

CONCEPTOS GENERALES DE SQL

Operador	Definición	Sentencia
DISTINCT	Elimina tuplas repetidas en el resultado	<pre>SELECT DISTINCT oficio FROM empleado;</pre> <p>Muestra una sola vez cada uno de los oficios de los empleados.</p>
BETWEEN	Cuando el dominio del predicado pertenece a un rango de valores	<pre>SELECT codigo_c, nombre FROM empleado WHERE salario BETWEEN 10000 AND 16000;</pre> <p>Muestra el código y el nombre de los empleados con salario entre 10000 y 16000 pesos inclusive.</p>
Cambio de nombre	Cuando se desea cambiar el nombre de las tablas en los productos cartesianos. Se emplea AS o se deja un espacio entre el nombre de la tabla y su alias	<pre>SELECT * FROM alumno A, materia M WHERE A.legajo= M.legajo;</pre> <p>Renombra a la tabla alumno con A y a la tabla materia con M.</p>
ORDER BY	Permite ordenar el resultado de la consulta a través de los atributos designados	<pre>SELECT * FROM empleado WHERE salario = 15000 ORDER BY nombre;</pre> <p>Muestra los datos de los empleados que cumplen la condición ordenados de manera ascendente por su nombre. Para ordenarlos de manera descendente se debe agregar DESC. Se puede indicar más de un criterio de ordenación, estos van separados por coma.</p>
LIKE	Cuando necesitamos buscar tuplas que "contengan" determinada información, sin necesidad de coincidir exactamente. Utiliza un comodín → %	<pre>SELECT * FROM empleado WHERE nombre LIKE "M%";</pre> <p>Muestra los datos de los empleados cuyo nombre comienza con M. Formatos: "Ma%" → comienza con Ma "%Ma%" → contiene Ma "%Ma" → termina con Ma</p>

CONCEPTOS GENERALES DE SQL

Funciones de agregación:

Son funciones que operan sobre un conjunto de tuplas de entrada y producen un único valor de salida. SQL define como funciones de agregación a las siguientes:

AVG	Retorna como resultado el promedio del atributo indicado para todas las tuplas del conjunto. El atributo debe ser de tipo numérico	<pre>SELECT AVG(salario) FROM empleado;</pre> Muestra el salario promedio de los empleados.
COUNT	Retorna como resultado la cantidad de tuplas involucradas en el conjunto de entrada	<pre>SELECT COUNT(*) FROM empleado WHERE depto_nro = 24;</pre> Muestra la cantidad de empleados que tiene el depto. 24.
MAX	Retorna como resultado el valor más grande dentro del conjunto de tuplas para el atributo indicado	<pre>SELECT MAX(salario) FROM empleado;</pre> Muestra el empleado que tiene el salario más alto.
MIN	Retorna como resultado el valor más pequeño dentro del conjunto de tuplas para el atributo indicado	<pre>SELECT MIN(comision) FROM empleado;</pre> Muestra la comisión más pequeña entre los empleados.
SUM	Retorna como resultado la suma del valor del atributo indicado para todas las tuplas del conjunto. El atributo debe ser de tipo numérico	<pre>SELECT SUM(salario) FROM empleado;</pre> Muestra la suma de todos los salarios de los empleados.

Funciones de Agrupación:

Son funciones que permiten generar subconjuntos de tuplas para obtener información precisa de cada grupo

GROUP BY	Permite agrupar las tuplas de una consulta por algún criterio para obtener un resultado	<pre>SELECT A.nombre, COUNT(A.legajo) FROM alumno A, inscripcion I WHERE A.legajo = I.legajo GROUP BY A.nombre;</pre> Muestra la cantidad de inscripciones que tiene cada uno de los alumnos.
----------	---	---

HAVING	Indica la condición de filtro que debe respetar un grupo. Dentro de esta cláusula se pueden utilizar funciones de agregación	<pre>SELECT A.nombre, nombre_materia FROM alumno A, examen E, materia M WHERE A.legajo = E.legajo AND E.idMateria = M.idMateria GROUP BY A.nombre, nombre_materia HAVING AVG(E.nota) > 6 ;</pre> Muestra por alumno el nombre de las materias cuyo promedio sea mayor a 6.
--------	--	---

Operaciones de pertenencia a conjuntos

Se emplean cuando se utilizan Subconsultas, o sea la consulta SQL contiene otra consulta dentro de ella

IN	<p>Permite comprobar si un elemento es parte de un conjunto. Para cada tupla del resultado se verifica que se encuentre en el resultado de la subconsulta</p>	<pre>SELECT legajo FROM alumno WHERE idCarrera IN (SELECT idCarrera FROM carrera WHERE duracion = 5);</pre> <p>La subconsulta retorna todos los idCarrera correspondientes a las carreras que tengan 5 años de duración. Para cada tupla de la tabla alumno se verifica si el idCarrera del alumno está dentro de ese conjunto de idCarreras que arroja la subconsulta. Si está el legajo del alumno es parte del resultado.</p>
NOT IN	<p>Permite comprobar si un elemento no es parte de un conjunto. Para cada tupla del resultado se verifica que no se encuentre en el resultado de la subconsulta</p>	<pre>SELECT legajo FROM alumno WHERE idcarrera NOT IN (SELECT idcarrera FROM carrera WHERE duracion = 5);</pre> <p>En este caso los legajo del resultado NO están dentro de los resultados de la subconsulta.</p>

Operaciones con Valores nulos

IS NULL IS NOT NULL	<p>Cuando un dominio de un atributo puede tener valores nulos incorpora al conjunto de valores posibles el valor NULL. Este valor se almacena por defecto. Entonces para preguntar en una consulta por este valor de dominio se utilizan estos operadores</p> <pre>SELECT nombre FROM empleado WHERE idlocalidad IS NULL;</pre>
------------------------	---

Operaciones de Insertar, Borrar y Modificar	
INSERTAR	<p>La cláusula utilizada para agregar tuplas a una tabla es INSERT INTO</p> <pre>INSERT INTO alumno(legajo, nombre, apellido) VALUES (123,"Maria","Lopez"), (234,"Mirta","Hilario");</pre> <pre>INSERT INTO alumno VALUES (123,"Maria","Lopez"), (234,"Mirta","Hilario");</pre> <p>Ambas instrucciones son válidas, la primera contiene los atributos de la tabla alumno, y la segunda no. Se debe tener cuidado en el segundo formato de respetar el ingreso de los datos según el orden de los atributos dados en la creación de la tabla.</p> <p>Si el dato es auto_incremet y no se especifica la lista de atributos antes del Values, de debe dejar el lugar correspondiente a ese dato</p>
BORRAR	<p>La cláusula utilizada es DELETE FROM</p> <pre>DELETE FROM alumno WHERE idLocalidad = (SELECT idlocalidad FROM localidad WHERE nombre = "Martinez");</pre> <p>Elimina de la tabla alumno todos los alumnos de la localidad de Martinez</p>
MODIFICAR	<p>La cláusula utilizada para modificar es UPDATE SET</p> <pre>UPDATE carrera SET duracion = 6 WHERE nombre LIKE "Ingeniería%";</pre> <p>Modifica la duración de las carreras que comienzan con Ingeniería y las lleva a 6 años.</p>