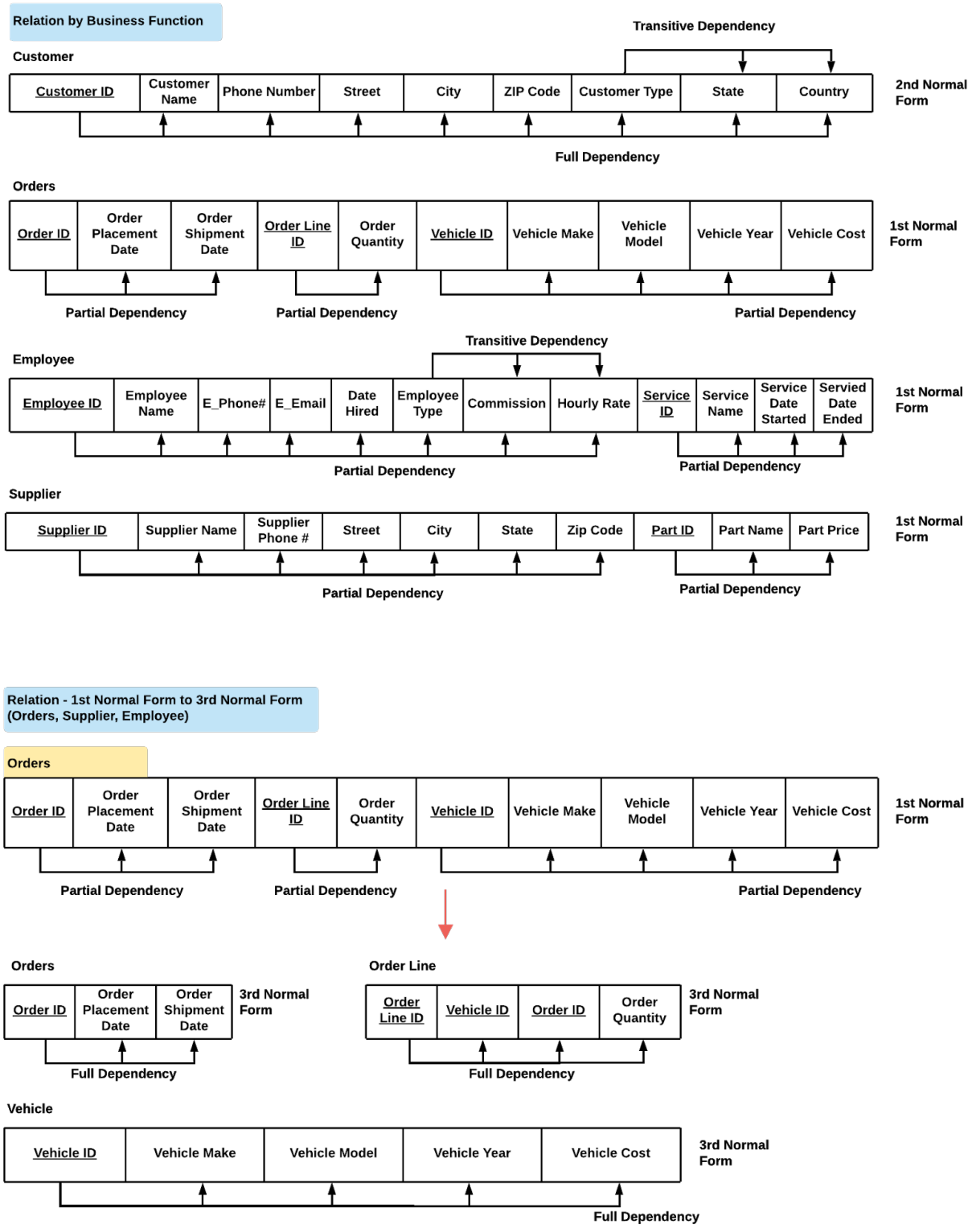
k. Each MECHANIC may use one or more PARTS**.** Each PART may be used by one and only one MECHANIC.

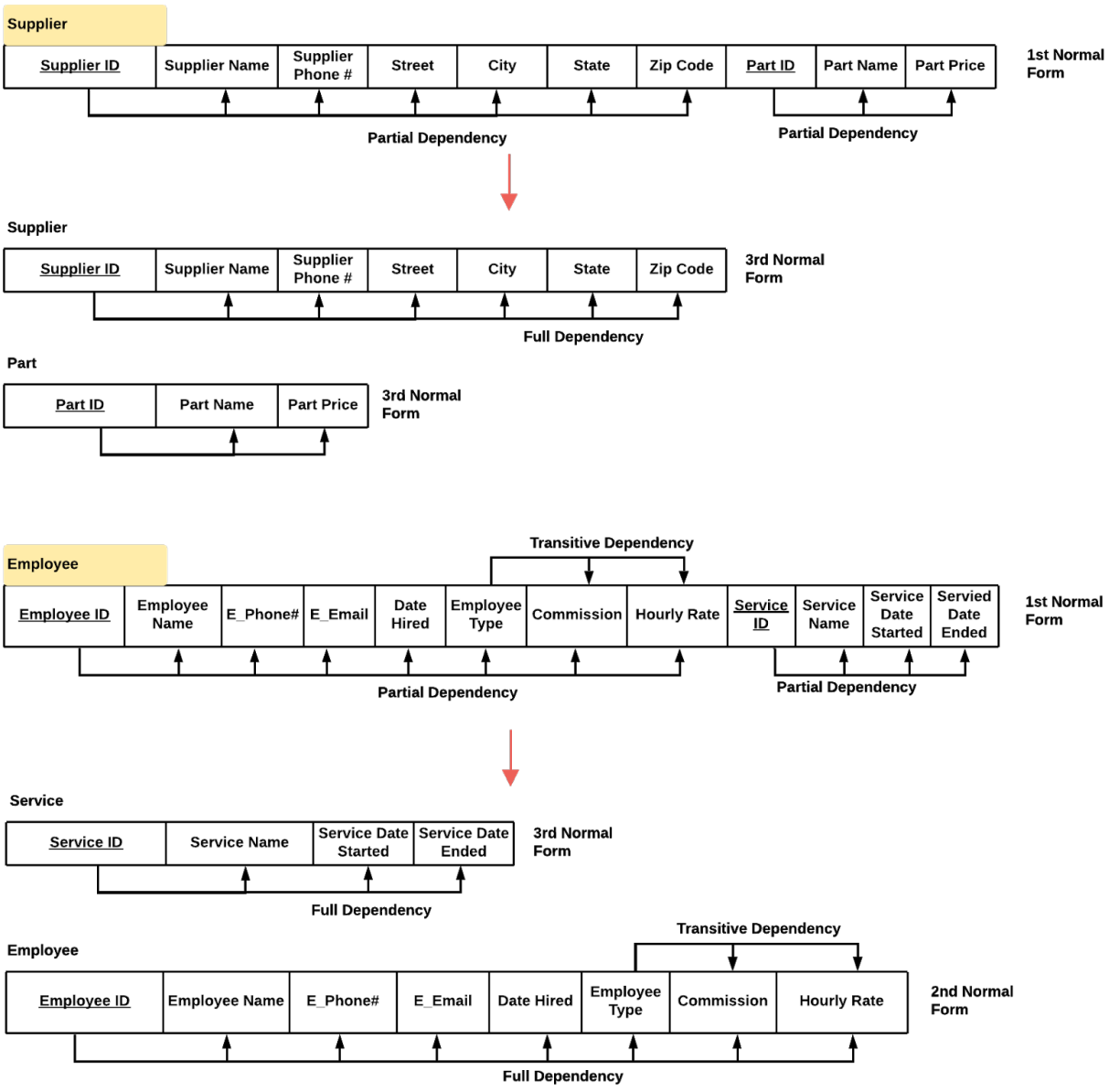
Each PART must be supplied by one and only one SUPPLIER**.** Each SUPPLIER may supply one or more PARTS.

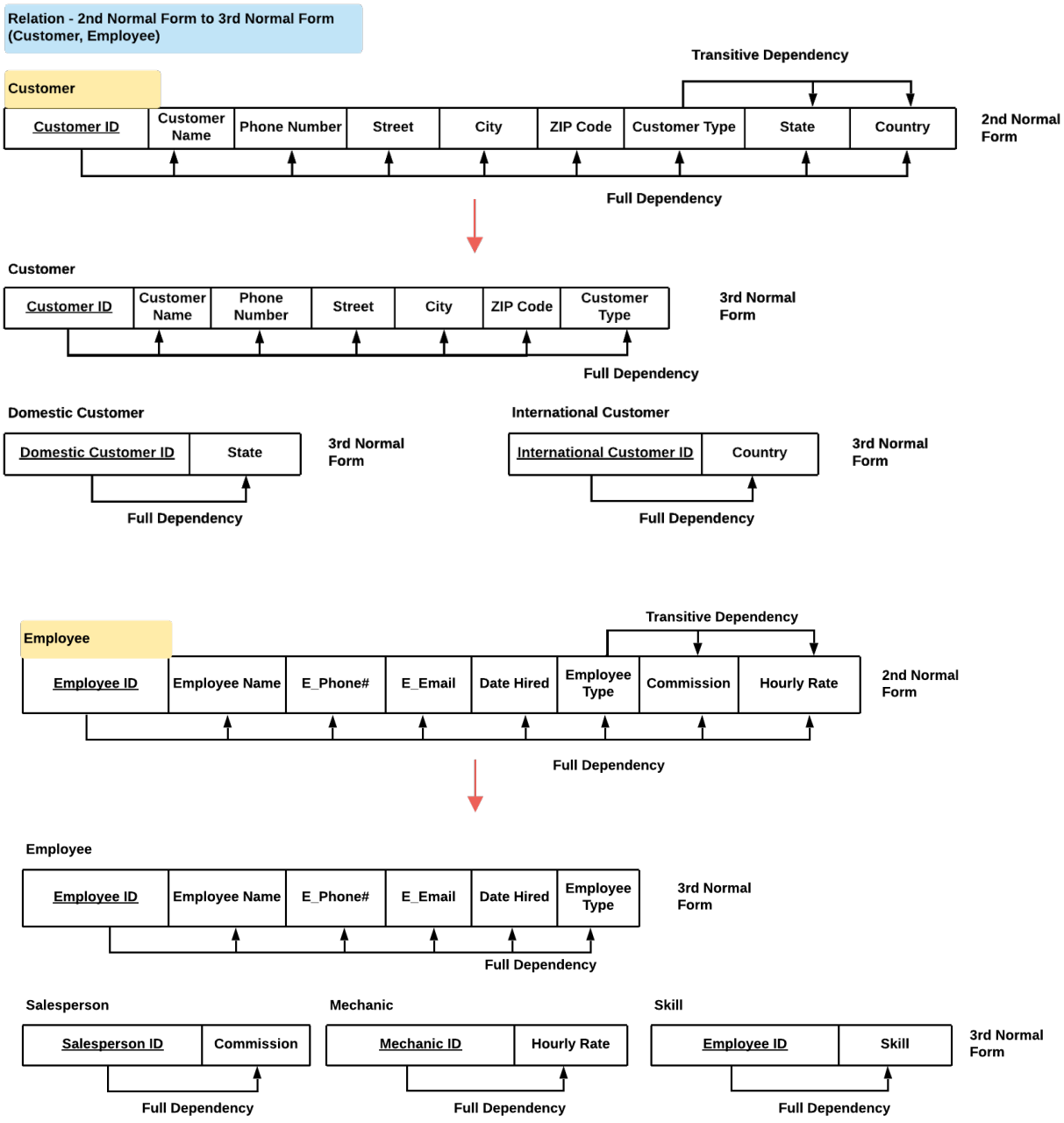
Referential Integrity Constraint



Functional Dependency Diagram and Normalization




Supplier
Supplier ID
Supplier Name
Supplier Phone #
Street
City
State
Zip Code
Part ID
Part Name
Part Price
1st Normal Form
Partial Dependency
Partial Dependency
Supplier
Supplier ID
Supplier Name
Supplier Phone #
Street
City
State
Zip Code
3rd Normal Form
Full Dependency
Part
Part ID
Part Name
Part Price
3rd Normal Form
Transitive Dependency
Employee
Employee ID
Employee Name
E_Phone#
E_Email
Date Hired
Employee Type
Commission
Hourly Rate
Service ID
Service Name
Service Date Started
Servied Date Ended
1st Normal Form
Partial Dependency
Partial Dependency
Service
Service ID
Service Name
Service Date Started
Service Date Ended
3rd Normal Form
Full Dependency
Transitive Dependency
Employee
Employee ID
Employee Name
E_Phone#
E_Email
Date Hired
Employee Type
Commission
Hourly Rate
2nd Normal Form
Full Dependency


Relation - 2nd Normal Form to 3rd Normal Form
(Customer, Employee)
Transitive Dependency
Customer
Customer ID
Customer Name
Phone Number
Street
City
ZIP Code
Customer Type
State
Country
2nd Normal Form
Full Dependency
Customer
Customer ID
Customer Name
Phone Number
Street
City
ZIP Code
Customer Type
3rd Normal Form
Full Dependency
Domestic Customer
Domestic Customer ID
State
3rd Normal Form
Full Dependency
International Customer
International Customer ID
Country
3rd Normal Form
Full Dependency
Transitive Dependency
Employee
Employee ID
Employee Name
E_Phone#
E_Email
Date Hired
Employee Type
Commission
Hourly Rate
2nd Normal Form
Full Dependency
Employee
Employee ID
Employee Name
E_Phone#
E_Email
Date Hired
Employee Type
3rd Normal Form
Full Dependency
Salesperson
Salesperson ID
Commission
Full Dependency
Mechanic
Mechanic ID
Hourly Rate
Full Dependency
Skill
Employee ID
Skill
3rd Normal Form
Full Dependency

Structured Query Language

Creation

Customer

```
-- customer table
CREATE TABLE IF NOT EXISTS customer
(
    customer_id           INTEGER     NOT NULL,
    customer_name         TEXT        NOT NULL,
    customer_phone_number VARCHAR(30) NOT NULL,
    customer_street       TEXT        NOT NULL,
    customer_city         TEXT        NOT NULL,
    customer_zip_code     VARCHAR(20) NOT NULL,
    customer_type         TEXT        NOT NULL,
    PRIMARY KEY (customer_id),
    CHECK ( customer_type = 'international' or customer_type = 'domestic')
);
```

Domestic Customer

```
-- domestic customer table subtype of customer
CREATE TABLE IF NOT EXISTS domestic_customer
(
    domestic_customer_id    INTEGER    NOT NULL,
    domestic_customer_state VARCHAR(2) NOT NULL,
    PRIMARY KEY (domestic_customer_id),
    FOREIGN KEY (domestic_customer_id) REFERENCES customer (customer_id)
);
```

International Customer

```
-- international customer table with subtype of customer
CREATE TABLE IF NOT EXISTS international_customer
(
    international_customer_id      INTEGER NOT NULL,
    international_customer_country TEXT    NOT NULL,
    PRIMARY KEY (international_customer_id),
    FOREIGN KEY (international_customer_id) REFERENCES customer (customer_id)
);
```

Employee

```
-- create employee table
CREATE TABLE IF NOT EXISTS employee
(
    employee_ID           INTEGER     NOT NULL,
    employee_name         TEXT        NOT NULL,
    employee_phone_number VARCHAR(30) NOT NULL,
    employee_email        TEXT        NOT NULL,
    employee_date_hired   DATE DEFAULT CURRENT_DATE NOT NULL,
    employee_type         TEXT        NOT NULL,
    PRIMARY KEY (employee_ID),
    CHECK ( employee_type = 'sales' or employee_type = 'mechanic')
);
```

Mechanic

```
-- create mechanic table subtype of employee
CREATE TABLE IF NOT EXISTS mechanic
(
    employee_id          INTEGER NOT NULL,
    mechanic_hourly_rate FLOAT   NOT NULL,
    PRIMARY KEY (employee_id),
    FOREIGN KEY (employee_id) REFERENCES employee (employee_ID)
);
```

Sales Person

```
-- sales person table subtype of employee
CREATE TABLE IF NOT EXISTS sales_person
(
    employee_id             INTEGER NOT NULL,
    sales_person_commission FLOAT   NOT NULL,
    PRIMARY KEY (employee_id),
    FOREIGN KEY (employee_id) REFERENCES employee (employee_ID)
);
```

Employee Skill

```
-- employee skill multi valued table
CREATE TABLE IF NOT EXISTS employee_skill
(
    employee_id    INTEGER NOT NULL,
    employee_skill TEXT    NOT NULL,
    PRIMARY KEY (employee_id, employee_skill),
    FOREIGN KEY (employee_id) REFERENCES employee (employee_ID)
);
```

Order

```
-- orders table
CREATE TABLE IF NOT EXISTS orders
(
    order_id             INTEGER NOT NULL,
    order_placement_date DATE DEFAULT CURRENT_DATE NOT NULL,
    order_shipment_date  DATE,
    customer_id          INTEGER NOT NULL,
    primary key (order_id),
    FOREIGN KEY (customer_id) REFERENCES customer (customer_id)
);
```

Vehicle

```
-- vehicle table
CREATE TABLE IF NOT EXISTS vehicle
(
    vehicle_id      INTEGER NOT NULL,
    vehicle_make    TEXT    NOT NULL,
    vehicle_model   TEXT    NOT NULL,
    vehicle_year    INT     NOT NULL,
    vehicle_cost    FLOAT   NOT NULL,
    sales_person_id INTEGER,
    PRIMARY KEY (vehicle_id),
    FOREIGN KEY (sales_person_id) REFERENCES sales_person (employee_id)
);
```

Order Line

```
-- order line table
CREATE TABLE IF NOT EXISTS order_line
(
    order_line_id  INTEGER NOT NULL,
    order_id       INTEGER NOT NULL,
    vehicle_id     INTEGER NOT NULL,
    order_quantity INT     NOT NULL,
    PRIMARY KEY (order_line_id),
    FOREIGN KEY (order_id) REFERENCES orders (order_id),
    FOREIGN KEY (vehicle_id) REFERENCES vehicle (vehicle_id)
);
```

Service

```
-- service table
CREATE TABLE IF NOT EXISTS service
(
    service_id           INTEGER NOT NULL,
    vehicle_id           INTEGER NOT NULL,
    service_mechanic_id  INTEGER NOT NULL,
    service_name         TEXT    NOT NULL,
    service_date_started DATE DEFAULT CURRENT_DATE NOT NULL,
    service_date_end     DATE,
    PRIMARY KEY (service_id),
    FOREIGN KEY (vehicle_id) REFERENCES vehicle (vehicle_id),
    FOREIGN KEY (service_mechanic_id) REFERENCES mechanic (employee_id)
);
```

Supplier

```
-- supplier table
CREATE TABLE IF NOT EXISTS supplier
(
    supplier_id           INTEGER     NOT NULL,
    company_name          TEXT        NOT NULL,
    supplier_phone_number VARCHAR(30) NOT NULL,
    supplier_street       TEXT        NOT NULL,
    supplier_city         TEXT        NOT NULL,
    supplier_state        VARCHAR(2),
    supplier_zip_code     INT         NOT NULL,
    PRIMARY KEY (supplier_id)
);
```

Part

```
-- part table
CREATE TABLE IF NOT EXISTS part
(
    part_id     INTEGER NOT NULL,
    part_name   TEXT    NOT NULL,
    part_price  FLOAT   NOT NULL,
    supplier_id INTEGER NOT NULL,
    mechanic_id INTEGER NOT NULL,
    primary key (part_id),
    FOREIGN KEY (supplier_id) REFERENCES supplier (supplier_id),
    FOREIGN KEY (mechanic_id) REFERENCES mechanic (employee_id)
);
```

Insertion

Customer

```
INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                     customer_type)
VALUES ('Aaron Fuller', 7383670449, 'Pennsylvania Avenue', 19002, 'Grant',
'domestic');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                     customer_type)
VALUES ('Antonio Gross', 6674435742, 'Main customer_street South', 20744,
'Higganum', 'domestic');
```

```
INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Celia Young', 4238425859, 'Church customer_street', 30741, 'Wood
River', 'domestic');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Wilson Goodwin', 5548581699, 'Fairview Avenue', 'Roy', 45840,
'domestic');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Leo Gill', 7184528143, 'Wall customer_street', 'Montross', 07026,
'domestic');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Reginald Hogan', 2833381181, 'Hudson customer_street', 'Marco
Island', 11542, 'international');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Elijah Joseph', 9239375532, 'Highland Avenue', 'Virgelle', 27526,
'international');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Katie Brady', 3958198655, 'Dogwood Drive', 'Bienville', 16506,
'international');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Mona Drake', 2179153374, 'Heritage Drive', 'Carthage', 08527,
'international');

INSERT INTO customer(customer_name, customer_phone_number, customer_street,
customer_city, customer_zip_code,
                    customer_type)
VALUES ('Derrick Summers', 9768634277, '4th customer_street', 'Floyd', 07726,
'international');
```

Domestic Customer

```
INSERT INTO domestic_customer(domestic_customer_id, domestic_customer_state)
VALUES (1, 'CA');

INSERT INTO domestic_customer(domestic_customer_id, domestic_customer_state)
```

```
VALUES (2, 'NY');

INSERT INTO domestic_customer(domestic_customer_id, domestic_customer_state)
VALUES (3, 'NV');

INSERT INTO domestic_customer(domestic_customer_id, domestic_customer_state)
VALUES (4, 'MA');

INSERT INTO domestic_customer(domestic_customer_id, domestic_customer_state)
VALUES (5, 'CA');
```

International Customer

```
INSERT INTO international_customer(international_customer_id,
international_customer_country)
VALUES (6, 'Chile');

INSERT INTO international_customer(international_customer_id,
international_customer_country)
VALUES (7, 'France');

INSERT INTO international_customer(international_customer_id,
international_customer_country)
VALUES (8, 'Greece');

INSERT INTO international_customer(international_customer_id,
international_customer_country)
VALUES (9, 'Japan');

INSERT INTO international_customer(international_customer_id,
international_customer_country)
VALUES (10, 'Mexico');
```

Employee

```
INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Abraham Kennedy', 5974784359, 'tmaek@me.com', '2017-04-01',
'sales');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Lorene Cannon', 6128891387, 'wmszeliga@mac.com', '2011-04-01',
'sales');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Ronald Saunders', 7203219940, 'johndo@yahoo.ca', '2012-04-01',
'sales');
```

```
INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Nicole Webb', 3698455909, 'quinn@gmail.com', '2013-04-01', 'sales');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Beatrice Morrison', 3188742111, 'shang@comcast.net', '2017-04-01',
'sales');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Charlene Williams', 6958416474, 'stecoop@optonline.net', '2014-04-
01', 'mechanic');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Arlene Lyons', 6194572578, 'speeves@verizon.net', '2015-04-01',
'mechanic');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Antonia Barton', 4488908506, 'mwitte@optonline.net', '2012-04-01',
'mechanic');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Cristina Chandler', 3754965474, 'squirrel@yahoo.ca', '2014-04-01',
'mechanic');

INSERT INTO employee(employee_name, employee_phone_number, employee_email,
employee_date_hired, employee_type)
VALUES ('Ronnie Ferguson', 2932422652, 'skippy@yahoo.com', '2018-04-01',
'mechanic');
```

Employee Skills

```
INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (1, 'Product Knowledge');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (1, 'Strategic Prospecting');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (1, 'Active Listening');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (2, 'Product Knowledge');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (2, 'Communication');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (2, 'Qualification Questioning');
```

```
INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (3, 'Objection Handling');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (3, 'Strategic Prospecting');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (3, 'Rapport Building on the Call');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (3, 'Qualification Questioning');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (4, 'Product Knowledge');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (4, 'Buyer-Seller Agreement');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (5, 'Objection Handling');
INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (5, 'Objection Prevention');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (5, 'Communication');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (6, 'Diagnostic');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (6, 'Customer service');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (6, 'Problem-solving');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (6, 'Technical aptitude');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (7, 'Customer service');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (7, 'Diagnostic');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (7, 'Technical aptitude');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (7, 'Problem-solving');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (8, 'Customer service');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (8, 'Diagnostic');
```

```
INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (9, 'Problem-solving');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (9, 'Technical aptitude');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (10, 'Customer service');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (10, 'Diagnostic');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (10, 'Technical aptitude');

INSERT INTO employee_skill (employee_id, employee_skill)
VALUES (10, 'Problem-solving');
```

Sales Person

```
INSERT INTO sales_person (employee_id, sales_person_commission)
VALUES (1, 5634.12);

INSERT INTO sales_person (employee_id, sales_person_commission)
VALUES (2, 5799.83);

INSERT INTO sales_person (employee_id, sales_person_commission)
VALUES (3, 8514.22);

INSERT INTO sales_person (employee_id, sales_person_commission)
VALUES (4, 8730.48);

INSERT INTO sales_person (employee_id, sales_person_commission)
VALUES (5, 4355.93);
```

Mechanic

```
INSERT INTO mechanic (employee_id, mechanic_hourly_rate)
VALUES (6, 40.74);

INSERT INTO mechanic (employee_id, mechanic_hourly_rate)
VALUES (7, 37.45);

INSERT INTO mechanic (employee_id, mechanic_hourly_rate)
VALUES (8, 50.87);

INSERT INTO mechanic (employee_id, mechanic_hourly_rate)
VALUES (9, 43.71);

INSERT INTO mechanic (employee_id, mechanic_hourly_rate)
VALUES (10, 30.86);
```

Vehicles

```
INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'TT1', 2015, 32000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'TT2', 2017, 34000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'TT3', 2018, 39000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'ST1', 2014, 42000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'ST2', 2016, 45000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'GT1', 2012, 72000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'GT2', 2014, 112000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'GT3', 2016, 202000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'FF', 2019, 332000);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost)
VALUES ('Cool', 'FFS', 2020, 352000);
INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost,
sales_person_id)
VALUES ('Giorgia', 'NINJA', 2017, 417826.20, 1);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost,
sales_person_id)
VALUES ('Pugito', 'Tumbler', 2018, 145842.20, 2);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost,
sales_person_id)
VALUES ('Cocanita', 'Titan', 2018, 18721.22, 3);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost,
sales_person_id)
VALUES ('EA', 'Cheetah', 2019, 18721.22,3);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost,
sales_person_id)
VALUES ('UBOAT', 'Ship', 2019, 24892.20, 4);

INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost,
sales_person_id)
VALUES ('HIT', 'Rocket', 2019, 24892.20, 4);
```

```
INSERT INTO vehicle (vehicle_make, vehicle_model, vehicle_year, vehicle_cost,
sales_person_id)
VALUES ('SHIP', 'Delta', 2019, 5782.20, 5);
```

Order

```
INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (1, '2018-03-15', '2018-03-17', 1);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (2, '2018-03-15', '2018-03-17', 1);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (3, '2018-04-25', '2018-05-3', 2);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (4, '2018-05-06', '2018-05-06', 3);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (5, '2018-08-29', '2018-09-03', 3);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (6, '2018-010-17', '2018-10-27', 4);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (7, '2018-11-05', '2018-11-07', 5);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (8, '2018-12-05', '2018-12-07', 6);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (9, '2019-04-16', '2019-04-17', 7);

INSERT INTO orders (order_id, order_placement_date, order_shipment_date,
customer_id)
VALUES (10, '2019-04-17', '2019-04-20', 8);
```

Order Line

```
INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (1, 4, 8);
```

```
INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (2, 1, 24);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (3, 1, 32);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (4, 4, 8);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (5, 6, 2);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (6, 2, 40);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (7, 3, 3);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (8, 5, 1);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (9, 3, 5);

INSERT INTO order_line (order_id, vehicle_id, order_quantity)
VALUES (10, 5, 4);
```

Supplier

```
INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('Audi', 6387622055, '71 Livingston St', 'Charlotte', 'NC', 28205);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('Bosche', 7017801479, '87 Oak Valley Road', 'Brainerd', 'MN', 56401);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('Titan', 9468600631, '7897 Roosevelt St.', 'Massapequa', 'NY',
11758);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('YOKOHOMA', 5543431211, '784 High Noon St.', 'Solon', 'OH', 44139);
```

```
INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('YAMAZAKI', 9297542285, '7922 Roosevelt Ave.', 'Ridgewood', 'NJ',
07450);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('S1 moto sport club', 3887319886, '8904 Armstrong Lane', 'Montclair',
'NJ', 07042);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('Bazuka', 5262312635, '44 Morris Ave', 'Westport', 'CT', 06880);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('AMD', 6999955820, '935 St Margarets Drive', 'Millington', 'TN',
38053);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('GRID', 3964893814, '8877 Snake Hill supplier_street', 'Fall River',
'MA', 02720);

INSERT INTO supplier (company_name, supplier_phone_number, supplier_street,
supplier_city, supplier_state,
                      supplier_zip_code)
VALUES ('TOYO', 2136244983, '8135 South Ave', 'Ellicott supplier_city', 'MD',
21042);
```

Part

```
INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('Audi sport suspension', 1800, 1, 6);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('Bosch moto oil', 89, 2, 6);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('Drift LSD', 3200, 3, 7);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('performance tires', 1400, 4, 7);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('Anti-roll bar', 900, 5, 8);
```

```
INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('Brambo brake system', 3000, 6, 8);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('B Turbo', 5600, 7, 9);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('ECU unit', 4000, 8, 9);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('Suports package', 7000, 9, 10);

INSERT INTO part (part_name, part_price, supplier_id, mechanic_id)
VALUES ('Grip tires', 900, 10, 10);
```

Service

```
INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (6, 6, 'suspension check', '2019-3-12', '2019-3-12');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (2, 6, 'oil change', '2019-3-21', '2019-3-21');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (3, 7, 'oil change', '2019-3-25', '2019-3-25');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (2, 7, 'air filter upgrade', '2019-4-02', '2019-4-22');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (1, 8, 'coolant change', '2019-4-12', '2019-4-13');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (6, 8, 'transmission replace', '2019-4-15', '2019-4-19');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (1, 9, 'suspension change', '2019-4-22', '2019-4-24');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (4, 9, 'CV joint change', '2019-5-8', '2019-5-12');

INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (5, 10, 'turbo maintenance', '2019-5-18', '2019-5-19');
```

```
INSERT INTO service(vehicle_id, service_mechanic_id, service_name,
service_date_started, service_date_end)
VALUES (3, 10, 'oil change', '2019-6-22', '2019-6-22');
```

Queries and Results

List All Domestic Customers.

```
SELECT *
FROM customer
WHERE customer_type = 'domestic';
```

	customer_id	customer_name	customer_phone_number	customer_street	customer_city	customer_zip_code	customer_type
1	1	Aaron Fuller	7383670449	Pennsylvania Avenue	19002	Grant	domestic
2	2	Antonio Gross	6674435742	Main customer_street South	20744	Higganum	domestic
3	3	Celia Young	4238425859	Church customer_street	30741	Wood River	domestic
4	4	Wilson Goodwin	5548581699	Fairview Avenue	Roy	45840	domestic
5	5	Leo Gill	7184528143	Wall customer_street	Montross	7026	domestic

List all the employee IDs, names, and dates of employees who were hired before the end of May 2015.

```
SELECT employee_id, employee_name, employee_date_hired
FROM employee
WHERE employee_date_hired <= '2015-05-30'
ORDER BY employee_date_hired DESC;
```

	employee_ID	employee_name	employee_date_hired
1	7	Arlene Lyons	2015-04-01
2	6	Charlene Williams	2014-04-01
3	9	Cristina Chandler	2014-04-01
4	4	Nicole Webb	2013-04-01
5	3	Ronald Saunders	2012-04-01
6	8	Antonia Barton	2012-04-01
7	2	Lorene Cannon	2011-04-01

List all employee ID and names of employees who have more than 2 skills.

```
SELECT employee.employee_id, employee.employee_name,
COUNT(employee_skill.employee_id) AS number_of_skills
FROM employee,
     employee_skill WHERE employee_skill.employee_id = employee.employee_id
GROUP BY employee.employee_id
HAVING count(employee_skill.employee_id) > 2;
```

	employee_ID	employee_name	number_of_skills
1	1	Abraham Kennedy	3
2	2	Lorene Cannon	3
3	3	Ronald Saunders	4
4	5	Beatrice Morrison	3
5	6	Charlene Williams	4
6	7	Arlene Lyons	4
7	10	Ronnie Ferguson	4

List all Japanese customers.

```
SELECT customer_id,
       international_customer_country,
       customer_name,
       customer_phone_number,
       customer_street,
       customer_city,
       customer_zip_code,
       customer_type
FROM international_customer
       INNER JOIN customer ON
international_customer.international_customer_id = customer.customer_id
WHERE international_customer_country = 'Japan';
```

	customer_id	international_customer_country	customer_name	customer_phone_number	customer_street	customer_city	customer_zip_code	customer_type
1	9	Japan	Mona Drake	2179153374	Heritage Drive	Carthage	8527	international

List customer ID, customer name, customer type, and how many orders each customer placed.

```
SELECT customer.customer_id, customer_name, customer_type,
COUNT(orders.customer_id) AS order_number
FROM customer
LEFT JOIN orders
ON customer.customer_id = orders.customer_id
GROUP BY customer.customer_id;
```

	customer_id	customer_name	customer_type	order_number
1	1	Aaron Fuller	domestic	2
2	2	Antonio Gross	domestic	1
3	3	Celia Young	domestic	2
4	4	Wilson Goodwin	domestic	1
5	5	Leo Gill	domestic	1
6	6	Reginald Hogan	international	1
7	7	Elijah Joseph	international	1
8	8	Katie Brady	international	1
9	9	Mona Drake	international	0
10	10	Derrick Summers	international	0

Which vehicle ID, make, model, year, are priced above the average cost of our cars, and how much are they?

```
SELECT vehicle_id, vehicle_make, vehicle_model, vehicle_year, vehicle_cost
FROM vehicle
WHERE vehicle_cost >
        (
          SELECT AVG(vehicle_cost)
          FROM vehicle
        );
```

	vehicle_id	vehicle_make	vehicle_model	vehicle_year	vehicle_cost
1	8	Cool	GT3	2016	202000.0
2	9	Cool	FF	2019	332000.0
3	10	Cool	FFS	2020	352000.0
4	11	Giorgia	NINJA	2017	417826.2
5	12	Pugito	Tumbler	2018	145842.2

Find out vehicle make, model which were made after 2015, and all the service info from the vehicle and service table.

```
SELECT vehicle.vehicle_make, vehicle.vehicle_model, vehicle.vehicle_year,
service.*
FROM vehicle
INNER JOIN service ON vehicle.vehicle_id = service.vehicle_id
WHERE vehicle_year > '2015'
ORDER BY service.vehicle_id;
```

	vehicle_make	vehicle_model	vehicle_year	service_id	vehicle_id	service_mechanic_id	service_name	service_date_started	service_date_end
1	Cool	TT2	2017	2	2	6	oil change	2019-3-21	2019-3-21
2	Cool	TT2	2017	4	2	7	air filter upgrade	2019-4-02	2019-4-22
3	Cool	TT3	2018	3	3	7	oil change	2019-3-25	2019-3-25
4	Cool	TT3	2018	10	3	10	oil change	2019-6-22	2019-6-22
5	Cool	ST2	2016	9	5	10	turbo maintenance	2019-5-18	2019-5-19

Which mechanic has an hourly rate of higher than $40, and works with parts that cost less than $3,000 and what are those parts called? List Mechanic employee ID and hourly rate.

```
SELECT employee_id, mechanic_hourly_rate, part_name, part_price
FROM mechanic
        LEFT JOIN part ON employee_id = mechanic_id
WHERE mechanic_hourly_rate > 40
AND employee_id =
        (
        SELECT mechanic_id
        FROM part
        WHERE part_price < 3000
        )
ORDER BY employee_id;
```

	employee_id	mechanic_hourly_rate	part_name	part_price
1	6	40.74	Audi sport suspension	1800.0
2	6	40.74	Bosch moto oil	89.0

Find out the sales person's average rate of commission of total vehicle cost.

```
SELECT AVG(sales_person_commission) / AVG(vehicle_cost)
FROM sales_person, vehicle;
```

	AVG(sales_person_commission) / AVG(vehicle_cost)
1	0.0585390590718573

How many cars are priced less than the average price of our cars?

```
SELECT COUNT(*)
FROM vehicle
WHERE vehicle_cost <
        (
          SELECT AVG(vehicle_cost)
          FROM vehicle
        )
```

	cars_priced_less_than_average
1	12

What percentage of cars are priced less than the average price? Round to 2 decimal places.

```
SELECT (ROUND(((CAST((SELECT COUNT(*)
FROM vehicle
WHERE vehicle_cost < (SELECT AVG(vehicle_cost) FROM vehicle)) AS FLOAT) /
                CAST((SELECT COUNT(*) FROM vehicle) AS FLOAT))) * 100, 2)) AS
cars_priced_less_than_average;
```

	cars_priced_less_than_average
1	70.59

List the customer name, customer type, customer phone number and order number for all customers. Include customer information for customer that don't have an order.

```
SELECT customer.customer_id, customer_name, customer_phone_number,
customer_type,
order_id
FROM customer LEFT OUTER JOIN orders
ON customer.customer_id = orders.customer_id
```

	customer_id	customer_name	customer_phone_number	customer_type	order_id
1	1	Aaron Fuller	7383670449	domestic	1
2	1	Aaron Fuller	7383670449	domestic	2
3	2	Antonio Gross	6674435742	domestic	3
4	3	Celia Young	4238425859	domestic	4
5	3	Celia Young	4238425859	domestic	5
6	4	Wilson Goodwin	5548581699	domestic	6
7	5	Leo Gill	7184528143	domestic	7
8	6	Reginald Hogan	2833381181	international	8
9	7	Elijah Joseph	9239375532	international	9
10	8	Katie Brady	3958198655	international	10
11	9	Mona Drake	2179153374	international	NULL
12	10	Derrick Summers	9768634277	international	NULL

Show the information required to make an invoice for order number 10.

```
SELECT customer.customer_id, customer_name, customer_street, customer_city,
customer_zip_code, orders.order_id, order_placement_date, vehicle_make,
vehicle_model,
vehicle_year, vehicle_cost
FROM customer, orders, vehicle
WHERE orders.customer_id = customer.customer_id
AND orders.order_id = vehicle.vehicle_id
AND orders.order_id = 10
```

	customer_id	customer_name	customer_street	customer_city	customer_zip_code	order_id	order_placement_date	vehicle_make	vehicle_model	vehicle_year	vehicle_cost
1	8	Katie Brady	Dogwood Drive	Bienville	16506	10	2019-04-17	Cool	FFS	2020	352000.0

Show all customers who have placed an order.

```
SELECT customer_name
FROM customer
WHERE customer_id IN
        (SELECT DISTINCT customer_id from orders)
```

	customer_name
1	Aaron Fuller
2	Antonio Gross
3	Celia Young
4	Wilson Goodwin
5	Leo Gill
6	Reginald Hogan
7	Elijah Joseph
8	Katie Brady

What is the name and address for the customer who placed order number 6?

```
SELECT customer_name, customer_street, customer_zip_code
FROM customer
WHERE customer.customer_id =
        (SELECT customer_id FROM orders WHERE order_id = 6)
```

	customer_name	customer_street	customer_zip_code
1	Wilson Goodwin	Fairview Avenue	45840

Show orders for vehicle id 2, display Order Id and Order Total.

```
SELECT order_line.order_id, sum (vehicle_cost*order_quantity) AS order_otal
FROM vehicle INNER JOIN order_line
ON order_line.vehicle_id = vehicle.vehicle_id
WHERE order_line.order_id
IN (SELECT order_line.order_id FROM order_line
WHERE order_line.vehicle_id = 2)
GROUP BY order_line.order_id
```

	order_id	order_total
1	6	1360000.0