Sprint 02 - SQL for Data Analyst 103

Tools	SQL
Date	@December 8, 2023
Status	Done

▼ EP01 - JOIN using WHERE clause

- Select data from multiple tables using `WHERE`
- Table can join by use PK primary key and FK foreign key

```
SELECT
        A.artistid,
    A.name artistName,
    B.title albumName
from artists A, albums B
WHERE A.artistid = B.artistid and A.name like 'C%';
          PK    =    FK
```

ArtistId	artistName	albumName
16	Caetano Veloso	Prenda Minha
16	Caetano Veloso	Sozinho Remix Ao Vivo
17	Chico Buarque	Minha Historia
18	Chico Science & Nação Zumbi	Afrociberdelia
18	Chico Science & Nação Zumbi	Da Lama Ao Caos

▼ EP02 - Convert WHERE to INNER JOIN

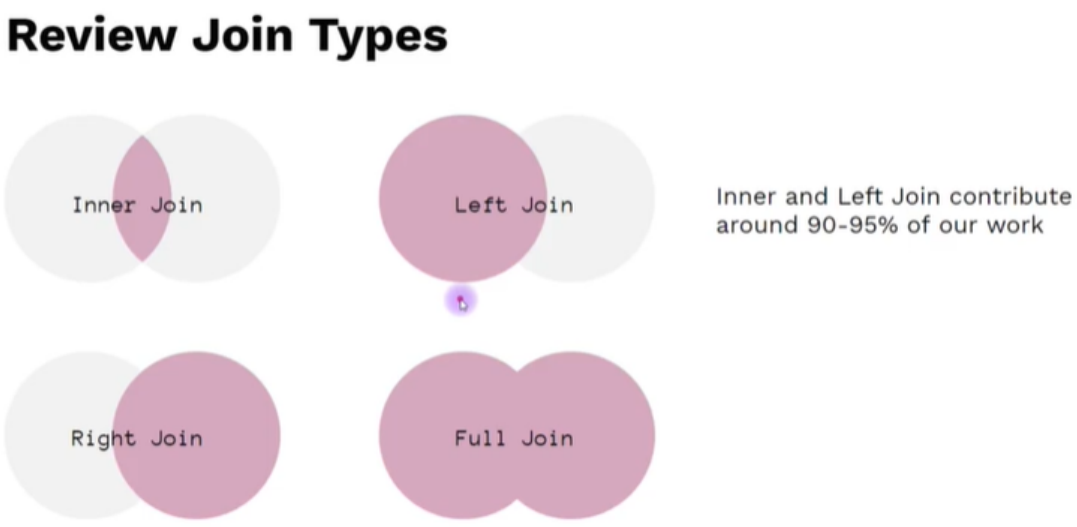
```
--'join >2 tables
-- ดู เพลงของ Aerosmith ทั้งหมด
SELECT
        A.artistid,
    A.name artistName,
    B.title albumName,
    C.name trackname --ชื่อเพลง
from artists A
inner join albums B on A.artistid = B.artistid
inner join tracks C on B.albumid = C.albumid
where A.name = 'Aerosmith' ;
```

ArtistId	artistName	albumName	trackname
3	Aerosmith	Big Ones	Walk On Water
3	Aerosmith	Big Ones	Love In An Elevator
3	Aerosmith	Big Ones	Rag Doll
3	Aerosmith	Big Ones	What It Takes

```
-- นับจำนวนเพลงของ Aerosmith
SELECT
    count(*) Aerosmith_Songs
from artists A
inner join albums B on A.artistid = B.artistid
inner join tracks C on B.albumid = C.albumid
where A.name = 'Aerosmith' ;
```

Aerosmith_Songs
15

▼ EP03 - Review JOIN Concepts



- Inner Join
  ดึงเฉพาะ row ที่ match ค่ากันได้ (Overlap) เท่านั้น
- Left Join
  จะเอา table ซ้ายเป็นตัวตั้งก่อน row ที่ match จะ show ค่า
  ถ้าไม่ match show Null values แทน
- Right Join
  สลับ table ที่จะใช้เป็นตัวตั้ง ส่วน syntax เขียนเหมือนกัน
- Full Join
  จะรวม Values ใน table ทั้งหมด ที่ match ได้และไม่ได้

▼ EP04 - Review CREATE TABLE

```
CREATE TABLE book_shop (
            id INT,
            name TEXT,
            release_year INT
);

CREATE TABLE favourite_book (
            id INT,
            author TEXT,
            reviews REAL -- REAL = ตัวเลขที่มีทศนิยม
);
```

```
INSERT INTO book_shop VALUES
(1, 'Think Like A FREAK', 2014),
(2, 'Ultralearning', 2019),
(3, 'Blue Ocean Strategy', 2015),
(4, 'The Power of Habit', 2012),
(5, 'Outliers', 2008);
```

id	name	release_year
1	Think Like A FREAK	2014
2	Ultralearning	2019
3	Blue Ocean Strategy	2015
4	The Power of Habit	2012
5	Outliers	2008

```
INSERT INTO favourite_book VALUES
(1, 'Steven D. Levitt, Stephen J. Dubner', 1904),
(4, 'Charles Duhigg', 12007),
(5, 'Malcolm Gladwell', 12863);
```

id	author	reviews
1	Steven D. Levitt, Stephen J. Dubner	1904
4	Charles Duhigg	12007
5	Malcolm Gladwell	12863

▼ EP05 - INNER vs. LEFT JOIN

```
-- inner join
SELECT * FROM book_shop A
```

```
INNER JOIN favourite_book B on A.id = B.id;
```

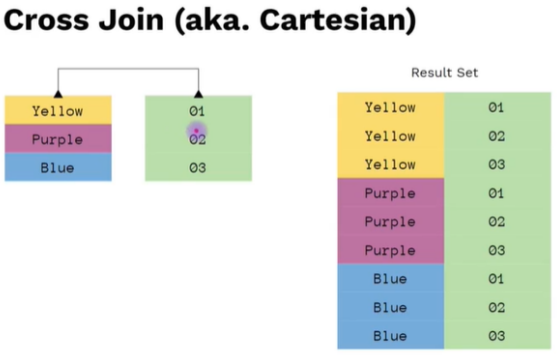
id	name	release_year	id	author	reviews
1	Think Like A F...	2014	1	Steven D. Levit...	1904
4	The Power of H...	2012	4	Charles Duhigg	12007
5	Outliers	2008	5	Malcolm Gladwell	12863

```
--left join
SELECT * FROM book_shop A
LEFT JOIN favourite_book B on A.id = B.id;
```

id	name	release_year	id	author	reviews
1	Think Like A F...	2014	1	Steven D. Levit...	1904
2	Ultralearning	2019	NULL	NULL	NULL
3	Blue Ocean Str...	2015	NULL	NULL	NULL
4	The Power of H...	2012	4	Charles Duhigg	12007
5	Outliers	2008	5	Malcolm Gladwell	12863

▼ EP06 - CROSS JOIN

- Cartesian product (3×3 แถว)



- ไม่ใช้ PK + FK
- สำรับ ไพ่

```
CREATE TABLE ranks (
            rank TEXT
```

```
);

CREATE TABLE suits (
            suits TEXT
);

INSERT INTO ranks
VALUES ('2'),('3'),('4'),('5'),('6'),('7'),('8'),('9'),

INSERT INTO suits
VALUES ('Clubs'),('Diamonds'),('Hearts'),('Spades');
```

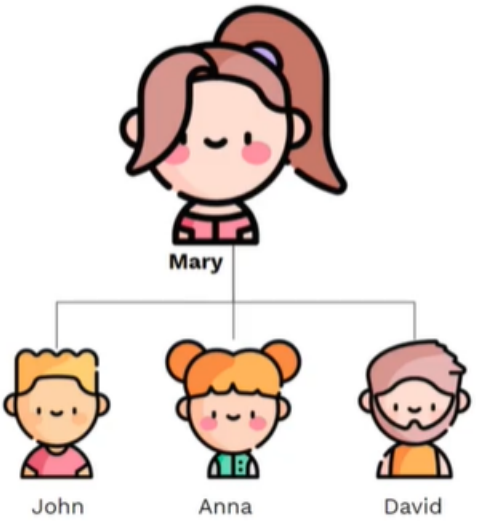
```
-- Cross join
SELECT * FROM ranks, suits Order by suits
                                                =
SELECT * FROM ranks CROSS JOIN suits ORder by suits
```

▼ EP07 - SELF JOIN

Self Join

Table can join itself (self-join)

ID	Name	REPORT_TO
1	Mary	
2	John	1
3	Anna	1
4	David	1



- การ join table ตัวมันเอง
- การดึงชื่อหัวหน้า และ ลูกน้อง ออกมาในครั้งเดียวด้วย Self Join

```
CREATE TABLE employee (
            id INT,
            name TEXT,
            level TEXT,
            manager_id INT
);

INSERT INTO employee VALUES
            (1, 'David', 'CEO', NULL),
            (2, 'John', 'SVP', 1),
            (3, 'Mary', 'VP', 2),
            (4, 'Adam', 'VP', 2),
            (5, 'Scott', 'Manager', 3),
            (6, 'Louise', 'Manager', 3),
            (7, 'Kevin', 'Manager', 4),
            (8, 'Takeshi', 'Manager', 4),
            (9, 'Joe', 'AM', 6),
            (10, 'Ana', 'AM', 7);
```

```
SELECT
        e1.name staff,
    e1.level staff_level,
    e2.name manager,
    e2.level manager_level,
    e1.name || ' reports to ' || e2.name AS comment
FROM employee e1, employee e2
WHERE e1.manager_id = e2.id;
```

staff	staff_level	manager	manager_level	comment
John	SVP	David	CEO	John reports to David
Mary	VP	John	SVP	Mary reports to John
Adam	VP	John	SVP	Adam reports to John
Scott	Manager	Mary	VP	Scott reports to Mary
Louise	Manager	Mary	VP	Louise reports to Mary

▼ EP08 - Intersect and Except

- `Intersect` คล้าย `inner join`
  จะ return row ที่ match กันได้ เท่านั้น
- `Except`
  จะ return row ที่ table ซ้ายมือมี แต่ table ขวามือ ไม่มี

```
CREATE TABLE book_shop (
            id INT,
            name TEXT,
            release_year INT
);

CREATE TABLE favourite_book (
            id INT,
            author TEXT,
            reviews REAL -- REAL = ตัวเลขที่มีทศนิยม
);
```

```
INSERT INTO book_shop VALUES
(1, 'Think Like A FREAK', 2014),
(2, 'Ultralearning', 2019),
(3, 'Blue Ocean Strategy', 2015),
(4, 'The Power of Habit', 2012),
(5, 'Outliers', 2008);
```

```
INSERT INTO favourite_book VALUES
(1, 'Steven D. Levitt, Stephen J. Dubner', 1904),
(4, 'Charles Duhigg', 12007),
(5, 'Malcolm Gladwell', 12863);
```

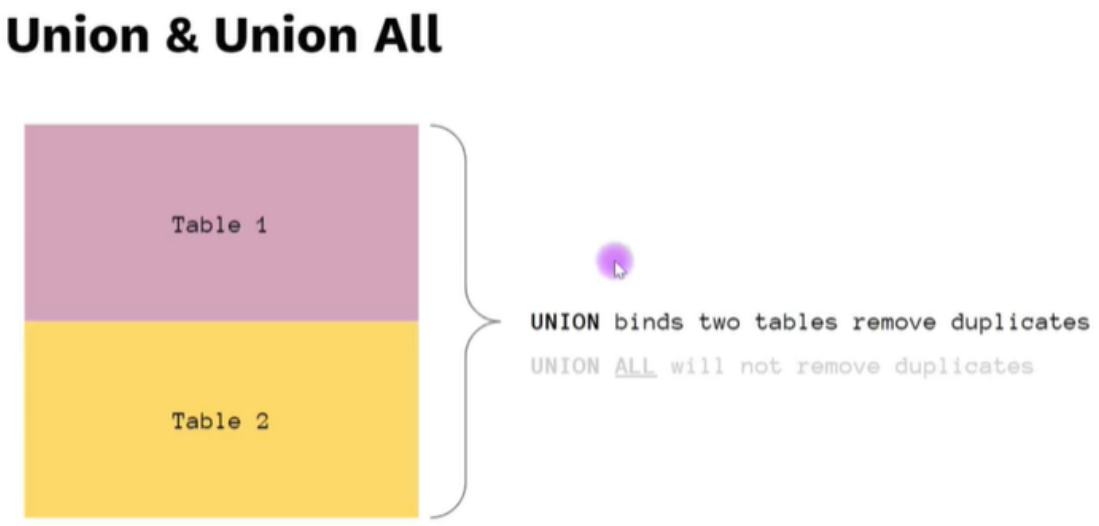
```
-- Intersect
SELECT id FROM book_shop
INTERSECT
```

```
SELECT id FROM favourite_book
Output :  1
                    4
                    5

-- Except
SELECT id FROM book_shop
EXCEPT
SELECT id FROM favourite_book
Output :  2
                    3
```

▼ EP09 - UNION

- การ append 2 table เข้าด้วยกัน
- UNION = ไม่เก็บ value ที่ซ้ำกัน
- UNION ALL = เก็บ value ที่ซ้ำกัน



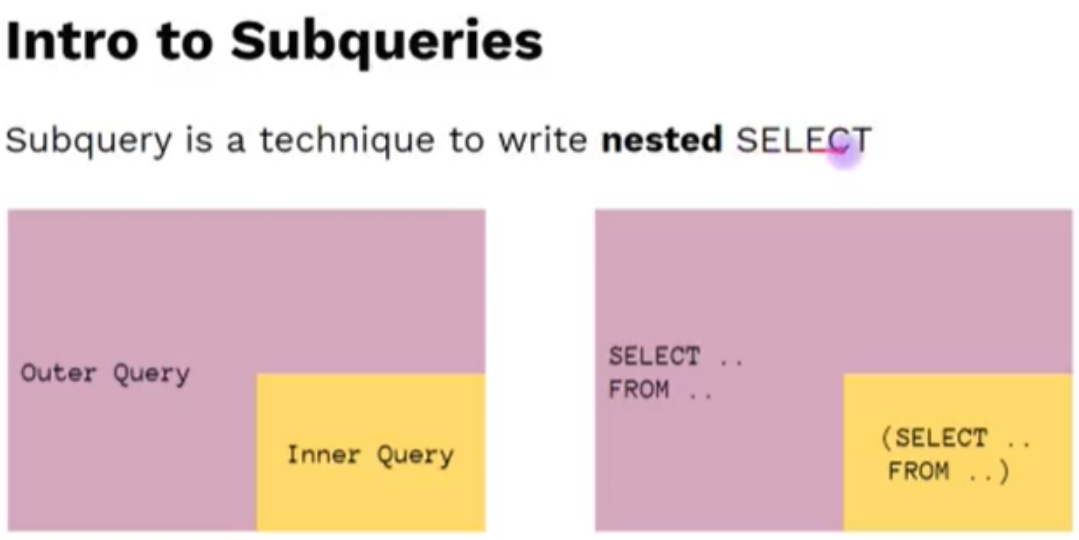

```
-- Create a new book shop table
CREATE TABLE book_shop_new (
            id INT,
            name TEXT,
            release_year INT
);
```

```
INSERT INTO book_shop_new VALUES
            (1, 'Think Like A FREAK', 2014),
            (6, 'Business Data Science', 2020),
            (7, 'Subliminal', 2018),
            (8, 'Good Strategy bad Strategy', 2015);
```

```
-- union two tables
SELECT * FROM book_shop
UNION
SELECT * FROM book_shop_new;

-- union all
SELECT * FROM book_shop
UNION ALL
SELECT * FROM book_shop_new;
```

▼ **EP10 - Intro to Subqueries**




```
-- ชื่อเพลง
SELECT * FROM tracks
WHERE milliseconds = (SELECT MAX(milliseconds)
                      FROM tracks);
```

TrackId	Name	Alb...	Me...	Ge...	Composer	Milliseconds	Bytes
2820	Occupation / Precipice	227	3	19	NULL	5286953	105...

```
-- ชื่อลูกค้า
SELECT firstname, lastname, country FROM
    (SELECT * FROM customers
     WHERE country = 'USA');
```

FirstName	LastName	Country
Frank	Harris	USA
Jack	Smith	USA
Michelle	Brooks	USA
Tim	Goyer	USA