

EL ÁLGEBRA Y EL CÁLCULO RELACIONAL

Álgebra relacional: Lenguaje procedimental(Lenguaje que proporciona un método paso – paso para la solución de problemas) para la manipulación de relaciones.

Cálculo relacional: Lenguaje no procedimental(Lenguaje que proporciona un medio de establecer qué se desea en lugar de cómo hacerlo) para la definición de soluciones a consultas.

Lenguajes relacionalmente completos: Son llamados así los lenguajes al menos tan poderosos como el álgebra relacional.

ÁLGEBRA RELACIONAL

Las operaciones del álgebra relacional manipulan relaciones. Estas operaciones usan una o dos relaciones existentes para crear una nueva relación.

El álgebra relacional consta de las siguientes nueve operaciones: Unión, intersección, diferencia, producto, selección, proyectar, reunión, división y asignación. Las cuatro primeras de estas operaciones se toman de la teoría de conjuntos de la Matemática. Las cuatro siguientes son operaciones nuevas que se aplican específicamente al modelo de datos relacional. La última operación es la operación estándar de los lenguajes de computación, de dar un valor a un nombre.

- Unión: Operación del álgebra relacional que crea un conjunto unión de 2 relaciones unión-compatible. Permite combinar ls datos de dos relaciones.

VENDEDOR := VENDEDOR_SUBORDINADO UNION VENDEDOR_JEFE

Unión Compatible: Dos o más relaciones que tienen columnas equivalentes en número y dominios.

- Intersección: Operación del álgebra relacional que crea un conjunto intersección de dos relaciones unión-compatible. Permite identificar las filas que son comunes a dos relaciones.

VENDEDOR_SUBORDINADO_JEFE := VENDEDOR_SUBORDINADO INTERSECCION VENDEDOR_JEFE

- Diferencia: Permite identificar filas que están en una relación y no en otra. Crea un conjunto diferencia de dos relaciones unión compatible.

VENDEDOR_JEFE_JEFE := VENDEDOR_JEFE - VENDEDOR_SUBORDINADO

Sustracción: La operación diferencia del álgebra relacional.

- Producto: Operación del álgebra relacional que crea el producto cartesiano de dos relaciones. Observe que el producto se crea:
 - Concatenando los atributos de las dos relaciones.
 - Uniendo cada fila en A, con cada una de las filas en B.

P_V := PRODUCTO * VENTA

- ♦ Selección: Operación del álgebra relacional que usa una condición para seleccionar filas de una relación.

VEND_TOKIO := SELECT (OFICINA = `Tokyo`

Las condiciones de selección son esencialmente las mismas condiciones usadas en las instrucciones IF en los lenguajes tradicionales de programación.

- ◆ Protección: Operación del álgebra relacional que crea una relación borrando columnas de una relación existente. Se lista simplemente la relación original seguida de las columnas que se quieren conservar encerradas entre corchetes

VENDEDOR_TOKIO [NOV_VENDEDOR]

Una característica importante es que automáticamente elimina filas duplicadas de la operación resultante. Si dos filas enteras cualquiera en una relación son idénticas columna por columna, la fila aparece sólo una vez en la relación.

- ◆ Reunión: Operación del álgebra relacional que conecta relaciones.

Reunión natural: Se asume que se quiere tomar la reunión natural de dos relaciones, A y B, las cuales tienen las columnas C1, ..., Cn en común. Entonces JOIN (A,B) se obtiene a través de estos tres pasos:

- Tome el producto de A y B. La relación resultante tendrá dos columnas para cada C1, ..., Cn.
- Elimine todas las filas del producto, excepto aquellas en las cuales los valores de las columnas C1,..., Cn en A son iguales, respectivamente, a los valores de esas columnas en B.
- Proyecte una copia de las columnas C1, ... Cn.

Se indica que si A tiene k columnas y B tiene m columnas, entonces la reunión natural de A y B tendrá (k+m-n) columnas, donde n es el número de columnas que A y B tienen en común.

La reunión *theta* (theta join) es una reunión con una condición específica que involucra a una columna de cada relación. Esta condición especifica que las dos columnas deberían compararse de alguna forma. Operadores de comparación: =, not =, <, >, <=, >=

La manera general de expresar esto que la reunión theta tome la forma JOIN (A, B: X Y)

Donde A y B son las relaciones que serán reunidas, X y Y son las columnas de las dos relaciones y la letra griega θ es uno de los operadores de comparación relacionados.

La reunión theta no se parece a la reunión natural, porque no incluye la eliminación de una o más columnas como paso final; si A tiene k columnas y B tiene m columnas, entonces la reunión theta de A y B tendrá k + m columnas.

Equirreunión: Reunión theta basada en la igualdad de columnas específicas.

Reunión Externa: Extiende la reunión natural; consiste de dos pasos. Primero se ejecuta una reunión natural. Entonces si un registro en una relación no se corresponde con un registro de la otra relación en la reunión natural, ese registro se añade a la relación que ha sido reunida y las columnas adicionales se llenan con valores nulos

- ◆ División: Operación del álgebra relacional que crea una nueva relación, seleccionando las filas en una relación que se corresponden con todas las filas en otra relación. Se asume que A, B y C son relaciones y se desea dividir B por C, dando A como resultado.
- ◆ Las columnas de C deben ser un subconjunto de las columnas de B. Las columnas de A son todas y sólo aquellas columnas de B que no son columnas de C.

- ♦ Una fila se encuentra en A sí y sólo si está asociada en B con cada fila de C

La operación división es la opuesta de la operación producto.

◇ Asignación: Operación del álgebra relacional que da un nombre a una relación.

IMPLEMENTACIÓN RELACIONAL CON SQL

Tipos de datos:

- ◇ Numéricos exactos:
- ◇ Integer(Enteros)
- ◇ Small Integer(Enteros Pequeños)
- ◇ Numeric(p,e)(Numéricos)
- ◇ Decimal(p,e)

Para los dos últimos tipos de datos se indica una precisión (p) y una escala(e). La precisión indica el total de números o dígitos en el número y la escala indica cuántos de éstos están a la derecha del punto decimal.

- Numéricos aproximados:
- Real
- Double precision(doble precisión)
- Float(floatante)

Se usan normalmente para cálculos científicos y de ingeniería.

- Cadenas de Caracteres:
- Character(n)(Cácter)
- Character varying(n)(Cácter Variable)

Los campos de character siempre almacenan caracteres, aun cuando tengan que rellenar con blancos a la derecha para completar la longitud n. Los campos carácter varuing sólo almacenan el número real de caracteres que se introdujeron(hasta un máximo de n)

- ♦ Cadenas de bits:
- ♦ Bit(n)
- ♦ Bit varying(n)

Estos campos se usan para banderas u otras máscaras de bits para el control.

- ◇ Fechas y horas:
- ◇ Date(fecha)
- ◇ Hour(hora)
- ◇ Timestamp(Sello de tiempo)
- ◇ Time con tiempo zonal
- ◇ Timestamp con tiempo zonal

El tipo date se da en formato aaaa/mm/dd, time se da en horas(0 a 23), minutos, segundos y décimas de segundo. El timestamp es la fecha más la hora(date

plus time)

- Intervalos
- Year-month(año-mes)
- Day-time(día-hora)

Dominio: Es un tipo de datos especializado que puede estar definido dentro de un esquema y utilizando en la definición de columnas.

```
CREATE TABLE TRABAJADOR (  
  
ID_TRABAJADOR IDENTIFICADOR  
PRIMARY KEY,  
  
NOM_TRABAJADOR CARÁCTER (12) ,  
  
TARIFA_HR NUMERIC(5, 2) ,  
  
OFICIO CARÁCTER (8) ,  
  
ID_SUPV NUMERIC(4),  
  
FOREIGN KEY ID_SUPV REFERENCES  
TRABAJADOR ON DELETE SET NULL )
```

Clave Externa Recursiva: Una clave externa que referencia a su propia relación.

La cláusula ON DELETE es similar a la cláusula ON UPDATE y ambas cláusulas tienen las opciones siguientes:

- CASCADE
- SET NULL SET DEFAULT

Consultas Simples: consultas que afectan a una sola tabla de la base de datos.

Select: Lista las columnas que se desean en el resultado de la consulta.

From: Lista una o más tablas que van a ser referidas en la consulta.

Where: Contiene una condición para seleccionar las filas de las tablas que se dan en la cláusula FROM.

La consulta SQL anterior se procesa por el sistema en el orden FROM, WHERE, SELECT.

El * en la cláusula SELECT significa la fila completa.

En la cláusula SELECT se pueden definir cálculos que involucren columnas numéricas y literales numéricos combinados con las operaciones aritméticas estándar agrupadas con paréntesis cuando sea necesario.

Order By: Se usa para ordenar el resultado de la consulta en orden alfanumérico ascendente por la columna que se especifique. En orden descendente entonces esto debe especificarse añadiendo a la orden DESC.

Between: Puede usarse en la comparación de algún valor con otros dos valores. Para saber si el valor a comparar está entre estos dos valores.

Operador de comparación IN: La cláusula WHERE ha evaluado true si el tipo del oficio de la fila se encuentra en el conjunto indicado entre paréntesis.

SQL tiene dos caracteres comodines, el % y el _. El subrayado vale por exactamente un carácter cualquiera. El tanto por ciento vale por cero o cualquier cantidad de caracteres no especificados. El operador *LIKE* se usa para comparar variables de caracteres con literales cuando se utilizan comodines.

Consultas Multitablas:

SQL conecta los datos entre las tablas de la misma manera que la reunión(join) del álgebra relacional.

Como es usual, la cláusula FROM se procesa primero. En este caso, sin embargo, puesto que hay dos tablas en la cláusula. El sistema crea el producto cartesiano de las filas en estas tablas. Esto significa que se crea(lógicamente) una tabla enorme que consiste de todas las columnas de ambas tablas, emparejando cada fila de una tabla con cada fila en la otra tabla.

Después de crear esta relación gigante se aplica la cláusula WHERE, como antes. Se

examina cada fila de la relación creada por la cláusula FROM conforme a la cláusula WHERE. No se consideran aquellas que no la satisfacen. SELECT se aplica a las filas restantes.

Alias: Nombre alternativo que se le da a una relación.

Subconsultas: Una subconsulta, o una consulta dentro de una consulta, puede ponerse dentro de la cláusula WHERE de una consulta. Esto produce una expansión de las capacidades de una cláusula WHERE.

La consulta que incluye a esta subconsulta se llama consulta externa(outer query) o consulta principal.

La subconsulta puede ejecutarse lógicamente antes de que alguna fila sea examinada por la consulta principal. En cierto sentido, la subconsulta es independiente de la consulta principal. Esta podría ejecutarse como una consulta propiamente dicha. Se dice que esta clase de subconsulta no está correlacionada con la consulta principal. Las subconsultas pudieran también estar correlacionadas.

Subconsultas Correlacionadas: Clase de subconsultas cuyos valores en ejecución dependen de la fila que está siendo examinada por la consulta principal.

Exists: Evalúa verdadero si el conjunto resultante es no vacío.

No Exists: Evalúa verdadero si el conjunto resultante es vacío.

Esta consulta puede resolverse en SQL usando una doble negación.

Funciones Integradas(Built-in Functions): SQL provee funciones que se denominan funciones integradas o funciones de conjuntos(set functions). Las cinco funciones son SUM, AVG, COUNT, MAX y MIN.

Las funciones MAX y MIN operan sobre una sola columna de la relación. Seleccionan

respectivamente el valor mayor y el menor de los que se encuentran en la columna. La solución a esta consulta no incluye una cláusula WHERE.

Distinct: Operador que elimina las filas duplicadas.

Si una función integrada aparece en una cláusula SELECT, entonces nada más que funciones integradas pueden aparecer en dicha cláusula SELECT, entonces nada más que funciones integradas pueden aparecer en dicha cláusula SELECT. La única excepción ocurre en combinación con la cláusula GROUP BY que se examinará a continuación.

Group by y Having:

Group by: Indica cuáles filas deben agruparse sobre un valor común de las columnas especificadas.

Having: Una cláusula que impone condiciones a los grupos.

La diferencia entre *Having* y *Where* es que *Where* se aplica a las filas, mientras que la cláusula *Having* se aplica a los grupos.

Funciones Integradas con Subconsultas:

Una función integrada puede aparear sólo en una cláusula SELECT o en una cláusula HAVING. Sin embargo, una cláusula SELECT que contenga una función integrada puede ser parte de una subconsulta.

Operaciones del Álgebra Relacional:

Operadores Union, Intersect y Except:

Como en Álgebra relacional, la unión, la intersección y la diferencia son operaciones que se aplican a dos relaciones a la vez, que deben ser unión compatible. En SQL, dos relaciones son unión compatible si tienen el mismo número de columnas y las columnas respectivas en cada relación tienen tipos de datos compatibles, Es decir, tipos de datos que se puedan convertir de manera directa de uno a otro.

Union: (SELECT * FROM
VEND_SUBORDINADO)

UNION

(SELECT * FROM VEND_JEFE)

O la forma alternativa:

SELECT *

FROM (TABLE VEND_SUBORDINADO
UNION TABLE VEND_JEFE)

Si se usa la forma

(SELECT * FROM
VEND_SUBORDINADO)

UNION ALL

(SELECT * FROM VEND_JEFE)

O la forma

SELECT *FROM (TABLE
VEND_SUBORDINADO **UNION ALL**
TABLE VEND_JEFE)

Entonces aquellas filas que aparezcan en
ambas relaciones aparecerán duplicadas en
la relación unión.

Intersect(intersección):

(SELECT * FROM
VEND_SUBORDINADO)

INTERSECT

(SELECT * FROM VEND_JEFE)

o la forma alternativa

SELECT *

FROM (TABLE VEND_SUBORDINADO
INTERSECT TABLE VEND_JEFE)

Si se usa la forma

(SELECT * FROM

VEND_SUBORDINADO)

INTERSECT ALL

(SELECT * FROM VEND_JEFE)

Al igual que con UNION, la palabra clave ALL indica que las filas duplicadas deben considerarse como si fuesen filas independientes.

Except:

(SELECT * FROM VEND_JEFE)

EXCEPT

(SELECT * FROM
VEND_SUBORDINADO)

Si se usa la sintaxis

(SELECT * FROM VEND_JEFE)

EXCEPT ALL

(SELECT * FROM
VEND_SUBORDINADO)

Y hay m copias de una determinada fila en VEND_JEFE y n copias de la misma fila en VEND_SUBORDINADO, donde $m \leq n$, entonces no habrá copias de la fila en el resultado.

Corresponding by: Va a continuación del operador para indicar las columnas que son compatibles en ambas relaciones y que son las que deben considerarse en la operación.

Join:

- **Natural Join:** Semánticamente, la reunión natural tiene el mismo significado en SQL que en el álgebra relacional.

TRABAJADOR NATURAL JOIN
ASIGNACIÓN

- **Join Using:** (Reunión Usando) Operación que conecta las relaciones cuando las columnas

comunes designadas tienen iguales valores.

A JOIN B USING (L,N)

- **Join On:** (Reunión Cuando) Operación que conecta las relaciones cuando ocurre una condición.

TRABAJADOR T JOIN TRABAJADOR
SUPB ON T.ID_SUPV =
SUPV.ID_TRABAJADOR

Operaciones de Modificación de la Base de Datos:

Insert: Permite insertar en una relación una fila mediante la especificación de los valores de cada una de las columnas de la fila, o insertar un grupo de filas especificando una consulta que nos daría el grupo de filas a insertar.

INSERT INTO ASIGNACIÓN
(ID_TRABAJADOR, ID_EDIFICIO,
FECHA_INICIO)

VALUES (1284, 485, 13/05)

Update: Operación que cambia los valores de las columnas en las filas.

UPDATE TRABAJADOR

SET TARIFA_HR =1.05 * TARIFA_HR

WHERE ID_SUPV = 1520

Delete: Operación que quita filas de una relación. Las operaciones de DELETE se aplican también a todas las filas que satisfacen la cláusula WHERE en la instrucción DELETE.

DELETE FROM TRABAJADOR

WHERE ID_SUPV = 1520

Usar SQL con lenguajes de procesamiento de datos

SQL Empotrado:(Embedded SQL)Conjunto de instrucciones que permite que SQL sea

utilizado con lenguajes de programación tradicionales.

Lenguaje Anfitrión:(Host Language)Lenguaje de los programas en los cuales se pueden escribir las instrucciones SQL

Instrucciones de Señalización:(Flag Statements)Instrucciones SQL que se incrustan en un programa de aplicación para señalar dónde comienza o termina un conjunto de instrucciones SQL

Cursor: Una facilidad SQL incrustado en la que se almacenan los resultados de una consulta SQL para su ulterior procesamiento

Instrucción OPEN cursor: Instrucciones SQL que causan que el SGBD procese una consulta y almacene el resultado en el cursor.

Instrucción FETCH: Una instrucción que toma una fila a partir de un cursor que ya ha sido abierto.

Instrucción CLOSE: Quita los datos que están en el cursor, de modo que éste puede ser abierto de nuevo asociado con nuevos datos que reflejen otros contenidos de la base de datos.

Definición de Vistas:

Tabla base: Una tabla que contiene información básica o real.

Una vista es como una ventana en una porción de la base de datos. Las vistas son útiles para mantener la confidencialidad al restringir el acceso a partes seleccionadas de la base da datos y para simplificar tipos de consultas que sean utilizados con frecuencia.

CREATE VIEW B TRABAJADOR

AS SELECT ID_TRABAJADOR,
NOV_TRABAJADOR, OFICIO, ID_SUPV

FROM TRABAJADOR

El nombre de la vista puede estar seguido de los nombres de las columnas en la vista encerrados entre paréntesis. La parte de esta instrucción de vista que sigue a la palabra AS se denomina especificación de consulta(query specification). Cualquier consulta que sea legal puede aparecer en la definición de una vista.

Especificación de Consulta: Definición de una consulta que se usa en una definición de vista, declaración de cursor u otra instrucción.

Los datos de una vista cambian dinámicamente en la medida que cambien los datos en su tabla base subyacente.

Limitaciones para la consulta de Vistas:

- La cláusula SELECT de una consulta sobre una vista de grupos no puede tener una función integrada.
- No se puede hacer una reunión (join) de una vista de grupos con ninguna otra vista o tabla.
- Una consulta sobre una vista de grupos no puede tener a su vez cláusulas GROUP BY o HAVING.
- Una subconsulta no puede referirse a una vista de grupos.

Una vista se puede actualizar sólo si su especificación de consulta cumple con:

- Tiene una cláusula SELECT que contiene sólo nombres de columnas y no contiene la palabra clave DISTINCT.
- Tiene sólo una referencia a tabla en la cláusula FROM(esta referencia no puede incluir ninguna de las operaciones del álgebra relacional UNION, INTERSECT, EXCEPT, JOIN). Dicha tabla en la cláusula FROM debe ser o una tabla base o una vista actualizable;
- No tiene subconsulta en su cláusula WHERE que refiera a la tabla en la cláusula FROM de la consulta principal;
- No contiene una cláusula GROUP BY o HAVING

El Esquema de Información: Uno de los

criterios de Codd para decir que un SGBD sea completamente relacional es que la información del sistema que describe la base de datos esté mantenida también en tablas relacionales, al igual que otros valores de datos. Esta información descriptiva , o metadato, se mantiene normalmente en un diccionario de datos. Una versión de este diccionario de datos se le llama esquema de información.