**Universidad Nacional de Costa Rica**

**Campus Sarapiquí**

**Diseño e implementación de Base de Datos**

**Segundo Avance de Proyecto**

**Integrantes:**

**Jason Francisco Gonzalez Garcia**

**Patrick Conejo Vey**

**Yancarlos Retana Quesada**

**Profesor:**

**Ileana Schmidt Fonseca**

**Fecha:**

**25/05/2017**

**Proyección**

La operación de proyección permite seleccionar algunos atributos de una relación (sea ésta una relación o el resultado de otras operaciones). Formalmente, dada una relación R(A1, A2, ... , An), y un subconjunto de atributos de esa relación (Ai, Aj, ..., Ak), la proyección sobre esos atributos se denota como R(Ai, Aj, ..., Ak), y contiene las partes de las tuplas con los valores de los atributos proyectados.

El signo por el cual se diferencia es por el “π”.

La operación de proyección en el lenguaje de consulta SQL se realiza simplemente especificando el subconjunto de atributos en la cláusula SELECT.

**Selección**

La selección es una operación que elige algunas tuplas de una relación y elimina el resto. La nueva relación contiene por tanto solo tuplas seleccionadas que cumplen una determinada condición de selección C La condición de selección es una condición lógica que permita decidir qué incluir y qué no.

El signo por el cual se diferencia es por el “σ”.

En SQL, la selección se especifica mediante la cláusula WHERE.

**Join**

La unión de 2 relaciones R y S es otra relación la cual va a tener los registros de R en S o en ambas, además se eliminan los registros duplicados. En esta relación R y S deben ser compatibles es decir que deben estar definidas sobre el mismo conjunto de atributos.

El signo por el cual se diferencia es por el “Y”.

En SQL, la combinación se hace incluyendo más de una relación en la cláusula FROM, y la condición C se coloca como cláusula WHERE.

**Unión**

La unión de dos relaciones devuelve una nueva relación que contiene todas las tuplas que aparecían en cualquiera de las relaciones originales, o en ambas.

El signo por el cual se diferencia es por el “U”.

En SQL, la selección se especifica mediante la cláusula UNION.

**Intersección**

La intersección de relaciones toma dos relaciones y devuelve una relación con las tuplas que aparecían en ambas relaciones originales. Como la unión, la intersección solo puede aplicarse a relaciones con atributos compatibles.

El signo por el cual se diferencia es por el “∩”.

En SQL, la selección se especifica mediante la cláusula INTERSECT.

**Producto cartesiano**

El producto cartesiano es una operación que toma dos relaciones y obtiene una nueva relación cuyas tuplas son la concatenación de las tuplas de la primera relación con las tuplas de la segunda. En general, si la primera relación de entrada tenía N tuplas y la segunda M tuplas, el producto cartesiano produce N×M tuplas, aunque en ocasiones serán menos dado que las tuplas repetidas habrán de eliminarse.

El signo por el cual se diferencia es por el “X”.

El producto cartesiano de dos relaciones en SQL se obtiene simplemente colocando más de una tabla en la cláusula FROM.

**Consultas**

1. **Listar el ID y nombre de cada lector**

TT Numero\_Tarjeta\_Lector, Nombre\_Lector, Ape1\_Lector, Ape2\_Lector (Lectores)

SELECT "Numero\_Tarjeta\_Lector", "Nombre\_Lector", "Ape1\_Lector", "Ape2\_Lector" FROM "Lectores";

1. **Listar todas las Sucursales de la biblioteca.**

TT Nombre\_Sucursal (Sucursales)

SELECT "Nombre\_Sucursal" FROM "Sucursales";

1. **Listar los libros que pertenecen a un autor X**

Πtitulo\_libro(σ id\_autor=id(Autores))

SELECT titulo\_libro FROM tblibros WHERE id\_autor = 1234;

1. **Listar las sucursales que tienen más de 4 copias de un libro X.**

IDLibro= πid\_libro(σ cant\_libro>4(tblibro))

Πnom\_sucursales(id\_libro(sucursal)==id(IDLIBRO))

1. **Libros prestados a un Lector en el último año.**

Π ID\_Libro (σ Numero\_Targeta\_Lector = ‘variableIDLector’’ and ‘anioActual’ -1 = Anio\_Fecha\_Prestamo (Compras))

select ID\_Libro where Numero\_Targeta\_Lector = ‘variableIDLector’ and ‘anioActual’ – 1 = Anio\_Fecha\_Prestamo FROM “Compras”

1. **Listar los lectores que han pedido libros prestados de más de dos diferentes** **sucursales.**

**Fuente:**

<http://archive.cnx.org/contents/ef8814ef-bf92-46b8-a1c0-ad55b796fb05@1/introducci-n-al-lgebra-relacional-con-ejemplos-en-sql>

<https://quintonivel2010.wordpress.com/2010/05/28/algebra-relacional/>