



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE
HIDALGO**

**INSTITUTO DE CIENCIAS BASICAS E
INGENIERIA**

**LICENCIATURA EN CIENCIAS
COMPUTACIONALES**

BASE DE DATOS DISTRIBUIDAS

MONTSERRAT SANCHEZ CARRASCO

1.3 PRACTICA.ÁLGEBRA RELACIONAL Y SQL

INTRODUCCIÓN:

Se llama álgebra relacional a un conjunto de operaciones simples sobre tablas relacionales, a partir de las cuales se definen operaciones más complejas mediante composición.

Definen, por tanto, un pequeño lenguaje de manipulación de datos.

El elemento fundamental del modelo relacional de bases de datos es la tabla relacional. Una tabla relacional es una representación extensional de una relación definida sobre un cierto dominio.

Álgebra relacional

Es un método que consiste básicamente en crear o construir nuevas relaciones a partir de relaciones existentes.

Existen 2 tipos de operadores algebraicos:

Operadores básicos o primitivos.

Operadores no básicos o derivados.

Operadores básicos o primitivos.

Se clasifican en:

Proyección (π): Este operador permite extraer columnas de una relación y de esta manera crea un subconjunto de atributos de la relación, además elimina las filas duplicadas.

Selección (σ): Este operador permite seleccionar un subconjunto de filas o registros de una relación y de acuerdo a la condición planteada los registros serán seleccionados para formar parte de un nuevo subconjunto.

Unión (\cup): 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.

Diferencia ($-$): La diferencia de 2 relaciones R y S es otra relación la cual va a tener los registros que están en R pero no están en S.

En esta relación R y S deben ser compatibles.

Producto cartesiano (\times): Es una relación que consiste en la concatenación de cada una de las filas de la relación R con cada una de las filas de la relación S.

EJERCICIOS.

1. Escribe la sintaxis para crear la tabla "Employee".

```
CREATE TABLE Employee (  
    employee_id INT PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    salary DECIMAL(10, 2)  
    joining_date DATE,  
    departament VARCHAR(20,)  
);
```

2. Escribe la sintaxis para insertar 7 registros (de la imagen) a la tabla "Employee".

```
INSERT INTO Employee (Employee_id, First_name, Last_name, Salary, Joining_date,  
Department)  
VALUES  
    (1, 'Bob', 'Kinto', 1000000, '2019-01-20', 'Finance'),  
    (2, 'Jerry', 'Kansxo', 6000000, '2019-01-15', 'IT'),  
    (3, 'Philip', 'Jose', 8900000, '2019-02-05', 'Banking'),  
    (4, 'John', 'Mathew', 2200000, '2019-02-28', 'Insurance'),  
    (5, 'Michael', 'Abraham', 200000, '2019-05-10', 'Finance'),  
    (6, 'Alex', 'Chreketo', 4000000, '2019-05-10', 'IT'),  
    (7, 'Yohan', 'Soso', 1230000, '2019-06-20', 'Banking');
```

$r \leftarrow r \cup E$:

```
Employee←Employee U {1,'Bob', 'Kinto', 1000000, '2019-01-20', 'Finance'}  
Employee←Employee U {2, 'Jerry', 'Kansxo', 6000000, '2019-01-15', 'IT'}  
Employee←Employee U {3, 'Philip', 'Jose', 8900000, '2019-02-05', 'Banking'}  
Employee←Employee U {4, 'John', 'Mathew', 2200000, '2019-02-28', 'Insurance'}  
Employee←Employee U {5, 'Michael', 'Abraham', 200000, '2019-05-10', 'Finance'}  
Employee←Employee U {6, 'Alex', 'Chreketo', 4000000, '2019-05-10', 'IT'}  
Employee←Employee U {7, 'Yohan', 'Soso', 1230000, '2019-06-20', 'Banking'}
```

3. Escribe la sintaxis para crear la tabla "Reward".

```
CREATE TABLE Reward (  
    Employee_ref_id INT,  
    date_reward DATE,  
    amount INT  
);
```

4. Escribe la sintaxis para insertar 4 registros (en la imagen) a la tabla "Reward".

```
INSERT INTO Reward (Employee_ref_id, date_reward, amount)  
VALUES  
    (1, '2019-05-11', 1000),  
    (2, '2019-02-15', 5000),  
    (3, '2019-04-22', 2000),  
    (1, '2019-06-20', 8000);
```

$r \leftarrow r \cup E$:

Reward \leftarrow Reward $\cup \{1, '2019-05-11', 1000\}$

Reward \leftarrow Reward $\cup \{2, '2019-02-15', 5000\}$

Reward \leftarrow Reward $\cup \{3, '2019-04-22', 2000\}$

Reward \leftarrow Reward $\cup \{1, '2019-06-20', 8000\}$

5. Obtener todos los empleados.

SELECT * FROM Employee;

$\sigma_{Employee=(Employee)}$

6. Obtener el primer nombre y apellido de todos los empleados.

SELECT first_name, last_name FROM Employee;

$\Pi(\text{first_name}, \text{last_name})(Employee)$

7. Obtener todos los valores de la columna "First_name" usando el alias "Nombre de empleado".

SELECT first_name AS "Nombre de empleado" FROM Employee;

$\Pi(\text{first_name AS "Nombre de empleado"})(Employee)$

8. Obtener todos los valores de la columna "Last_name" en minúsculas.

SELECT LOWER(last_name) AS "Apellido en minúsculas" FROM Employee;

$\Pi(\text{last_name}) \text{ AS "Apellido en minúsculas" } (Employee)$

9. Obtener todos los valores de la columna "Last_name" en mayúsculas.

SELECT UPPER(last_name) AS "Apellido en mayúsculas" FROM Employee;

$\Pi(\text{last_name}) \text{ AS "Apellido en minúsculas" } (Employee)$

10. Obtener los nombre únicos de la columna "Department".

SELECT DISTINCT Department FROM Employee;

$\Pi(\text{Department}) (Employee)$

11. Obtener los primeros 4 caracteres de todos los valores de la columna "First_name".

SELECT LEFT(first_name, 4) AS Primeros_4_caracteres FROM Employee;

$\Pi(\text{first_name}, 4) \text{ AS Primeros_4_caracteres } (Employee)$

12. Obtener la posición de la letra "h" en el nombre del empleado con First_name = "Jhon".

SELECT INSTR(first_name, 'h') AS Posicion_de_h FROM Employee WHERE first_name = 'Jhon';

$\Pi(\text{first_name}, 'h') \text{ AS Posicion_de_h FROM Employee WHERE first_name = 'Jhon';}$
 $(Employee)$

13. Obtener todos los valores de la columna "First_name" después de remover los espacios en blanco de la derecha.

SELECT RTRIM(first_name) AS First_name_sin_espacios FROM Employee;

$\Pi(\text{last_name}) (\text{first_name}) \text{ AS First_name_sin_espacios}(Employee)$

14. Obtener todos los valores de la columna "First_name" después de remover los espacios en blanco de la izquierda.

SELECT LTRIM(first_name) AS First_name_sin_espacios_izquierda FROM Employee;

$\Pi(\text{first_name}) \text{ AS First_name_sin_espacios_izquierda } (Employee)$