Ingeniería en Sistemas ula

Bases de datos

Lenguajes de consulta

SQL

Ingeniería en Sistemas ula

Características adicionales de SQL

- Técnicas para especificar **consultas complejas**: consultas anidadas, funciones agregadas, agrupamiento, consultas recursivas y reunión natural de relaciones
- **Vistas**, **disparadores** y **aserciones**
- Algunos SGBD en su implementación de SQL aún contienen comandos para **crear índices**.

Ingeniería en Sistemas ula

Comparaciones con valores nulos

- Un valor nulo tiene 3 interpretaciones:
  - El valor del atributo es **desconocido**
  - El valor de atributo **no está disponible**
  - El valor del atributo **no es aplicable a esa tupla**

- SQL **no** distingue entre estos significados.
- SQL usa una lógica de 3 valores: *Verdadero, Falso* y *Desconocido*

Lógica de tres valores

AND	Verdadero	Falso	Desconocido
Verdadero	Verdadero	Falso	Desconocido
Falso	Falso	Falso	Falso
Desconocido	Desconocido	Falso	Desconocido

OR	Verdadero	Falso	Desconocido
Verdadero	Verdadero	Verdadero	Verdadero
Falso	Verdadero	Falso	Desconocido
Desconocido	Verdadero	Desconocido	Desconocido

NOT	
Verdadero	Falso
Falso	Verdadero
Desconocido	Desconocido

Comparaciones con valores nulos

- En las consultas de selección, proyección y reunión natural sólo se escogen las tuplas cuya expresión evaluada de la cláusula WHERE resulte en valor Verdadero.
- Para comparar un valor de atributo con NULL use los operadores IS o IS NOT

Ej: Lista de los empleados que no tienen asignado un jefe

SELECT nombre, apellido **FROM** Empleado **WHERE** jefe **IS NULL**;

Consultas anidadas

- Bloques SELECT-FROM-WHERE dentro del WHERE de otra consulta (consulta externa)

- El operador de comparación **IN**, compara un valor x con un conjunto de valores y genera como resultado Verdadero si x es uno de los elementos del conjunto.
- **SELECT** numpry **FROM** Proyecto **WHERE** numpry **IN** (*SELECT* nump *FROM* TrabajaEn, Empleado *WHERE* ciemp=jefeci AND apellido='Suárez');

Consultas anidadas

- En la comparación se pueden usar tuplas de valores, colocandolas entre paréntesis
- Consulte las cédulas de todos los empleados que trabajen la misma combinación de (proyecto, horas) en algún proyecto en que trabaje el empleado de cédula 20.159.874:

```
SELECT DISTINCT ciemp FROM TrabajaEn
WHERE (numpry, horas) IN (SELECT numpry,
horas FROM TrabajaEn WHERE
ciemp='20159874');
```

Consultas anidadas

- La palabra clave ALL puede combinarse con los operadores >, >=, <, <= y <>
- Otros operadores equivalentes a IN: =ANY o =SOME
- ANY o SOME pueden combinarse con los operadores >, >=, <, <= y <>

SELECT apellido, nombre FROM Empleado WHERE salario > ALL (SELECT salario FROM Empleado WHERE dpto=5)

Ingeniería en Sistemas ula

Consultas correlacionadas

- Cuando una condición en el WHERE de una consulta anidada referencia a algún atributo declarado en la consulta externa, las dos consultas están correlacionadas.
- Recordar: la consulta anidada se evalúa una vez para cada tupla de la consulta externa.

Consultas correlacionadas

```
SELECT apellido, nombre FROM Empleado AS E, Dependiente AS D WHERE E.cedula=D.cedula AND E.sexo=D.sexo AND E.nombre=D.nombre
```

```
SELECT apellido, nombre FROM Empleado AS E WHERE E.cedula IN (SELECT cedula FROM Dependiente AS D WHERE E.sexo=D.sexo AND E.nombre=D.nombre)
```

Función EXISTS

- Se usa para chequear si el resultado de una consulta anidada correlacionada es vacío o no.
- El resultado de EXISTS es un booleano de valor verdadero si el resultado de la consulta anidada contiene al menos una tupla, o falso si no contiene tuplas

Función EXISTS

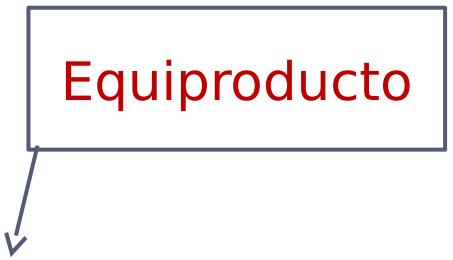
```
SELECT apellido, nombre FROM Empleado AS E WHERE EXISTS (SELECT * FROM Dependiente AS D WHERE E.sexo=D.sexo AND E.nombre=D.nombre AND E.cedula=D.cedula)

SELECT apellido, nombre FROM Empleado AS E WHERE NOT EXISTS (SELECT * FROM Dependiente AS D WHERE E.cedula=D.cedula)
```

JOIN

- Permite especificar una tabla resultante de una operación de reunión natural con las tablas de la cláusula FROM.
- Este concepto permite especificar la reunión natural y varios tipos de reuniones externas

```
SELECT nombre, apellido, direccion FROM
(Empleado JOIN Departamento ON
id_dpto=numdpto) WHERE
nombredpto='Investigación'
```



LEFT JOIN

SELECT * **FROM** A **LEFT JOIN** B **ON** condicion

- Genera una relacion con los atributos de la tabla A

```
SELECT * FROM Cliente CL, Producto P LEFT
JOIN Compra C ON
P.codproducto=C.codProCom AND
C.rifProv=CL.rif AND CL.nomCli='Fulano'
```

RIGHT JOIN

SELECT * **FROM** A **RIGHT JOIN** B **ON** condicion

- Genera una relacion con los atributos de la tabla B

SELECT * FROM Producto P RIGHT JOIN Compra C ON P.codproducto=C.codProCom AND C.rifProv='J953468823' AND P.nomPro='Lapicero'

Funciones agregadas

- Se usan para resumir información de varias tuplas en un resumen de una sola tupla.
- El agrupamiento se usa para crear subgrupos de tuplas antes de hacer el resumen.

SUM, MAX, MIN, AVG

Aserciones

- Se usan para especificar restricciones diferentes a las del modelo relacional (clave primaria, atributo de valor único, integridad de entidad e integridad referencial)

- Aserciones declarativas – LDD

Aserciones

```
CREATE ASSERTION limiteSalario
CHECK (NOT EXISTS ( SELECT * FROM
Empleado E, Empleado G, Departamento D
WHERE E.salario>G.salario AND
E.numdpto=D.numero AND
D.cedulagerente=G.cedula));
```

Disparadores

- Se usan para especificar accionesque realizará automáticamente el SGBD cuando ocurran ciertos eventos o condiciones.
- **Técnica de BD activas**

- Monitoreo de las actualizaciones en la BD
- Actualización de datos derivados de forma automática
- Mantener la consistencia de la BD

Disparadores

```
CREATE TRIGGER salarioExcedido
BEFORE INSERT OR UPDATE OF salario,
cedulasupervisor ON Empleado
FOR EACH ROW
WHEN (NEW.salario > (SELECT salario FROM
Empleado WHERE
cedula=NEW.cedulasupervisor))
InformarSupervisor(NEW.cedulasupervisor,
NEW.cedula)
```

Eventos

Condiciones

Acciones

Vistas

- Una vista es una tabla que se deriva de otras tablas (tablas base o vistas definidas previamente). Es una tabla virtual, no necesariamente existe físicamente. Limita las operaciones de actualización posibles para una vista.

Posibles uso:

- Especificar una tabla que necesitamos referenciar con frecuencia, aunque no sea una tabla base.
- Como mecanismo de seguridad y autorización

Ingeniería en Sistemas ula

Vistas

```
CREATE VIEW TrabajanEn AS
SELECT nombre, apellido, nombreproyecto,
horas
FROM Empleado AS E, Proyecto AS P,
TrabajaEn AS T
WHERE E.cedula=T.cedula AND
T.proyecto=P.numpry;

SELECT nombre, apellido FROM TrabajanEn
WHERE nombreproyecto='Freedom';
```

Vistas

CREATE VIEW InfDpto(*nombredpto, numEmpl, salarioTotal)* **AS**
SELECT D.nombre, COUNT(*), SUM(salario)
FROM Empleado AS E, Departamento AS D
WHERE E.dpto=D.id GROUP BY D.nombre;

DROP VIEW InfDpto;

Vistas

Se supone que las vistas siempre están actualizadas (responsabilidad del SGBD), si se modifican las tuplas de las tablas base sobre las que se define la vista, los cambios se reflejan automáticamente en la vista.

La vista se materializa cuando especificamos una consulta sobre la vista, no cuando se define una vista.

Ingeniería en Sistemas ula

Procedimientos almacenados

Stored Procedures
Procedimientos o funciones que son almacenadas y ejecutadas por el SGBD en el servidor de BD.

Muchos SGBD soportan procedimientos almacenados que se escriben en un lenguaje de programación de propósito general (también puede contener simples comandos SQL)

Ingeniería en Sistemas ula

Procedimientos almacenados

CREATE PROCEDURE
<nombreDelProcedimiento> (<parametros>)
<declaraciones Locales>
<cuerpoDeLaFuncion>;

CREATE FUNCTION
<nombreDelProcedimiento> (<parametros>)
RETURNS <tipoRetorno>
<declaraciones Locales>
<cuerpoDeLaFuncion>;

Procedimientos almacenados

CALL <nombreDelProcedimiento o funcion>
(<lista de argumentos>)