Carson Gustavel

Database Systems Assignment 2

Schema:

Pets(PetID, Name, Age, Street#, City, ZipCode, State, TypeofPet)

Owners(OID, LastName, Street#, City, ZipCode, State, Age, AnnualIncome)

Owns(PetID, Year, OID, PetAgeatOwnership, PricePaid)

Likes(PetID, TypeofFood)

Foods(FoodID, Name, Brand, TypeofFood, Price, ItemWeight, ClassofFood)

Purchases(OID, FoodID, PetID, Month, Year, Quantity)

Exercises:

1. List all pets (PetID, Name, TypeofPet) living in Moscow, Idaho and owned by a minor who has no income.

```
WITH MinorOwners AS (
  SELECT OID
  FROM Owners
  WHERE Age < 18 AND AnnualIncome = 0
)
SELECT p.PetID, p.Name, p.TypeofPet
FROM Pets p
JOIN Owns o ON p.PetID = o.PetID
JOIN MinorOwners mo ON o.OID = mo.OID
WHERE p.City = 'Moscow' AND p.State = 'Idaho';
```

2. List all pet owners and their pets (OID, LastName, PetID, Name) who do not live with their pets.

```
SELECT ow.OID, ow.LastName, p.PetID, p.Name
FROM Owners ow
JOIN Owns o ON ow.OID = o.OID
JOIN Pets p ON o.PetID = p.PetID
WHERE NOT (
  ow.Street# = p.Street#
  AND ow.City = p.City
  AND ow.ZipCode = p.ZipCode
  AND ow.State = p.State
);
```

3. List pets (PetID, Name) who never ate the types of food they love.

```
WITH AteTypes AS (
  SELECT DISTINCT p.PetID, f.TypeofFood
  FROM Purchases pu
  JOIN Foods f ON pu.FoodID = f.FoodID
  JOIN Pets p ON pu.PetID = p.PetID
)
SELECT DISTINCT p.PetID, p.Name
FROM Pets p
JOIN Likes l ON p.PetID = l.PetID
WHERE NOT EXISTS (
  SELECT 1
  FROM AteTypes a
  WHERE a.PetID = p.PetID
    AND a.TypeofFood = l.TypeofFood
);
```

4. List the brands and names of the food (PetID, Pets.Name, FoodID, Foods.Name, Brand) a pet could potentially eat if their owners bought for them.

```
WITH OwnerFoods AS (
  SELECT DISTINCT pu.OID, f.TypeofFood, f.FoodID, f.Name AS FoodName, f.Brand
  FROM Purchases pu
  JOIN Foods f ON pu.FoodID = f.FoodID
)
SELECT DISTINCT p.PetID, p.Name AS PetName, of.FoodID, of.FoodName, of.Brand
FROM Pets p
JOIN Owns o ON p.PetID = o.PetID
JOIN OwnerFoods of ON o.OID = of.OID
JOIN Likes l ON p.PetID = l.PetID
WHERE l.TypeofFood = of.TypeofFood;
```

5. List the highest priced food for each brand (FoodID, Brand, Price) without using group by operator.

```
SELECT f1.FoodID, f1.Brand, f1.Price
FROM Foods f1
LEFT JOIN Foods f2
```

```
  ON f1.Brand = f2.Brand AND f1.Price < f2.Price
WHERE f2.FoodID IS NULL;
```