

TRABAJO PRACTICO N°2

BASE DE DATOS(DML)

Nombre y Apellido: Agustin Casabone

Prof: Yanina Scudero

Materia: Base de datos

Comisión: 2°1ra

Introducción

En este trabajo se verán las consultas solicitadas para el trabajo practico n°2 de BASE DE DATOS , con sus tablas, sus claves (primarias y foráneas) y lo que se pide para llevarlas a cabo , también se enseñará los resultados que se ejecutaran de las consultas

Que es DML (Data Manipulation Language)

Se encarga de los datos que se encuentran dentro de las tablas, Su deber es insertar, actualizar, eliminar y consultar la información que se encuentra dentro de las tablas.

Tablas y sus datos insertados

Proveedores

```
CREATE TABLE PROVEEDORES(  
    NUMERO INT NOT NULL,  
    NOMBRE VARCHAR(15) NOT NULL,  
    DOMICILIO VARCHAR(15),  
    LOCALIDAD VARCHAR(20) NOT NULL,  
    PRIMARY KEY(NUMERO)  
);
```

```
INSERT INTO PROVEEDORES (NUMERO, NOMBRE, DOMICILIO, LOCALIDAD) VALUES (101, 'GOMEZ', 'NAZCA 920', 'CAPITAL FEDERAL');  
INSERT INTO PROVEEDORES (NUMERO, NOMBRE, DOMICILIO, LOCALIDAD) VALUES (102, 'PÉREZ', 'ARGERICH 1030', 'CAPITAL FEDERAL');  
INSERT INTO PROVEEDORES (NUMERO, NOMBRE, DOMICILIO, LOCALIDAD) VALUES (103, 'VÁZQUEZ', 'SARMIENTO 450', 'RAMOS MEJÍA');  
INSERT INTO PROVEEDORES (NUMERO, NOMBRE, DOMICILIO, LOCALIDAD) VALUES (104, 'LÓPEZ', 'ALSINA 720', 'AVELLANEDA');
```

proveedores (4r × 4c)				
#	NUMERO 	NOMBRE	DOMICILIO	LOCALIDAD
1	101	GOMEZ	NAZCA 920	CAPITAL FEDERAL
2	102	PÉREZ	ARGERICH 1030	CAPITAL FEDERAL
3	103	VÁZQUEZ	SARMIENTO 450	RAMOS MEJÍA
4	104	LÓPEZ	ALSINA 720	AVELLANEDA

Productos

```
CREATE TABLE PRODUCTOS(  
    PNRO INT NOT NULL,  
    PNOMBRE VARCHAR(15) NOT NULL,  
    PRECIO DOUBLE NOT NULL,  
    TAMAÑO VARCHAR(15) NOT NULL,  
    LOCALIDAD VARCHAR(20) NOT NULL,  
    PRIMARY KEY(PNRO)  
);
```

```
INSERT INTO PRODUCTOS (PNRO, PNOMBRE, PRECIO, TAMAÑO, LOCALIDAD) VALUES (001, 'TALCO', 5, 'CHICO', 'CAPITAL FEDERAL');  
INSERT INTO PRODUCTOS (PNRO, PNOMBRE, PRECIO, TAMAÑO, LOCALIDAD) VALUES (002, 'TALCO', 7, 'MEDIANO', 'CAPITAL FEDERAL');  
INSERT INTO PRODUCTOS (PNRO, PNOMBRE, PRECIO, TAMAÑO, LOCALIDAD) VALUES (003, 'CREMA', 8, 'GRANDE', 'RAMOS MEJIA');  
INSERT INTO PRODUCTOS (PNRO, PNOMBRE, PRECIO, TAMAÑO, LOCALIDAD) VALUES (004, 'CEPILLO', 2, 'GRANDE', 'AVELLANEDA');  
INSERT INTO PRODUCTOS (PNRO, PNOMBRE, PRECIO, TAMAÑO, LOCALIDAD) VALUES (005, 'ESMALTE', 1.20, 'NORMAL', 'CHACARITA');
```

Productos (5r × 5c)						
#	PNRO		PNOMBRE	PRECIO	TAMAÑO	LOCALIDAD
1	1	1	TALCO	5	CHICO	CAPITAL FEDERAL
2	2	2	TALCO	7	MEDIANO	CAPITAL FEDERAL
3	3	3	CREMA	8	GRANDE	RAMOS MEJIA
4	4	4	CEPILLO	2	GRANDE	AVELLANEDA
5	5	5	ESMALTE	1,2	NORMAL	CHACARITA

Prov-prod

```
CREATE TABLE PROVPROD (
    NUMERO INT NOT NULL,
    PNRO INT NOT NULL,
    CANTIDAD INT NOT NULL,
    PRIMARY KEY(NUMERO, PNRO),
    FOREIGN KEY (NUMERO) REFERENCES PROVEEDORES(NUMERO),
    FOREIGN KEY (PNRO) REFERENCES PRODUCTOS(PNRO)
);
```


```
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (101, 001, 300);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (101, 002, 200);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (101, 003, 400);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (101, 004, 200);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (101, 005, 100);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (102, 001, 300);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (102, 002, 400);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (103, 002, 200);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (104, 002, 200);
INSERT INTO PROVPROD (NUMERO, PNRO, CANTIDAD) VALUES (104, 004, 300);
```

Consultas

1) Obtener los detalles completos de todos los productos.

```
SELECT * FROM productos;
```

Resultado:

Productos (5r × 5c)						
#	PNRO 	PNOMBRE	PRECIO	TAMAÑO	LOCALIDAD	
1	1	TALCO	5	CHICO	CAPITAL FEDERAL	
2	2	TALCO	7	MEDIANO	CAPITAL FEDERAL	
3	3	CREMA	8	GRANDE	RAMOS MEJIA	
4	4	CEPILLO	2	GRANDE	AVELLANEDA	
5	5	ESMALTE	1,2	NORMAL	CHACARITA	

Lo que podemos ver en esta consulta, es que nos pide mostrar en su totalidad la tabla productos con cada una de sus filas y columnas, se puede todos los datos insertados como son PNRO, PNOMBRE, PRECIO, etc. también se nos muestra su llave primaria cual es PNRO.

2) Obtener los detalles completos de todos los proveedores de Capital.

```
SELECT *  
FROM Proveedores  
WHERE LOCALIDAD ="CAPITAL FEDERAL";
```

Resultado:

Proveedores (2r × 4c)				
#	NUMERO 	NOMBRE	DOMICILIO	LOCALIDAD
1	101	GOMEZ	NAZCA 920	CAPITAL FEDERAL
2	102	PÉREZ	ARGERICH 1030	CAPITAL FEDERAL

Como podemos ver, es similar a la anterior consulta, pero nos piden mostrar solo a los proveedores que estén en capital federal.

3) Obtener todos los envíos en los cuales la cantidad está entre 200 y 300.

```
SELECT *  
FROM PROVPROD  
WHERE CANTIDAD BETWEEN 200 AND 300;
```

Resultado:

PROVPROD (7r × 3c)				
#	NUMERO	PNRO	CANTIDAD	
1	101	1	300	
2	101	2	200	
3	101	4	200	
4	102	1	300	
5	103	2	200	
6	104	2	200	
7	104	4	300	

en este caso, podemos ver todos los envíos que se han realizado entre las cantidades de 200 y 300, se utiliza BETWEEN para conocer el rango de estos números.

4) Obtener los números de los productos enviados por algún proveedor de Avellaneda.

```
SELECT NUMERO,PNRO
FROM PROVPROD
WHERE NUMERO IN (
    SELECT NUMERO
    FROM PROVEEDORES
    WHERE LOCALIDAD = 'AVELLANEDA'
);
```

Resultado:

PROVPROD (2r × 2c)			
#	NUMERO	PNRO	
1	104	2	
2	104	4	

Nos piden mostrar los números que hayan sido enviados por proveedores de Avellaneda, para esto se usara IN que usara una Subconsulta para traer el número que identifica al proveedor de avellaneda (que en este caso es el 104) luego en la consulta principal compara el número que tiene en WHERE para saber si coinciden con que tiene en la lista prov-prod.

5) Obtener la cantidad total del producto 001 enviado por el proveedor 103.

```
SELECT SUM(CANTIDAD)
FROM PROVPROD
WHERE PNRO =001 AND NUMERO =103;
```

Resultado :

PROVPROD (1r × 1c)	
#	SUM(CANTIDAD)
1	(NULL)

Nos piden obtener la cantidad total del producto 001 enviado por el proveedor 103, se usará SUM que mostrará la cantidad total del proveedor 103, aun que como se puede ver mostrará NULL ya que el proveedor no envió nada del producto 001.

6) Obtener los números de productos y localidades en la cual la segunda letra del nombre de la localidad sea A

```
SELECT PNRO, LOCALIDAD
FROM productos
WHERE LOCALIDAD LIKE '_A%';
```

Resultado:

productos (3r × 2c)		
#	PNRO	LOCALIDAD
1	1	CAPITAL FEDERAL
2	2	CAPITAL FEDERAL
3	3	RAMOS MEJIA

Nos piden que obtengamos los números de los productos y localidades en los cual su segunda letra sea una a, su usara LIKE usan el guion bajo para especificar que queremos sea la segunda letra de la localidad.

7) Obtener los precios de los productos enviados por el proveedor 102.

```
SELECT PP.NUMERO, P.PRECIO
FROM PRODUCTOS AS p,PROVPROD AS pp
WHERE P.PNRO = PP.PNRO AND PP.NUMERO = 102;
```

Resultado:

PRODUCTOS (2r × 2c)		
#	NUMERO 	PRECIO
1	102	5
2	102	7

Nos piden obtener los precios de los productos que fueron enviados por el proveedor 102, debido a que la columna precio se encuentra en la tabla productos y la de envíos realizados por el proveedor se encuentra en prov-prod, se utilizara un alias para poder identificar los precios y otro para distinguir el número Del proveedor. En WHERE se buscará que el PNRO de productos se igual PNRO de proveedores al igual que su número.

8) Construir una lista de todas las localidades en las cuales esté situado por lo menos un proveedor o un producto.

```
SELECT LOCALIDAD
FROM PROVEEDORES
UNION
SELECT LOCALIDAD
FROM PRODUCTOS;
```

Resultado:

PROVEEDORES (4r × 1c)	
#	LOCALIDAD
1	CAPITAL FEDERAL
2	RAMOS MEJÍA
3	AVELLANEDA
4	CHACARITA

Se nos pide hacer una lista donde por lo menos haya un proveedor y un producto, para esto se usará UNION que mostrara la combinación de las tablas y nos dará las localidades donde hay tanto productos como proveedores.

9) Cambiar a “Chico” el tamaño de todos los productos medianos.

```
UPDATE PRODUCTOS
SET TAMAÑO = 'CHICO'
WHERE TAMAÑO = 'MEDIANO';
```

Resultado:

Sin UPDATE:

Productos (5r × 5c)						
#	PNRO	PNOMBRE	PRECIO	TAMAÑO	LOCALIDAD	
1	1	TALCO	5	CHICO	CAPITAL FEDERAL	
2	2	TALCO	7	MEDIANO	CAPITAL FEDERAL	
3	3	CREMA	8	GRANDE	RAMOS MEJIA	
4	4	CEPILLO	2	GRANDE	AVELLANEDA	
5	5	ESMALTE	1,2	NORMAL	CHACARITA	

Con UPDATE:

productos (5r × 5c)						
#	PNRO	PNOMBRE	PRECIO	TAMAÑO	LOCALIDAD	
1	1	TALCO	5	CHICO	CAPITAL FEDERAL	
2	2	TALCO	7	CHICO	CAPITAL FEDERAL	
3	3	CREMA	8	GRANDE	RAMOS MEJIA	
4	4	CEPILLO	2	GRANDE	AVELLANEDA	
5	5	ESMALTE	1,2	NORMAL	CHACARITA	

Se nos pide actualizar todos los productos medianos a chico, se utiliza UPDATE que actualizara la tabla y SET especificara que todos los medianos pasaran a ser chicos, aunque en este caso solo cambiara un solo producto (talco) ya que es el único que es un producto mediano.

10) Eliminar todos los productos para los cuales no haya envíos.

```
DELETE FROM PRODUCTOS
WHERE PNRO NOT IN (
SELECT PNRO
FROM PROVPROD
);
```

Se Nos pide eliminar los productos que no tengan envíos, para esto se usara DELETE y se especificara con NOT IN para borrar los registros que no es estén en el PNRO de prov-prod, no se mostrara el resultado ya que todos los productos tienen envío.

11) Insertar un proveedor (107). El nombre y la localidad son Rosales y Wilde, el domicilio no se conoce todavía.

```
INSERT INTO PROVEEDORES (NUMERO, NOMBRE, DOMICILIO, LOCALIDAD) VALUES (107, 'ROSALES', NULL, 'WILDE');
```

Resultado:

Antes del insert

proveedores (4r x 4c)				
#	NUMERO 	NOMBRE	DOMICILIO	LOCALIDAD
1	101	GOMEZ	NAZCA 920	CAPITAL FEDERAL
2	102	PÉREZ	ARGERICH 1030	CAPITAL FEDERAL
3	103	VÁZQUEZ	SARMIENTO 450	RAMOS MEJÍA
4	104	LÓPEZ	ALSINA 720	AVELLANEDA

Después del insert

#	NUMERO 	NOMBRE	DOMICILIO	LOCALIDAD
1	101	GOMEZ	NAZCA 920	CAPITAL FEDERAL
2	102	PÉREZ	ARGERICH 1030	CAPITAL FEDERAL
3	103	VÁZQUEZ	SARMIENTO 450	RAMOS MEJÍA
4	104	LÓPEZ	ALSINA 720	AVELLANEDA
5	107	ROSALES	(NULL)	WILDE

En esta ultima consulta, se nos solicita que insertemos un nuevo proveedor, se usara un INSERT para inserta los nuevos datos para la tabla de proveedores.