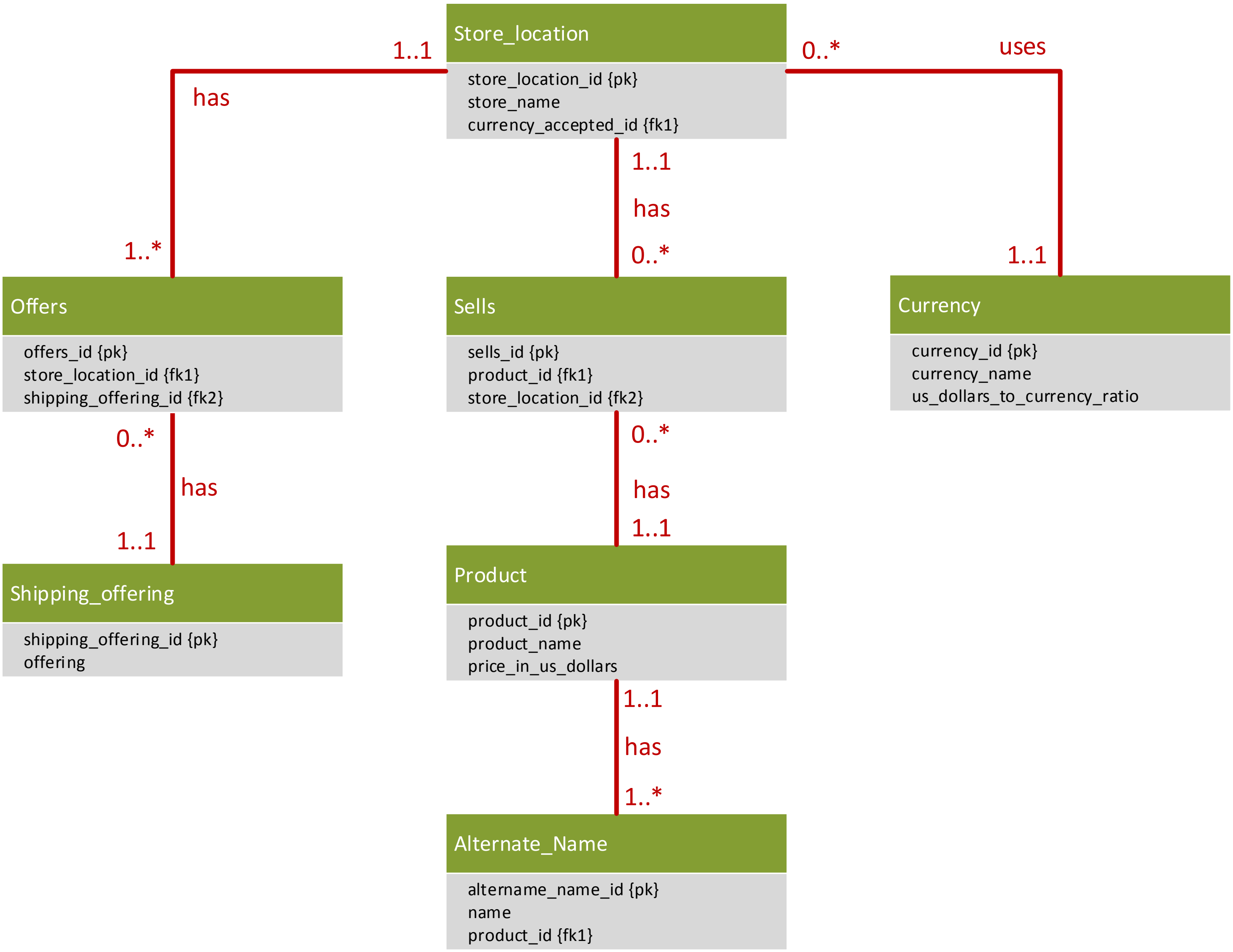
Section One – Subqueries

Section Background

In this section, you will practice crafting subqueries for the schema illustrated below.



This schema's structure supports basic medical product and currency information for an international medical supplier, including store locations, the products they sell, shipping offerings, the currency each location accepts, as well as conversion factors for converting from U.S. dollars into the accepted currency. Due to the specific and technical nature of the names of medical products, the supplier also keeps a list of alternative names for each product that may help customers identify them. This schema models prices and exchange rates at a specific point in time. While a real-world schema would make provision for changes to prices and exchange rates over time, the tables needed to support this have been intentionally excluded from our schema, because their addition would add unneeded complexity on your journey of learning subqueries, expressions, and value manipulation. The schema has just the right amount of complexity for your learning.

The data for the tables is listed below.

Currencies

Name	Ratio
British Pound	0.67
Canadian Dollar	1.34
US Dollar	1.00
Euro	0.92
Mexican Peso	16.76

Store Locations

Name	Currency
Berlin Extension	Euro
Cancun Extension	Mexican Peso
London Extension	British Pound
New York Extension	US Dollar
Toronto Extension	Canadian Dollar

Product

Name	US Dollar Price
Glucometer	$50
Bag Valve Mask	$25
Digital Thermometer	$250
Electronic Stethoscope	$350
Handheld Pulse Oximeter	$450

Sells

Store Location	Product
Berlin Extension	Glucometer
Berlin Extension	Bag Valve Mask
Berlin Extension	Digital Thermometer
Berlin Extension	Handheld Pulse Oximeter
Cancun Extension	Bag Valve Mask
Cancun Extension	Digital Thermometer

Cancun Extension	Handheld Pulse Oximeter
London Extension	Glucometer
London Extension	Bag Valve Mask
London Extension	Digital Thermometer
London Extension	Electronic Stethoscope
London Extension	Handheld Pulse Oximeter
New York Extension	Glucometer
New York Extension	Bag Valve Mask
New York Extension	Digital Thermometer
New York Extension	Electronic Stethoscope
New York Extension	Handheld Pulse Oximeter
Toronto Extension	Glucometer
Toronto Extension	Bag Valve Mask
Toronto Extension	Digital Thermometer
Toronto Extension	Electronic Stethoscope
Toronto Extension	Handheld Pulse Oximeter

Shipping_offering

Offering
Same Day
Overnight
Two Day

Offers

Store Location	**Shipping Offering**
Berlin Extension	Two Day
Cancun Extension	Two Day
London Extension	Same Day
London Extension	Overnight
London Extension	Two Day
New York Extension	Overnight
New York Extension	Two Day
Toronto Extension	Two Day

Alternate Names

Name	**Product**
Glucose Meter	Glucometer
Blood Glucose Meter	Glucometer
Glucose Monitoring System	Glucometer
Thermometer	Digital Thermometer

Ambu Bag	Bag Valve Mask
Oxygen Bag Valve Mask	Oxygen Bag Valve Mask
Cardiology Stethoscope	Electronic Stethoscope
Portable Pulse Oximeter	Handheld Pulse Oximeter
Handheld Pulse Oximeter System	Handheld Pulse Oximeter

The DDL and DML to create and populate the tables in the schema are listed below. You can copy and paste this into your SQL client to create and populate the tables.

```
DROP TABLE Sells;
DROP TABLE Offers;
DROP TABLE Store_location;
DROP TABLE Alternate_name;
DROP TABLE Product;
DROP TABLE Currency;
DROP TABLE Shipping_offering;

CREATE TABLE Currency (
currency_id DECIMAL(12) NOT NULL PRIMARY KEY,
currency_name VARCHAR(255) NOT NULL,
us_dollars_to_currency_ratio DECIMAL(12,2) NOT NULL);

CREATE TABLE Store_location (
store_location_id DECIMAL(12) NOT NULL PRIMARY KEY,
store_name VARCHAR(255) NOT NULL,
currency_accepted_id DECIMAL(12) NOT NULL);

CREATE TABLE Product (
product_id DECIMAL(12) NOT NULL PRIMARY KEY,
product_name VARCHAR(255) NOT NULL,
price_in_us_dollars DECIMAL(12,2) NOT NULL);

CREATE TABLE Sells (
sells_id DECIMAL(12) NOT NULL PRIMARY KEY,
product_id DECIMAL(12) NOT NULL,
store_location_id DECIMAL(12) NOT NULL);

CREATE TABLE Shipping_offering (
shipping_offering_id DECIMAL(12) NOT NULL PRIMARY KEY,
offering VARCHAR(255) NOT NULL);

CREATE TABLE Offers (
offers_id DECIMAL(12) NOT NULL PRIMARY KEY,
store_location_id DECIMAL(12) NOT NULL,
shipping_offering_id DECIMAL(12) NOT NULL);

CREATE TABLE Alternate_name (
alternate_name_id DECIMAL(12) NOT NULL PRIMARY KEY,
name VARCHAR(255) NOT NULL,
product_id DECIMAL(12) NOT NULL);

ALTER TABLE Store_location
ADD CONSTRAINT fk_location_to_currency FOREIGN KEY(currency_accepted_id)
REFERENCES Currency(currency_id);
```

```sql
ALTER TABLE Sells
ADD CONSTRAINT fk_sells_to_product FOREIGN KEY(product_id) REFERENCES
Product(product_id);

ALTER TABLE Sells
ADD CONSTRAINT fk_sells_to_location FOREIGN KEY(store_location_id) REFERENCES
Store_location(store_location_id);

ALTER TABLE Offers
ADD CONSTRAINT fk_offers_to_location FOREIGN KEY(store_location_id) REFERENCES
Store_location(store_location_id);

ALTER TABLE Offers
ADD CONSTRAINT fk_offers_to_offering FOREIGN KEY(shipping_offering_id)
REFERENCES Shipping_offering(shipping_offering_id);

ALTER TABLE Alternate_name
ADD CONSTRAINT fk_name_to_product FOREIGN KEY(product_id)
REFERENCES Product(product_id);

INSERT INTO Currency(currency_id, currency_name, us_dollars_to_currency_ratio)
VALUES(1, 'Britsh Pound', 0.67);
INSERT INTO Currency(currency_id, currency_name, us_dollars_to_currency_ratio)
VALUES(2, 'Canadian Dollar', 1.34);
INSERT INTO Currency(currency_id, currency_name, us_dollars_to_currency_ratio)
VALUES(3, 'US Dollar', 1.00);
INSERT INTO Currency(currency_id, currency_name, us_dollars_to_currency_ratio)
VALUES(4, 'Euro', 0.92);
INSERT INTO Currency(currency_id, currency_name, us_dollars_to_currency_ratio)
VALUES(5, 'Mexican Peso', 16.76);

INSERT INTO Shipping_offering(shipping_offering_id, offering)
VALUES (50, 'Same Day');
INSERT INTO Shipping_offering(shipping_offering_id, offering)
VALUES (51, 'Overnight');
INSERT INTO Shipping_offering(shipping_offering_id, offering)
VALUES (52, 'Two Day');

--Glucometer
INSERT INTO Product(product_id, product_name, price_in_us_dollars)
VALUES(100, 'Glucometer', 50);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10000, 'Glucose Meter', 100);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10001, 'Blood Glucose Meter', 100);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10002, 'Glucose Monitoring System', 100);

--Bag Valve Mask
INSERT INTO Product(product_id, product_name, price_in_us_dollars)
VALUES(101, 'Bag Valve Mask', 25);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10003, 'Ambu Bag', 101);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10004, 'Oxygen Bag Valve Mask', 101);

--Digital Thermometer
INSERT INTO Product(product_id, product_name, price_in_us_dollars)
```

```
VALUES(102, 'Digital Thermometer', 250);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10005, 'Thermometer', 102);

--Electronic Stethoscope
INSERT INTO Product(product_id, product_name, price_in_us_dollars)
VALUES(103, 'Electronic Stethoscope', 350);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10006, 'Cardiology Stethoscope', 103);

--Handheld Pulse Oximeter
INSERT INTO Product(product_id, product_name, price_in_us_dollars)
VALUES(104, 'Handheld Pulse Oximeter', 450);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10007, 'Portable Pulse Oximeter', 104);
INSERT INTO Alternate_name(alternate_name_id, name, product_id)
VALUES(10008, 'Handheld Pulse Oximeter System', 104);

--Berlin Extension
INSERT INTO Store_location(store_location_id, store_name, currency_accepted_id)
VALUES(10, 'Berlin Extension', 4);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1000, 10, 100);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1001, 10, 101);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1002, 10, 102);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1003, 10, 104);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
VALUES(150, 10, 52);

--Cancun Extension
INSERT INTO Store_location(store_location_id, store_name, currency_accepted_id)
VALUES(11, 'Cancun Extension', 5);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1004, 11, 101);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1005, 11, 102);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1006, 11, 104);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
VALUES(151, 11, 52);

--London Extension
INSERT INTO Store_location(store_location_id, store_name, currency_accepted_id)
VALUES(12, 'London Extension', 1);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1007, 12, 100);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1008, 12, 101);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1009, 12, 102);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1010, 12, 103);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1011, 12, 104);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
```

```
VALUES(152, 12, 50);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
VALUES(153, 12, 51);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
VALUES(154, 12, 52);

--New York Extension
INSERT INTO Store_location(store_location_id, store_name, currency_accepted_id)
VALUES(13, 'New York Extension', 3);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1012, 13, 100);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1013, 13, 101);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1014, 13, 102);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1015, 13, 103);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1016, 13, 104);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
VALUES(155, 13, 51);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
VALUES(156, 13, 52);

--Toronto Extension
INSERT INTO Store_location(store_location_id, store_name, currency_accepted_id)
VALUES(14, 'Toronto Extension', 2);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1017, 14, 100);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1018, 14, 101);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1019, 14, 102);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1020, 14, 103);
INSERT INTO Sells(sells_id, store_location_id, product_id)
VALUES(1021, 14, 104);
INSERT INTO Offers(offers_id, store_location_id, shipping_offering_id)
VALUES(157, 14, 52);
```

As a reminder, for each step that requires SQL, make sure to capture a screenshot of the command and the results of its execution. *Further, make sure to eliminate unneeded columns from the result set, to name your columns something user-friendly and human readable, and to format any prices as currencies.*

Section Steps

1. *Create Table Structure* – Create the tables in the schema, including all of their columns, datatypes, and constraints, and populate the tables with data. You can do so by executing the DDL and DML above in your SQL client. You only need to capture