

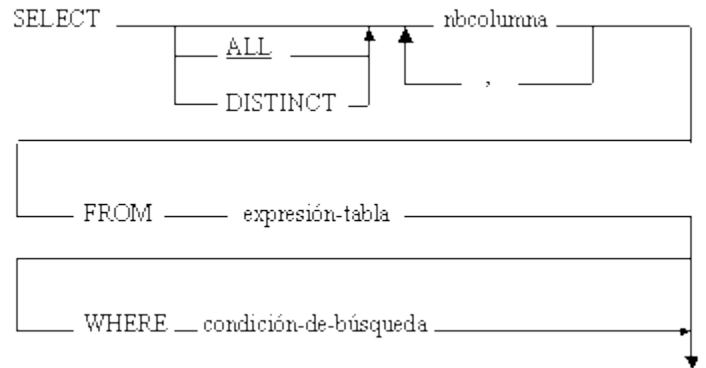
GUIA RÁPIDA DE SQ-1 SINTANXIS

Sintaxis SELECT Simplificada

```

SELECT
  [ALL | DISTINCT ]
  select_expr [, select_expr ...]
  [FROM table_references
  [WHERE where_condition]
  [GROUP BY {col_name | expr | position}
  [ASC | DESC], ... ]
  [HAVING where_condition]
  [ORDER BY {col_name | expr | position}
  [ASC | DESC], ... ]
  [LIMIT { row_count }
  ]
  
```

Diagrama Sintáctico SELECT



- SELECT** se usa para recuperar filas seleccionadas
- Cada **select_expr** indica una columna que se quiere recuperar.
- table_references** indica la tabla o tablas de las que se recuperan filas.
- where_definition** consiste de la palabra clave **WHERE** seguida por una expresión que indica la condición o condiciones que las filas deben satisfacer para ser seleccionadas.

- Las palabras que aparecen en mayúsculas son palabras reservadas (**SELECT, ALL, DISTINCT, FROM, WHERE.**)
- Una sentencia válida se construye siguiendo la línea a través del diagrama hasta el punto que marca el final.
- Las líneas se siguen de izquierda a derecha y de arriba abajo.
- Cuando se quiere alterar el orden normal se indica con una flecha.

(SELECT) Seleccionar todos los registros.

Sintaxis:

SELECT * **FROM** table_name;

Ejemplo:

SELECT * **FROM** estudiante;

Resultado:

codigo	nombre	apellido	sexo	correo	fechaNacimiento
1	JOSE	CASTILLO	M	jose@gmail.com	2000-01-01
2	MARIA	CISNERO	F	maria@gmail.com	2000-01-31

(SELECT) Seleccionar columnas específicas.

Sintaxis:

SELECT column1, column2, ... **FROM** table_name;

Consulta:

SELECT codigo,nombre,sexo **FROM** estudiante;

Resultado:

codigo	nombre	sexo
1	JOSE	M
2	MARIA	F

(DISTINCT) se utiliza para devolver sólo valores distintos.

Sintaxis:

SELECT **DISTINCT** column1, column2, ...

FROM table_name;

Ejemplo:

SELECT **DISTINCT** estado

FROM estudiante;

estado
Aragua
Carabobo
Guarico

(AS) Se utiliza para agregar un nombre temporal a las columnas y tablas.

Sintaxis:

SELECT column_name **AS** alias_name

FROM table_name;

Ejemplo:

```

SELECT
  cedula AS estudiante,
  CONCAT(nombre, ' ', apellido) AS nombre
FROM preparaduria.estudiante;
  
```

estudiante	nombre
25357951	MARIACISNERO
25753951	JOSECASTILLO

(LIMIT) Se utiliza para limitar la cantidad de filas a mostrar.

Sintaxis:

SELECT * **FROM** table_name **LIMIT** row_count;

Ejemplo:

SELECT * **FROM** estudiante **LIMIT** 1;

cedula	nombre	apellido	sexo
25357951	MARIA	CISNERO	F
NULL	NULL	NULL	NULL

(ORDER BY) se utiliza para ordenar el conjunto de resultados en orden ascendente o descendente.

Sintaxis:

SELECT ... **FROM** table_name **ORDER BY** column_name [ASC | DESC];

Ejemplo:

SELECT * **FROM** estudiante **ORDER BY** cedula **DESC**;

cedula	nombre	apellido
25777951	MARCOS	PEREIRA
25753951	JOSE	CASTILLO
25663951	GENESIS	CASTRO

(WHERE) Se utiliza para filtrar registros de acuerdo a un criterio.

Sintaxis:

SELECT ... **FROM** table_name **WHERE** condition;

Ejemplo:

SELECT *
FROM estudiante
WHERE sexo = "M";

cedula	nombre	apellido	sexo
25753951	JOSE	CASTILLO	M
25777951	MARCOS	PEREIRA	M

Operadores en la cláusula WHERE

Operador	Descripcion
=	Igual
<>	No es igual. Nota: En algunas versiones de SQL este operador puede escribirse como !=
>	Mayor que
<	Menos de
>=	Mayor o igual
<=	Menor o igual
BETWEEN	Entre un rango inclusivo
LIKE	Buscar un patrón
IN	Para especificar varios valores posibles para una columna

(LIKE) buscar un patrón específico en una columna.

Sintaxis:

SELECT... **FROM** table_name **WHERE** columnN **LIKE** pattern;

Comodines:

(%) El signo de porcentaje representa cero, uno, o varios caracteres

(_) El subrayado representa un solo carácter

Ejemplo1: Nombres que empiecen por "M" sin importar la cantidad de caracteres que le siguen.

SELECT * **FROM** estudiante

WHERE nombre **LIKE** "M%";

cedula	nombre	apellido
25357951	MARIA	CISNERO
25777951	MARCOS	PEREIRA
NULL	NULL	NULL

Ejemplo2: Nombres que empiecen por "A" y le sigan solo 2 caracteres. (3 caracteres en total)

SELECT * **FROM** estudiante **WHERE** nombre

LIKE "A__";

cedula	nombre	apellido	sexo
25757988	ANA	GUTIERREZ	F
NULL	NULL	NULL	NULL

(IN) permite especificar varios valores en una cláusula WHERE.

Sintaxis:

SELECT column_name(s) **FROM** table_name
WHERE column_name **IN** (value1, value2, ...);

Ejemplo:

SELECT cedula,nombre,estado **FROM** estudiante
WHERE estado **IN** ('Aragua','Carabobo');

cedula	nombre	estado
25357951	MARIA	Aragua
25663951	GENESIS	Carabobo

(BETWEEN) selecciona los valores dentro de un rango determinado (números, textos, fechas)

Sintaxis:

SELECT column_name **FROM** table_name
WHERE column_name **BETWEEN** value1 **AND** value2;

Ejemplo:

SELECT cedula,nombre,fechaNacimiento
FROM estudiante **WHERE** fechaNacimiento
BETWEEN '1990-01-01' **AND** '2000-12-31';

cedula	nombre	fechaNacimiento
25357951	MARIA	2000-01-31
25663951	GENESIS	2000-01-02