

10. Lectura SQL y Álgebra Relacional

SQL (Structured Query Language) es el lenguaje estándar utilizado por los Sistemas de Gestión de Bases de Datos (**DBMS**) para definir, manipular y consultar datos en una base de datos relacional.

Las operaciones de consulta en SQL están estrechamente relacionadas con las operaciones del **álgebra relacional**

Todas las consultas que pueden plantearse con **álgebra relacional** pueden expresarse mediante sentencias **SQL**.

Sintaxis Básica de una Consulta SQL:

Esta estructura se compone de tres cláusulas principales:

Sintaxis.sql

```
1  /* SELECT especifica las columnas que se desean recuperar.*/  
2  SELECT columnas  
3  /*FROM indica las tablas de las cuales se obtendrán los datos.*/  
4  /*WHERE filtra las filas según una condición dada*/  
5  FROM tablas WHERE condición;
```

Esquema de Referencia Utilizado

Para ilustrar la equivalencia entre las operaciones del álgebra relacional y las consultas SQL, se utiliza el siguiente esquema de referencia:

- **Materiales(Clave, Descripción, Precio)**
- **Proveedores(RFC, RazónSocial)**
- **Proyectos(Número, Denominación)**
- **Entregan(Clave, RFC, Número, Fecha, Cantidad)**

Notación Utilizada para el Álgebra Relacional

Para facilitar la comprensión, se emplea la siguiente notación en lugar de las letras griegas tradicionales del álgebra relacional:

- **SL{condición}**: Selección con el criterio especificado.
- **PR{lista de columnas}**: Proyección de las columnas indicadas.
- **JN**: Reunión natural (natural join).
- **JN{condición}**: Reunión con el criterio especificado (theta join).
- **UN**: Unión.
- **IN**: Intersección.
- **-**: Diferencia.
- **X**: Producto cartesiano.

Equivalencias entre Operaciones del Álgebra Relacional y Consultas SQL

A continuación, se presentan ejemplos que ilustran la equivalencia entre las operaciones del álgebra relacional y las consultas SQL:

Consulta de una Relación

- **Álgebra Relacional:**
materiales
- **SQL:**

Consulta.sql

```
1  SELECT * FROM materiales;
```

Selección(σ)

Afecta cardinalidad por que es el numero de duplas

- **Álgebra Relacional:**
 $SL_{\{clave=1000\}}(materiales)$
- **SQL:**

Selección.sql

```
1  SELECT * FROM materiales
2  WHERE clave = 1000;
```

Proyección (π)

- **Álgebra Relacional:**
 $PR_{\{clave, rfc, fecha\}}(entregan)$
- **SQL**

Proyección.sql

```
1  SELECT clave, rfc, fecha FROM entregan;
```

(Natural Join)(\bowtie)

- **Álgebra Relacional:**
Entregan JN proveedores
- **SQL**

Natural_Join.sql

```
1  SELECT * FROM entregan, proveedores
```

```
2 WHERE entregan.rfc = proveedores.rfc;
```

(Theta Join)(θ condición)

- **Álgebra Relacional:**
entregan \Join {entregan.numero \leq proyectos.numero} proyectos
- **SQL:**

Theta_Join.sql

```
1 SELECT * FROM entregan, proyectos
2 WHERE entregan.numero <= proyectos.numero;
```

Unión (\cup)

- **Álgebra Relacional:**
SL{clave=1000}(entregan) \cup SL{clave=2000}(entregan)
- **SQL:**

Unión.sql

```
1 (SELECT * FROM entregan WHERE clave = 1000)
2 UNION
3 (SELECT * FROM entregan WHERE clave = 2000);
```

- **Intersección(\cap)**
Álgebra Relacional: PR{clave}(SL{numero=5001}(entregan)) \cap PR{clave}(SL{numero=5018}(entregan))
- **SQL:**

Intersección.sql

```
1 (SELECT clave FROM entregan WHERE numero = 5001)
2 INTERSECT
3 (SELECT clave FROM entregan WHERE numero = 5018);
```

La operación **INTERSECT** está disponible en sistemas como Oracle, pero no en SQL Server o Access. En estos casos, se pueden utilizar subconsultas para lograr el mismo resultado.

Diferencia ($-$)

- **Álgebra Relacional:**
entregan - SL{clave=1000}(entregan)
- **SQL:**

Diferencia

```
1 (SELECT * FROM entregan)
2 MINUS
```

```
3 (SELECT * FROM entregan WHERE clave = 1000);
```

La operación MINUS** está disponible en Oracle, pero no en SQL Server o Access. Se pueden utilizar subconsultas para obtener resultados equivalentes en estos sistemas.

Producto Cartesiano en SQL (×)

El **producto cartesiano** ocurre cuando se combinan todas las tuplas de dos relaciones, formando una nueva relación con todas las combinaciones posibles.

- **Álgebra Relacional:**
entregan X materiales
- **SQL:**

Producto_Cartesiano.sql

```
1 SELECT * FROM entregan, materiales;  
2
```

Aunque el **producto cartesiano** es una operación fundamental en el álgebra relacional, en SQL rara vez se usa directamente, ya que generalmente se combina con condiciones de unión (JOIN) para evitar combinaciones no deseadas.

Las funciones agregadas son:

SUM(expresión)

Suma de la expresión.

AVG(expresión)

Promedio de la expresión

MIN(expresión)

Mínimo de la expresión

MAX(expresión)

Máximo de la expresión

COUNT(*)

Número de tuplas (renglones)

COUNT(expresión)

número de renglones en los que la expresión es no nula.

STD(expresión)

Desviación estándar de la expresión

Dudas

¿Qué operación del álgebra relacional se usa más en SQL?

¿Cuál es la diferencia entre una **UNIÓN** y un **PRODUCTO CARTESIANO** en SQL?

¿Qué ocurre si olvidamos la condición en un **JOIN** en SQL?

¿Cómo se hace una intersección de conjuntos en SQL si `INTERSECT` no está disponible?

¿Cuál es la diferencia entre `MINUS` y `EXCEPT` en SQL?