

SQL-03 | JOINS

Lecture Queries

Question: Get the report of all the vendors who primarily sell fresh produce and who don't and mark it in front of their names.

Question: Get the report of all the vendors who primarily sell fresh produce and who don't and mark it in front of their names.

```
SELECT
  vendor_id,
  vendor_name,
  vendor_type,
  CASE
    WHEN LOWER(vendor_type) LIKE '%fresh%'
    THEN 'Fresh Produce'
    ELSE 'Other'
  END AS vendor_type_condensed
FROM farmers_market.vendor
```

Question: Let's say we wanted to list each product name along with its product category name.

Question: Let's say we wanted to list each product name along with its product category name.

```
SELECT * FROM  
    product  
LEFT JOIN product_category  
    ON product.product_category_id = product_category.product_category_id
```

With table aliasing:

```
SELECT  
    p.product_id,  
    p.product_name,  
    pc.product_category_id,  
    pc.product_category_name  
FROM product AS p  
    LEFT JOIN product_category AS pc  
    ON p.product_category_id = pc.product_category_id
```

Question: Get all the Customers who have not purchased anything from the market yet.

Question: Get all the Customers who have not purchased anything from the market yet.

```
SELECT c.* # select columns from customer table only
FROM customer AS c
LEFT JOIN customer_purchases AS cp
  ON c.customer_id = cp.customer_id
WHERE cp.customer_id IS NULL
```

Question: Get a list of customers' zip codes who made a purchase on 2019-04-06.

Question: Get a list of customers' zip codes who made a purchase on 2019-04-06.

```
SELECT c.customer_id,  
       c.customer_first_name,  
       c.customer_last_name,  
       cp.vendor_id,  
       c.customer_zip,  
       cp.market_date  
FROM customer c  
RIGHT JOIN customer_purchases cp  
       ON c.customer_id = cp.customer_id  
where market_date = '2019-04-06';
```

Question: Let's say we want details about all booths, as well as every vendor booth assignment along with the vendor details.

Question: Let's say we want details about all farmer's market booths, as well as every vendor booth assignment for every market date.

```
SELECT
    b.booth_number,
    b.booth_type,
    vba.market_date,
    v.vendor_id,
    v.vendor_name,
    v.vendor_type
FROM booth AS b
    LEFT JOIN vendor_booth_assignments AS vba ON b.booth_number = vba.
booth_number
    JOIN vendor AS v ON v.vendor_id = vba.vendor_id
ORDER BY b.booth_number, vba.market_date
```

SELF JOIN - Find all the customers who belong to the same ZIP

```
SELECT A.CustomerName AS CustomerName1,
```

```
B.CustomerName AS CustomerName2, A.Zip
```

```
FROM Customers A, Customers B
```

```
WHERE A.CustomerID <> B.CustomerID
```

```
AND A.Zip = B.Zip;
```

CROSS JOIN

CROSS JOIN is used to combine all possibilities of the two or more tables and returns the result that contains every row from all contributing tables. The CROSS JOIN is also known as CARTESIAN JOIN, which provides the Cartesian product of all associated tables.

