

SQL como lenguaje de consulta de datos

Laboratorio de Base de datos

Cristian E. Sánchez

ORACLE – Consultas

- Iniciar con el primer punto del taller
 - Crear las tablas con sus correspondientes relaciones
 - Insertar los registros entregados por el profesor

ORACLE – SELECT

- Permite obtener la información de una tabla

```
SELECT  
column_1,  
column_2,  
...  
FROM  
table_name;
```

```
Select anio_modelo, placa  
From Autos;
```

ORACLE – JOINS

- Oracle joins se utiliza para combinar columnas de dos o más tablas en función de los valores de las columnas relacionadas.
- Las columnas relacionadas suelen ser la columna de llave principal de la primera tabla y la columna de llave foránea de la segunda tabla.
- Inner join, left join, right join, full outer join y cross join

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

palette_a

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown

palette_b

ORACLE – JOINS

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

palette_a

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown

palette_b

ORACLE – INNER JOIN

SELECT

```
a.id id_a,  
a.color color_a,  
b.id id_b,  
b.color color_b
```

FROM

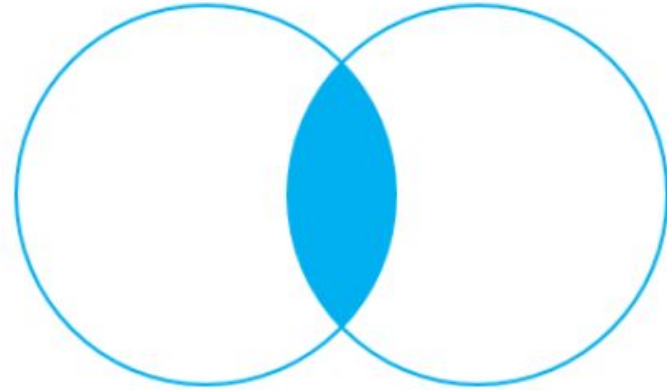
```
palette_a a
```

```
INNER JOIN palette_b b ON a.color = b.color;
```

ID_A	COLOR_A	ID_B	COLOR_B
2	Green	1	Green
1	Red	2	Red

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown



INNER JOIN

ORACLE – LEFT JOIN

SELECT

```
a.id id_a,  
a.color color_a,  
b.id id_b,  
b.color color_b
```

FROM

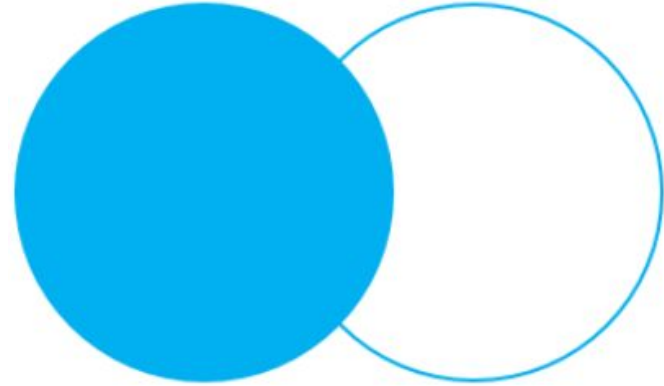
```
palette_a a
```

```
LEFT JOIN palette_b b ON a.color = b.color;
```

ID_A	COLOR_A	ID_B	COLOR_B
2	Green	1	Green
1	Red	2	Red
3	Blue	(null)	(null)
4	Purple	(null)	(null)

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown



LEFT OUTER JOIN

ORACLE – LEFT JOIN

SELECT

a.id id_a,
a.color color_a,
b.id id_b,
b.color color_b

FROM

palette_a a

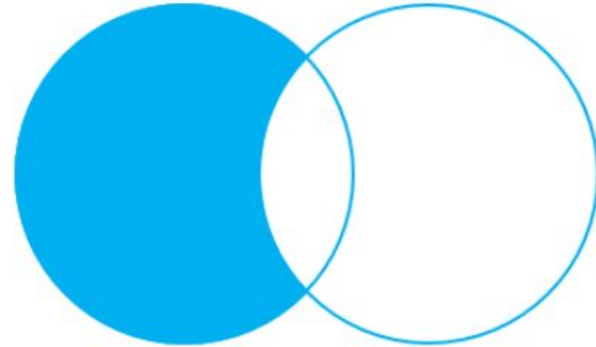
LEFT JOIN palette_b b **ON** a.color = b.color

WHERE b.id **IS NULL**;

ID_A	COLOR_A	ID_B	COLOR_B
3	Blue	(null)	(null)
4	Purple	(null)	(null)

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown



LEFT OUTER JOIN – only
rows from the left table

ORACLE – RIGHT JOIN

SELECT

```
a.id id_a,  
a.color color_a,  
b.id id_b,  
b.color color_b
```

FROM

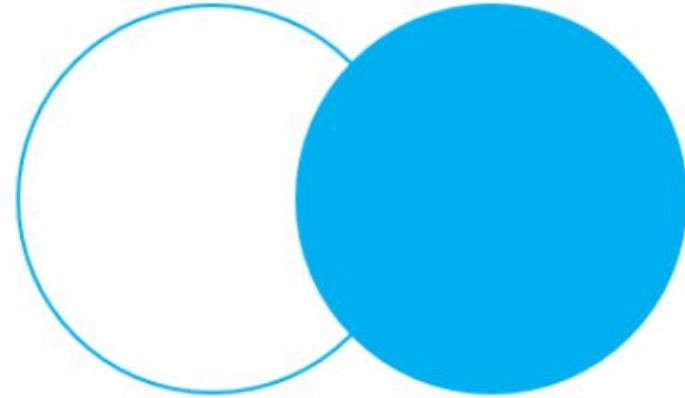
```
palette_a a
```

```
RIGHT JOIN palette_b b ON a.color = b.color;
```

ID_A	COLOR_A	ID_B	COLOR_B
1	Red	2	Red
2	Green	1	Green
(null)	(null)	4	Brown
(null)	(null)	3	Cyan

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown



RIGHT OUTER JOIN

ORACLE – RIGHT JOIN

SELECT

a.id id_a,
a.color color_a,
b.id id_b,
b.color color_b

FROM

palette_a a

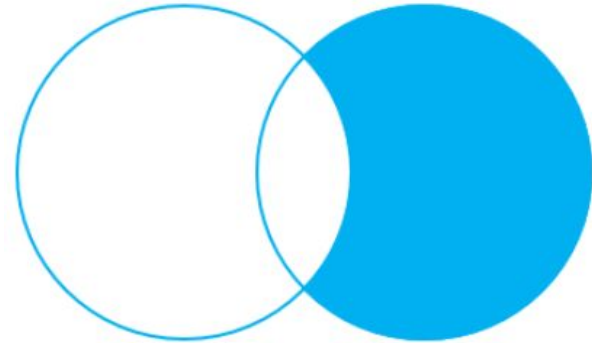
RIGHT JOIN palette_b b **ON** a.color = b.color

WHERE a.id **IS NULL**;

ID_A	COLOR_A	ID_B	COLOR_B
(null)	(null)	4	Brown
(null)	(null)	3	Cyan

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown



RIGHT OUTER JOIN – only
rows from the right table

ORACLE – RIGHT JOIN

SELECT

```
a.id id_a,  
a.color color_a,  
b.id id_b,  
b.color color_b
```

FROM

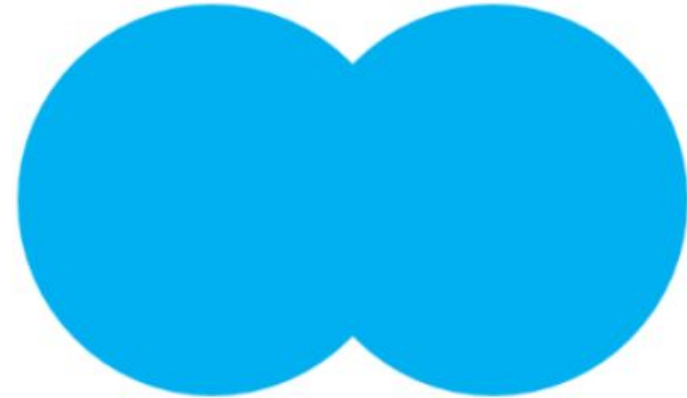
```
palette_a a
```

```
FULL OUTER JOIN palette_b b ON a.color = b.color
```

ID_A	COLOR_A	ID_B	COLOR_B
1	Red	2	Red
2	Green	1	Green
3	Blue	(null)	(null)
4	Purple	(null)	(null)
(null)	(null)	4	Brown
(null)	(null)	3	Cyan

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown



FULL OUTER JOIN

ORACLE – RIGHT JOIN

SELECT

```
a.id id_a,  
a.color color_a,  
b.id id_b,  
b.color color_b
```

FROM

```
palette_a a
```

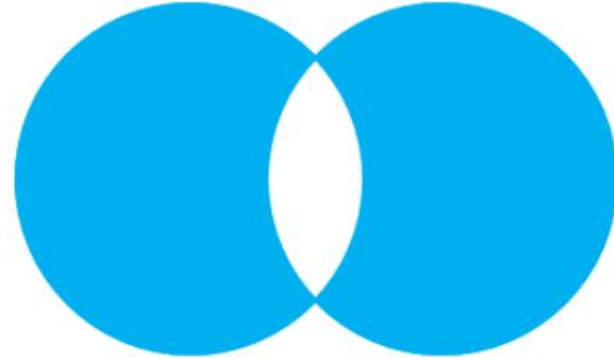
```
FULL JOIN palette_b b ON a.color = b.color
```

```
WHERE a.id IS NULL OR b.id IS NULL;
```

ID_A	COLOR_A	ID_B	COLOR_B
(null)	(null)	3	Cyan
(null)	(null)	4	Brown
3	Blue	(null)	(null)
4	Purple	(null)	(null)

ID	COLOR
1	Red
2	Green
3	Blue
4	Purple

ID	COLOR
1	Green
2	Red
3	Cyan
4	Brown



FULL OUTER JOIN – only
rows unique to both tables

EJERCICIO

Obtener los nombres de las editoriales, que tienen pedidos, junto con la información de sus respectivos pedidos

ORACLE – EXISTS

- Este es un operador Boolean
- Generalmente usado para subconsultas para verificar la existencia de un conjunto de datos (registros)

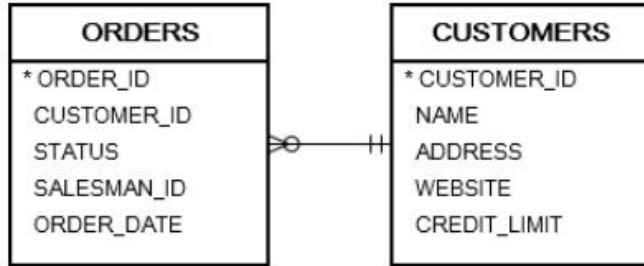
```
SELECT *  
FROM  
  table_name  
WHERE  
  EXISTS(subquery);
```

- Si el subquery devuelve algún registro, entonces **EXISTS** retorna True.
- El operador **EXISTS** termina el proceso del subquery una vez se encuentre el primer registro

ORACLE – EXISTS

Observe que Oracle ignora la lista del subquery. Por lo tanto se puede usar cualquier columna o un valor. En nuestro caso, usamos 1

- Usar el operador **EXISTS** para encontrar todos los clientes que tienen una orden de compra.



```
SELECT
  name
FROM
  customers c
WHERE
  EXISTS (
    SELECT
      1
    FROM
      orders
    WHERE
      customer_id = c.customer_id
  )
ORDER BY
  name;
```

EJERCICIO

Obtener editoriales que **tienen** un pedido

Obtener editoriales que **no tienen** un pedido