

**UNIVERSIDAD PRIVADA FRANZ TAMAYO**  
**INGENIERÍA DE SISTEMAS**  
**BASE DE DATOS II**



**PROYECTO BASE DE DATOS**  
**“BASE DE DATOS VETERINARIA”**

**Estudiantes: Huaycho Quispe Jhomar**

**Docente: Lic. William Barra Paredes**

**Fecha: 06/12/2022**

**LA PAZ, BOLIVIA**

**2022**

## 1. INTRODUCCION

El presente trabajo muestra una base de datos orientada al control de ventas, servicios y el control de vacunas a mascotas, dentro de una veterinaria.

## 2. ANALISIS Y DISEÑO DE BASE DE DATOS

### a) CONTEXTO DE BASE DE DATOS

Dado que la base de datos se centrará en el manejo de registros de control dentro de una veterinaria el nombre más adecuado a este es “DB\_VETERINARIA”.

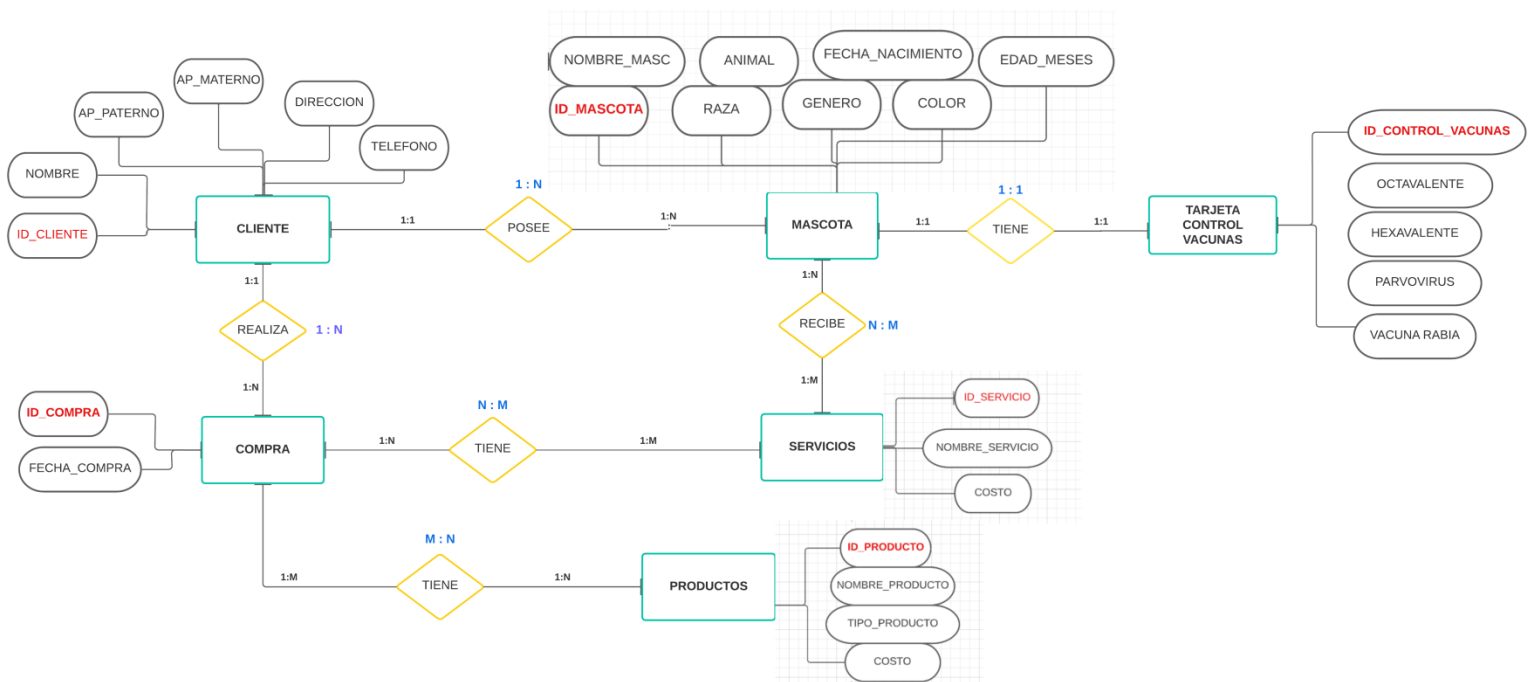
### b) ENTIDADES/ TABLA DE SISTEMA

<b>CLIENTE</b>	Almacena los datos necesarios del cliente, nombre, apellidos, teléfono, etc.
<b>COMPRA</b>	Esta tabla almacena el ID_CLIENTE Quien ha realizado la compra y la fecha de la compra
<b>MASCOTA</b>	Almacena los datos necesarios de la mascota, nombre, raza, que animal es, etc.
<b>PRODUCTO</b>	Almacena el nombre del producto, precio, y el tipo al que pertenece Ejem : (alimento, salud, etc.).
<b>SERVICIOS</b>	Almacena el nombre de los servicios, nombre y costo.
<b>TARJETA_CONTROL_VACUNAS</b>	Contiene las vacunas que debe recibir cada mascota

### c) DISEÑO DE LA BASE DE DATOS

#### MODELO ENTIDAD D RELACION

Enlace Github: [https://github.com/JHOMARHUAYCHO/BASE-DE-DATOS-2/blob/main/HITO%205/PROYECTO%20FINAL%20DB\\_VETERINARIA/DB\\_VETERINARIA%20-%20DIAGRAMA%20ENTIDAD%20RELACION.png](https://github.com/JHOMARHUAYCHO/BASE-DE-DATOS-2/blob/main/HITO%205/PROYECTO%20FINAL%20DB_VETERINARIA/DB_VETERINARIA%20-%20DIAGRAMA%20ENTIDAD%20RELACION.png)



## CODIGO SQL

Enlace Github : [https://github.com/JHOMARHUAYCHO/BASE-DE-DATOS-2/blob/main/HITO%205/PROYECTO%20FINAL%20DB\\_VETERINARIA/DB\\_VETERINARIA.sql](https://github.com/JHOMARHUAYCHO/BASE-DE-DATOS-2/blob/main/HITO%205/PROYECTO%20FINAL%20DB_VETERINARIA/DB_VETERINARIA.sql)

```
CREATE DATABASE DB_VETERINARIA;
USE DB_VETERINARIA;

CREATE TABLE CLIENTE
(
    ID_CLIENTE INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,
    NOMBRE_CLIENT VARCHA (30) ,
    AP_PATERNO VARCHA (30) ,
    AP_MATERNO VARCHA (30) ,
    DIRECCION VARCHA (30) ,
    TELEFONO INTEGER
);

CREATE TABLE MASCOTA
(
    ID_MASCOTA INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,
    NOMBRE_MASC VARCHA (30) ,
    ANIMAL VARCHA (30) ,
    FECHA_NACIMIENTO DATE,
    EDAD_MESES INTEGER,
    GENERO VARCHA (30) ,
    RAZA VARCHA (40) ,
    COLOR VARCHA (30) ,
    DUEÑO INTEGER NOT NULL,

    FOREIGN KEY (DUEÑO) REFERENCES CLIENTE (ID_CLIENTE)
);

CREATE TABLE COMPRA
(
    ID_COMPRA INTEGER AUTO_INCREMENT NOT NULL PRIMARY KEY ,
    COD_CLIENTE INTEGER NOT NULL,
    FECHA_COMPRA DATE,

    FOREIGN KEY (COD_CLIENTE) REFERENCES CLIENTE (ID_CLIENTE)
);

CREATE TABLE PRODUCTO
```

```

(
    ID_PRODUCTO INTEGER AUTO_INCREMENT PRIMARY KEY NOT NULL,
    NOMBRE_PRODUCTO VARCHAR(40),
    TIPO_PRODUCTO VARCHAR(30),
    COSTO REAL
);
CREATE TABLE SERVICIO
(
    ID_SERVICIO INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,
    NOMBRE_SERVICIO VARCHAR(30),
    COSTO REAL
);

CREATE TABLE CONTENIDO_COMPRA
(
    ID_CONT_COMPRA INTEGER NOT NULL PRIMARY KEY AUTO_INCREMENT,
    COD_COMPRA INTEGER NOT NULL ,
    COD_PRODUCTO INTEGER NOT NULL ,

    FOREIGN KEY (COD_COMPRA) REFERENCES COMPRA(ID_COMPRA),
    FOREIGN KEY (COD_PRODUCTO) REFERENCES PRODUCTO(ID_PRODUCTO)
);

CREATE TABLE CONTENIDO_SERVICIOS
(
    ID_CONT_SERVICIO INTEGER NOT NULL AUTO_INCREMENT PRIMARY KEY ,
    COD_COMPRA INTEGER NOT NULL,
    COD_SERVICIO INTEGER NOT NULL ,
    FOREIGN KEY (COD_COMPRA) REFERENCES COMPRA(ID_COMPRA),
    FOREIGN KEY (COD_SERVICIO) REFERENCES SERVICIO(ID_SERVICIO)
);

CREATE TABLE RECIBE_SERVICIOS
(
    ID_RECIBE_SERVICIO INTEGER NOT NULL AUTO_INCREMENT PRIMARY KEY,
    COD_MASCOTA INTEGER NOT NULL ,
    COD_SERVICIO INTEGER NOT NULL ,
    FOREIGN KEY (COD_MASCOTA) REFERENCES MASCOTA(ID_MASCOTA),
    FOREIGN KEY (COD_SERVICIO) REFERENCES SERVICIO(ID_SERVICIO)
);

CREATE TABLE CONTROL_VACUNAS
(
    ID_CONTROL_VAC INTEGER NOT NULL AUTO_INCREMENT PRIMARY KEY ,
    OCTAVALENTE VARCHAR(15),
    HEXAVALENTE VARCHAR(15),
    PARVOVIRUS VARCHAR(15),
    VACUNA_RABIA VARCHAR(15),

    MASCOTA INTEGER NOT NULL ,
    FOREIGN KEY (MASCOTA) REFERENCES MASCOTA(ID_MASCOTA)
);

#####
#####
##### LLENADO DE DATOS
#####

```

```
#////////////////////////////////////  
////////////////////////////////////
```

```
INSERT INTO  
    CLIENTE (NOMBRE_CLIENT, AP_PATERNO, AP_MATERNO, DIRECCION, TELEFONO)  
VALUES ('LOREANA', 'SAMUDIO', 'LOPEZ', 'CALACOTO CALLE 10-1353', 76583942),  
       ('ROBERTO', 'GONZALES', 'SURITA', 'VIACHA CALLE 13-234', 65783982),  
       ('CLAUDIA', 'TORRES', 'ROCHA', 'HORIZONTES CALLE 12-4532', 74846251),  
       ('LEONARDO', 'VILLCA', 'IMACA', 'CEJA CALLE 10-2342', 76897658),  
       ('KARLA', 'LUARTE', 'PEREZ', 'MONTES CALLE 10-1342', 65453478),  
       ('RODRIGO', 'FERNANDEZ', 'OBRADOR', 'SENKATA CALLE 10-  
53453', 67453478);
```

```
INSERT INTO MASCOTA (NOMBRE_MASC, ANIMAL, FECHA_NACIMIENTO, GENERO, RAZA,  
COLOR, DUEÑO)  
VALUES ('TITO', 'PERRO', '2021-12-12', 'MACHO', 'SANBERNARDO', 'CAFE Y  
BLANCO', 1),  
       ('PELUSA', 'GATO', '2020-08-15', 'HEMBRA', 'MEZTIZO', 'BLANCO Y  
NEGRO', 2),  
       ('WOOPER', 'PERRO', '2021-12-12', 'MACHO', 'ROTTWEILER', 'CAFE  
NEGRO', 4),  
       ('SNOWY', 'GATO', '2020-06-12', 'HEMBRA', 'MEZTIZO', 'BLANCO', 3),  
       ('FROGGY', 'PERRO', '2019-11-25', 'MACHO', 'MEZTIZO', 'BLANCO Y  
PLOMO', 5),  
       ('TOTO', 'PERRO', '2021-01-15', 'MACHO', 'ALASKA', 'NEGRO Y BLANCO', 6),  
       ('SASI', 'GATO', '2018-04-10', 'HEMBRA', 'BOMBAY', 'NEGRO', 1),  
       ('MISHA', 'GATO', '2015-06-19', 'HEMBRA', 'SIAMES', 'NEGRO Y MARRON', 3);
```

```
INSERT INTO PRODUCTO (NOMBRE_PRODUCTO, TIPO_PRODUCTO, COSTO)  
VALUES ('COLLAR', 'ACCESORIO', 15),  
       ('RICOCAT SALMON', 'ALIMENTO', 7),  
       ('RICOCAT CARNE', 'ALIMENTO', 8),  
       ('RICOCRACK CARNE', 'ALIMENTO', 22),  
       ('HUESO MASTICABLE', 'ACCESORIO', 26),  
       ('BEBEDERO PORTATIL', 'ACCESORIO', 35),  
       ('TAZON PARA GATO', 'ACCESORIO', 16),  
       ('VARA PARA GATOS', 'ACCESORIO', 20),  
       ('PELUCHE MINITORTUGUITA', 'ACCESORIO', 20),  
       ('COMEDOR PARA PERRO', 'ACCESORIO', 34),  
       ('COMEDOR DOBLE PARA GATO', 'ACCESORIO', 22),  
       ('COMEDOR DOCIFICADOR DE AGUA', 'ACCESORIO', 64),  
       ('COMEDOR DOCIFICADOR DE COMIDA', 'ACCESORIO', 70),  
       ('COCHONETA NARANJA', 'ACCESORIO', 32),  
       ('OSPRET SHAMPOO', 'ESTETICA HIGIENE', 65),  
       ('OSPRET SHAMPOO PERFUME FRUTAL', 'ESTETICA HIGIENE', 57),  
       ('REMOVEDOR DE SEDUMEN', 'ESTETICA HIGIENE', 30),  
       ('APETICAT JARABE', 'SALUD', 40),  
       ('PET LACT SUSTITUTO LACTEO', 'SALUD', 121),  
       ('ARTRIN', 'SALUD', 95),  
       ('APRAX SUSPENSION', 'SALUD', 75),  
       ('TIDY CATS ARENA', 'ESTETICA SANTARIA', 21),
```

```

        ('STONE CAT ARENA SANITARIA', 'ESTETICA SANITARIA', 43),
        ('CEPILLO CHOM CHOM ROLLER', 'ESTETICA SANITARIA', 330);

INSERT INTO SERVICIO(NOMBRE_SERVICIO, COSTO)
VALUES ('ESTERILIZACION PERRO MACHO', 300),
        ('ESTERILIZACION GATO MACHO', 200),
        ('ESTERILIZACION GATO HEMBRA', 300),
        ('ESTERILIZACION PERRO HEMBRA', 500),
        ('DESPARAZITACION', 40),
        ('CONSULTA DOMICILIO', 50),
        ('CONSULTA CONSULTORIO', 100),
        ('VACUNA OCTAVALENTE', 70),
        ('VACUNA HEXAVALENTE', 60),
        ('VACUNA PARVOVIRUS', 60),
        ('VACUNA RABIA', 50);

INSERT INTO COMPRA (COD_CLIENTE, FECHA_COMPRA)
VALUES (1, '2021-11-11'),
        (3, '2021-11-11'),
        (4, '2021-11-14'),
        (5, '2021-11-15'),
        (6, '2021-12-20'),
        (4, '2021-12-25'),
        (5, '2022-12-29'),
        (2, '2022-01-02'),
        (3, '2022-01-02');

INSERT INTO CONTENIDO_COMPRA(COD_COMPRA, COD_PRODUCTO)
VALUES (1, 1),
        (1, 2),
        (1, 3),
        (2, 2),
        (2, 5),
        (3, 4),
        (5, 3),
        (5, 2),
        (6, 15),
        (7, 12),
        (7, 7),
        (8, 3),
        (8, 2),
        (9, 4),
        (9, 6);

INSERT INTO CONTENIDO_SERVICIOS(COD_COMPRA, COD_SERVICIO)
VALUES (1, 1),
        (3, 8),
        (4, 10),
        (5, 11),
        (8, 3),
        (9, 8),
        (9, 10),
        (9, 7),
        (9, 10);

INSERT INTO RECIBE_SERVICIOS(COD_MASCOTA, COD_SERVICIO)
VALUES (1, 1),
        (3, 8),

```

```
(5,10),
(6,11),
(2,3),
(4,8),
(4,10),
(8,7),
(8,10);
```

```
#####
##### CONSULTAS #####
#####
```

```
# CONSULTA 1: MUESTRA ID_CLIENTE NOMBRE DEL DUEÑO NOMBRE DE LA MASCOTA
ID_MASCOTA ORDENADO POR APELLIDO PATERNOO
```

```
SELECT CLIENTE.ID_CLIENTE,
        CONCAT(CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO, '
', CLIENTE.NOMBRE_CLIENTE) AS DUEÑO,
        MASCOTA.NOMBRE_MASC AS MASCOTA,
        MASCOTA.ANIMAL,
        MASCOTA.GENERO,
        MASCOTA.ID_MASCOTA
FROM CLIENTE
INNER JOIN MASCOTA ON CLIENTE.ID_CLIENTE = MASCOTA.DUEÑO
ORDER BY AP_PATERNO;
```

```
#CONSULTA 2: MUESTRA A LA MASCOTA SU RAZA Y CUANTAS VACUNAS YA HA
RECIBIDO
```

```
SELECT MASCOTA.NOMBRE_MASC,
        MASCOTA.RAZA,
        CONTROL_VACUNAS.HEXAVALENTE,
        CONTROL_VACUNAS.OCTAVALENTE,
        CONTROL_VACUNAS.PARVOVIRUS,
        CONTROL_VACUNAS.VACUNA_RABIA
FROM MASCOTA
INNER JOIN CONTROL_VACUNAS ON MASCOTA.ID_MASCOTA =
CONTROL_VACUNAS.MASCOTA;
```

```
#CONSULTA 3: MUESTRA EL NOMBRE DE LAS MASCOTAS Y EL SERVICIO QUE HAN
RECIBIDO
```

```
SELECT
        MASCOTA.ID_MASCOTA,
        MASCOTA.NOMBRE_MASC,

        SERVICIO.NOMBRE_SERVICIO,
        CONCAT(CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO, '
', CLIENTE.NOMBRE_CLIENTE) AS DUEÑO
FROM RECIBE_SERVICIOS
INNER JOIN MASCOTA ON RECIBE_SERVICIOS.COD_MASCOTA = MASCOTA.ID_MASCOTA
INNER JOIN SERVICIO ON RECIBE_SERVICIOS.COD_SERVICIO =
SERVICIO.ID_SERVICIO
INNER JOIN CLIENTE ON MASCOTA.DUEÑO = CLIENTE.ID_CLIENTE
ORDER BY MASCOTA.ID_MASCOTA;
```



```
#CONSULTA 4: MUESTRA TODAS LAS COMPRAS DE PRODUCTOS QUE SE HAN REALIZADO
SELECT
```

```
    COMPRA.ID_COMPRA,
    CONCAT(CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO, '
', CLIENTE.NOMBRE_CLIENTE) AS DUEÑO,
    PRODUCTO.NOMBRE_PRODUCTO,
    COSTO,
    COMPRA.FECHA_COMPRA
FROM CONTENIDO_COMPRA
INNER JOIN PRODUCTO ON CONTENIDO_COMPRA.COD_PRODUCTO =
PRODUCTO.ID_PRODUCTO
INNER JOIN COMPRA ON CONTENIDO_COMPRA.COD_COMPRA = COMPRA.ID_COMPRA
INNER JOIN CLIENTE ON CLIENTE.ID_CLIENTE= COMPRA.COD_CLIENTE
ORDER BY COMPRA.ID_COMPRA;
```

```
#CONSULTA 5; MUESTRA TODOS LOS SERVICIOS COMPRADOS POR LA FECHA DE
COMPRA
```

```
SELECT
    ID_COMPRA,
    CONCAT(CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO, '
', CLIENTE.NOMBRE_CLIENTE) AS DUEÑO,
    NOMBRE_SERVICIO,
    COSTO,
    FECHA_COMPRA
FROM CONTENIDO_SERVICIOS
INNER JOIN COMPRA ON CONTENIDO_SERVICIOS.COD_COMPRA = COMPRA.ID_COMPRA
INNER JOIN CLIENTE ON COMPRA.COD_CLIENTE = CLIENTE.ID_CLIENTE
INNER JOIN SERVICIO ON CONTENIDO_SERVICIOS.COD_SERVICIO =
SERVICIO.ID_SERVICIO
ORDER BY FECHA_COMPRA;
```

```
#####
#///////// FUNCIONES ///////////////////////////////////////////
#/////////
```

```
# FUNCION 1
```

```
CREATE OR REPLACE FUNCTION PRECIO_TOTAL(CODIGO_COMPRA INTEGER)
RETURNS REAL
BEGIN
    DECLARE RESPUESTA REAL DEFAULT 0;
    DECLARE TOTAL_SERVICIO REAL DEFAULT 0;
    DECLARE TOTAL_PRODUCTOS REAL DEFAULT 0;

    SELECT SUM(PRODUCTO.COSTO) INTO TOTAL_PRODUCTOS
    FROM PRODUCTO
    INNER JOIN CONTENIDO_COMPRA ON PRODUCTO.ID_PRODUCTO =
CONTENIDO_COMPRA.COD_PRODUCTO
    WHERE COD_COMPRA=CODIGO_COMPRA;

    SELECT SUM(SERVICIO.COSTO) INTO TOTAL_SERVICIO
    FROM SERVICIO
    INNER JOIN CONTENIDO_SERVICIOS ON SERVICIO.ID_SERVICIO =
CONTENIDO_SERVICIOS.COD_SERVICIO
```

```

        WHERE COD_COMPRA=CODIGO_COMPRA;

        IF (TOTAL_PRODUCTOS IS NULL) THEN SET TOTAL_PRODUCTOS=0;
        END IF;

        IF (TOTAL_SERVICIO IS NULL) THEN SET TOTAL_SERVICIO=0;
        END IF;

        SET RESPUESTA=TOTAL_SERVICIO+TOTAL_PRODUCTOS;
        RETURN RESPUESTA;
    end;

SELECT PRECIO_TOTAL(9);

#FUNCION 2

CREATE OR REPLACE FUNCTION BUSCAR_MASCOTA(NOMBRE_MASCOTA VARCHAR(30))
RETURNS TEXT
BEGIN
    DECLARE RESPUESTA TEXT DEFAULT '';

    SELECT CONCAT('DUEÑO: ', CLIENTE.NOMBRE_CLIENT, '
', CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO, '      MASCOTA:
', MASCOTA.NOMBRE_MASC) INTO RESPUESTA
    FROM MASCOTA
    INNER JOIN CLIENTE ON MASCOTA.DUEÑO = CLIENTE.ID_CLIENTE
    WHERE NOMBRE_MASC=NOMBRE_MASCOTA;

    RETURN RESPUESTA;
end;

SELECT BUSCAR_MASCOTA('SNOWY');

#FUNCION 3

CREATE OR REPLACE FUNCTION CANTIDAD_MASCOTAS_CLIENTES(CODIGO INTEGER)
RETURNS TEXT
BEGIN
    DECLARE RESPUESTA TEXT DEFAULT '';
    DECLARE NUMERO_MASCOTAS INTEGER DEFAULT 0;
    DECLARE NOMBRE VARCHAR(30) DEFAULT '';

    SELECT CONCAT(CLIENTE.NOMBRE_CLIENT, '
', CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO) INTO NOMBRE
    FROM CLIENTE
    WHERE ID_CLIENTE=CODIGO;

    SELECT COUNT(MASCOTA.ID_MASCOTA) INTO NUMERO_MASCOTAS
    FROM MASCOTA
    WHERE DUEÑO=CODIGO;

    SET RESPUESTA=CONCAT('EL DUEÑO: ', NOMBRE, '      POSEE
', NUMERO_MASCOTAS, ' MASCOTAS');
    RETURN RESPUESTA;
end;

```

```

SELECT CANTIDAD_MASCOTAS_CLIENTES(1);

#FUNCION 4
CREATE OR REPLACE FUNCTION COMPRAR_PRODUCTO(CLIENTE_ID INTEGER,
PRODCUTO_ID INTEGER)
RETURNS VARCHAR(30)
BEGIN
    DECLARE AUX INTEGER DEFAULT 0;
    DECLARE RESP VARCHAR(30) DEFAULT 'REGISTRADO';

    INSERT INTO COMPRA(COD_CLIENTE, FECHA_COMPRA) VALUES
(CLIENTE_ID, NOW());

    SELECT MAX(ID_COMPRA) INTO AUX FROM COMPRA;

    INSERT INTO CONTENIDO_COMPRA( COD_COMPRA, COD_PRODUCTO) VALUES
(AUX, PRODCUTO_ID);

    RETURN RESP;

end;

SELECT COMPRAR_PRODUCTO(2,3);

#FUNCION 5
CREATE OR REPLACE FUNCTION COMPRAR_VACUNA(CLIENTE_ID INTEGER,
SERVICIO_ID INTEGER, MASCOTA_ID INTEGER)
RETURNS VARCHAR(30)
BEGIN
    DECLARE AUX INTEGER DEFAULT 0;
    DECLARE RESP VARCHAR(30) DEFAULT 'REGISTRADO';
    IF SERVICIO_ID >= 8 AND SERVICIO_ID <= 11 THEN

        INSERT INTO COMPRA(COD_CLIENTE, FECHA_COMPRA) VALUES
(CLIENTE_ID, CURDATE());

        SELECT MAX(ID_COMPRA) INTO AUX FROM COMPRA;

        INSERT INTO CONTENIDO_SERVICIOS( COD_COMPRA,
COD_SERVICIO) VALUES (AUX, SERVICIO_ID);
        INSERT INTO RECIBE_SERVICIOS(COD_MASCOTA, COD_SERVICIO)
VALUES (MASCOTA_ID, SERVICIO_ID);
        ELSE SET RESP='ID DEL SERVICIO INCORRECTO';
        END IF;

    RETURN RESP;

end;
SELECT COMPRAR_VACUNA(1,9,1);

#FUNCION 6

CREATE OR REPLACE FUNCTION COMPRAR_SERVICIO(CLIENTE_ID INTEGER,
SERVICIO_ID INTEGER)
RETURNS VARCHAR(50)
BEGIN
    DECLARE AUX INTEGER DEFAULT 0;

```

```

        DECLARE RESP VARCHAR(50) DEFAULT 'REGISTRADO';
        IF SERVICIO_ID<8 OR SERVICIO_ID>11 THEN

                INSERT INTO COMPRA(COD_CLIENTE, FECHA_COMPRA) VALUES
(CLIEN TE_ID, CURDATE());
                SELECT MAX(ID_COMPRA) INTO AUX FROM COMPRA;
                INSERT INTO CONTENIDO_SERVICIOS( COD_COMPRA,
COD_SERVICIO) VALUES (AUX,SERVICIO_ID);

        ELSE SET RESP='LOS ID DE SERVICIOS VACUNAS NO SON ACEPTADOS';
        END IF;

        RETURN RESP;
end;

SELECT COMPRAR_SERVICIO(4,2);

#////////////////////////////////////
#//////////          VISTAS          //////////////////////////////////
#////////////////////////////////////

# VISTA 1: MUESTRA ID_CLIENTE NOMBRE DEL DUEÑO NOMBRE DE LA MASCOTA
ID_MASCOTA ORDENADO POR APELLIDO PATERNOO

CREATE OR REPLACE VIEW CLIENTES_Y_MASCOTAS AS
SELECT CLIENTE.ID_CLIENTE,
        CONCAT(CLIENTE.AP_PATERNO, ' ',CLIENTE.AP_MATERNO, '
',CLIENTE.NOMBRE_CLIENTE)AS DUEÑO,
        MASCOTA.NOMBRE_MASC AS MASCOTA,
        MASCOTA.ANIMAL,
        MASCOTA.GENERO,
        MASCOTA.ID_MASCOTA
FROM CLIENTE
INNER JOIN MASCOTA ON CLIENTE.ID_CLIENTE = MASCOTA.DUEÑO
ORDER BY AP_PATERNO;

SELECT * FROM CLIENTES_Y_MASCOTAS;

#VISTA 2: MUESTRA A LA MASCOTA SU RAZA Y CUANTAS VACUNAS YA HA RECIBIDO

CREATE OR REPLACE VIEW CONTROL_DE_VACUNAS AS
SELECT MASCOTA.NOMBRE_MASC,
        ID_MASCOTA AS MASCOTA,
        DUEÑO,
        CONTROL_VACUNAS.HEXAVALENTE,
        CONTROL_VACUNAS.OCTAVALENTE,
        CONTROL_VACUNAS.PARVOVIRUS,
        CONTROL_VACUNAS.VACUNA_RABIA
FROM MASCOTA
INNER JOIN CONTROL_VACUNAS ON MASCOTA.ID_MASCOTA =
CONTROL_VACUNAS.MASCOTA;

SELECT*FROM CONTROL_VACUNAS;
#VISTA 3; MUESTRA TODOS LOS SERVICIOS COMPRADOS ORDENADOS POR LA FECHA DE
COMPRA

```

```

CREATE OR REPLACE VIEW LISTA_SERVICIOS_VENDIDOS AS
SELECT
    ID_COMPRA,
    CONCAT(CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO, '
', CLIENTE.NOMBRE_CLIENT) AS DUEÑO,
    NOMBRE_SERVICIO,
    COSTO,
    FECHA_COMPRA
FROM CONTENIDO_SERVICIOS
INNER JOIN COMPRA ON CONTENIDO_SERVICIOS.COD_COMPRA = COMPRA.ID_COMPRA
INNER JOIN CLIENTE ON COMPRA.COD_CLIENTE = CLIENTE.ID_CLIENTE
INNER JOIN SERVICIO ON CONTENIDO_SERVICIOS.COD_SERVICIO =
SERVICIO.ID_SERVICIO
ORDER BY FECHA_COMPRA;

SELECT * FROM LISTA_SERVICIOS_VENDIDOS;

#VISTA 4; MUESTRA TODOS LOS PRODUCTOS COMPRADOS ORDENADOS POR LA FECHA DE
COMPRA

CREATE OR REPLACE VIEW LISTA_PRODUCTOS_VENDIDOS AS
SELECT
    COMPRA.ID_COMPRA,
    CONCAT(CLIENTE.AP_PATERNO, ' ', CLIENTE.AP_MATERNO, '
', CLIENTE.NOMBRE_CLIENT) AS DUEÑO,
    PRODUCTO.NOMBRE_PRODUCTO,
    COSTO,
    COMPRA.FECHA_COMPRA
FROM CONTENIDO_COMPRA
INNER JOIN PRODUCTO ON CONTENIDO_COMPRA.COD_PRODUCTO =
PRODUCTO.ID_PRODUCTO
INNER JOIN COMPRA ON CONTENIDO_COMPRA.COD_COMPRA = COMPRA.ID_COMPRA
INNER JOIN CLIENTE ON CLIENTE.ID_CLIENTE = COMPRA.COD_CLIENTE
ORDER BY COMPRA.ID_COMPRA;

SELECT * FROM LISTA_PRODUCTOS_VENDIDOS;

#VISTA 5; MUESTRA LISTAS ORDENADAS POR NOMBRE DE LOS SERVICIOS Y
PRODUCTOS

CREATE OR REPLACE VIEW LISTA_PRODUTOS AS
SELECT
    NOMBRE_PRODUCTO,
    COSTO
FROM PRODUCTO
ORDER BY NOMBRE_PRODUCTO;
SELECT * FROM LISTA_PRODUTOS;

CREATE OR REPLACE VIEW LISTA_SERVICIOS AS
SELECT
    NOMBRE_SERVICIO,
    COSTO
FROM SERVICIO
ORDER BY NOMBRE_SERVICIO;

SELECT *FROM LISTA_SERVICIOS;

```

```

#////////////////////////////////////
#////////////////////      TRIGGER      //////////////////////////////////////
#////////////////////////////////////

CREATE OR REPLACE TRIGGER PONER_EDAD_MASCOTA
BEFORE INSERT ON MASCOTA
FOR EACH ROW
BEGIN
    SET
NEW.EDAD_MESES=TIMESTAMPDIFF(MONTH,NEW.FECHA_NACIMIENTO,CURDATE());
    end;

CREATE OR REPLACE TRIGGER CREAR_CONTROLVACUNAS_PARA_MASCOTAS
AFTER INSERT ON MASCOTA
FOR EACH ROW
BEGIN
    INSERT INTO CONTROL_VACUNAS( OCTAVALENTE, HEXAVALENTE,
PARVOVIRUS, VACUNA_RABIA, MASCOTA)
VALUES ('NO','NO','NO','NO',NEW.ID_MASCOTA);
    end;

#TRIGER 1: MODIFICA LA TABLA CONTROL VACUNAS, CADA VEZ QUE EL CLIENTE
COMPRA UN SERVICIO DE
#
VACUNAS PARA SU MASCOTA

CREATE OR REPLACE TRIGGER MODIFICAR_VACUNAS_RECIBIDAS
AFTER INSERT ON RECIBE_SERVICIOS
FOR EACH ROW
BEGIN
    IF NEW.COD_SERVICIO>=8 AND NEW.COD_SERVICIO<=11 THEN

    CASE
        WHEN NEW.COD_SERVICIO=8 THEN UPDATE CONTROL_VACUNAS SET
OCTAVALENTE='SI' WHERE MASCOTA=NEW.COD_MASCOTA;

        WHEN NEW.COD_SERVICIO=9 THEN UPDATE CONTROL_VACUNAS SET
HEXAVALENTE='SI' WHERE MASCOTA=NEW.COD_MASCOTA;

        WHEN NEW.COD_SERVICIO=10 THEN UPDATE CONTROL_VACUNAS SET
PARVOVIRUS='SI' WHERE MASCOTA=NEW.COD_MASCOTA;

        WHEN NEW.COD_SERVICIO=11 THEN UPDATE CONTROL_VACUNAS SET
VACUNA_RABIA='SI' WHERE MASCOTA=NEW.COD_MASCOTA;

    END CASE;
    END IF;
end;

# TRIGGER AUDITORIA 1

```

```

CREATE TABLE CAMBIOS_TABLA_PRODUCTO
(
    ID_CAMBIO_TABLA_PRECIO INTEGER AUTO_INCREMENT PRIMARY KEY,
    NOMBRE_ANTERIOR VARCHAR(40),
    NOMBRE_NUEVO VARCHAR(40),
    PRECIO_ANTERIOR REAL,
    PRECIO_NUEVO REAL,
    TIPO_ANTERIOR VARCHAR(40),
    TIPO_NUEVO VARCHAR(40),
    ID_DEL_PRODUCTO INTEGER,
    FECHA_DEL_CAMBIO DATETIME
);

CREATE OR REPLACE TRIGGER AUDI_CAMBIOS_TABLA_PRODUCTO
AFTER UPDATE ON PRODUCTO
FOR EACH ROW
BEGIN
    INSERT INTO CAMBIOS_TABLA_PRODUCTO
        (NOMBRE_ANTERIOR, NOMBRE_NUEVO, PRECIO_ANTERIOR, PRECIO_NUEVO,
        TIPO_ANTERIOR, TIPO_NUEVO, ID_DEL_PRODUCTO, FECHA_DEL_CAMBIO)
        VALUES
        (OLD.NOMBRE_PRODUCTO, NEW.NOMBRE_PRODUCTO, OLD.COSTO, NEW.COSTO, OLD.TIPO_PRODUCTO, NEW.TIPO_PRODUCTO, OLD.ID_PRODUCTO, NOW());
    end;

# TRIGGER AUDITORIA 2

CREATE TABLE CAMBIOS_TABLA_SERVICIO
(
    ID_CAMBIO INTEGER AUTO_INCREMENT PRIMARY KEY,
    NOMBRE_ANTERIOR VARCHAR(40),
    NOMBRE_NUEVO VARCHAR(40),
    PRECIO_ANTERIOR REAL,
    PRECIO_NUEVO REAL,
    ID_DEL_SERVICIO INTEGER,
    FECHA_DEL_CAMBIO DATETIME
);

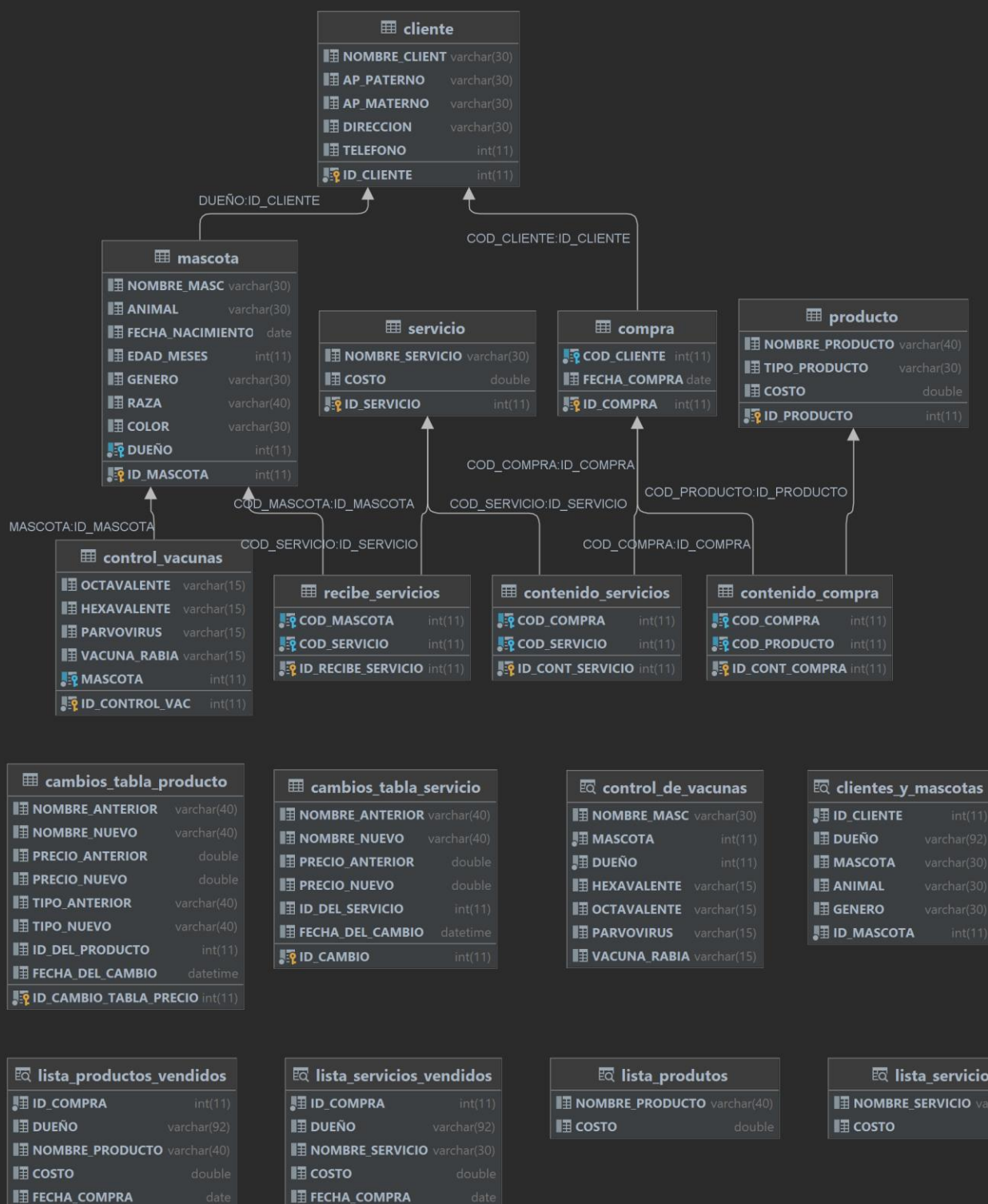
CREATE OR REPLACE TRIGGER AUDI_CAMBIOS_TABLA_SERVICIO
AFTER UPDATE ON SERVICIO
FOR EACH ROW
BEGIN
    INSERT INTO CAMBIOS_TABLA_SERVICIO
        (NOMBRE_ANTERIOR, NOMBRE_NUEVO, PRECIO_ANTERIOR, PRECIO_NUEVO,
        ID_DEL_SERVICIO, FECHA_DEL_CAMBIO)
        VALUES (OLD.NOMBRE_SERVICIO, NEW.NOMBRE_SERVICIO, OLD.COSTO, NEW.COSTO, OLD.ID_SERVICIO, NOW());
    end;

SELECT * FROM CAMBIOS_TABLA_SERVICIO;

```

## MODELO LOGICO

Enlace: [https://github.com/JHOMARHUAYCHO/BASE-DE-DATOS-2/blob/main/HITO%205/PROYECTO%20FINAL%20DB\\_VETERINARIA/DB\\_VETERINARIA%20-%20MODELO%20LOGICO.png](https://github.com/JHOMARHUAYCHO/BASE-DE-DATOS-2/blob/main/HITO%205/PROYECTO%20FINAL%20DB_VETERINARIA/DB_VETERINARIA%20-%20MODELO%20LOGICO.png)





### 3. UTILIDAD IMÁGENES DEL FUNCIONAMIENTO

#### IMAGEN 1

Trigger de control funciona correctamente, este trigger coloca “SI” si el cliente a comprado la vacuna para su mascota, al mismo tiempo se muestra la **vista** que nos muestra la información

The screenshot displays the SQL Server Enterprise Manager interface. The left pane shows the database structure, including the 'db\_veterinaria' database and its 'views' folder. The right pane shows the 'control\_de\_vacunas' trigger, which is a 'BEFORE' trigger on the 'clientes\_y\_mascotas' table. The trigger logic checks if a vaccine has been purchased for a specific client and pet, and if so, it updates the 'vacunas' column in the 'clientes\_y\_mascotas' table to 'SI'.

The central pane shows the SQL query for the trigger, which is a SELECT statement that retrieves the client name, pet name, owner name, and vaccine status for all records in the 'clientes\_y\_mascotas' table. The results are displayed in a table with 8 rows.

ID	NOMBRE_MASC	MASCOTA	DUÑO	HEXAVALENTE	OCTAVALENTE	PARVOVIRUS	VAC
1	TIITO	1	1	NO	NO	NO	NO
2	PELUSA	2	2	NO	NO	NO	NO
3	WOOPER	3	4	NO	SI	NO	NO
4	SNOWY	4	3	NO	SI	SI	NO
5	FROGGY	5	5	NO	NO	SI	NO
6	TOTO	6	6	NO	NO	NO	SI
7	SASI	7	1	NO	NO	NO	NO
8	MISHA	8	3	NO	NO	SI	NO

The bottom pane shows the SQL query for the view, which is a SELECT statement that retrieves the client name, pet name, owner name, and vaccine status for all records in the 'clientes\_y\_mascotas' table. The results are displayed in a table with 8 rows.

ID	NOMBRE_MASC	MASCOTA	DUÑO	HEXAVALENTE	OCTAVALENTE	PARVOVIRUS	VAC
1	TIITO	1	1	NO	NO	NO	NO
2	PELUSA	2	2	NO	NO	NO	NO
3	WOOPER	3	4	NO	SI	NO	NO
4	SNOWY	4	3	NO	SI	SI	NO
5	FROGGY	5	5	NO	NO	SI	NO
6	TOTO	6	6	NO	NO	NO	SI
7	SASI	7	1	NO	NO	NO	NO
8	MISHA	8	3	NO	NO	SI	NO

## IMAGEN DE LAS VISTAS

### vista de las ventas realizadas

Database Explorer: @localhost [2] db\_veterinaria > views > lista\_productos\_vendidos

Query: `SELECT t.* FROM db_veterinaria.lista_productos_vendidos t LIMIT 501`

Results: 15 rows retrieved starting from 1 in 73 ms (execution: 11 ms, fetching: 62 ms)

ID_COMpra	DUEÑO	NOMBRE_PRODUCTO	COSTO	FECHA_COMPRA
1	SAMUDIO LOPEZ LOREANA	COLLAR	15	2021-11-11
2	SAMUDIO LOPEZ LOREANA	RICO CAT SALMON	7	2021-11-11
3	SAMUDIO LOPEZ LOREANA	RICO CAT CARNE	8	2021-11-11
4	TORRES ROCHA CLAUDIA	RICO CAT SALMON	7	2021-11-11
5	TORRES ROCHA CLAUDIA	HUESO MASTICABLE	26	2021-11-11
6	VILLCA IMACA LEONARDO	RICO CRACK CARNE	22	2021-11-14
7	FERNANDEZ OBRADOR RODRIGO	RICO CAT CARNE	8	2021-12-20
8	FERNANDEZ OBRADOR RODRIGO	RICO CAT SALMON	7	2021-12-20
9	VILLCA IMACA LEONARDO	OSPRET SHAMPOO	65	2021-12-25
10	LUARTE PEREZ KARLA	COMEDOR DOGIFICADOR DE AGUA	64	2022-12-29
11	LUARTE PEREZ KARLA	TAZON PARA GATO	16	2022-12-29
12	GONZALES SURITA ROBERTO	RICO CAT CARNE	8	2022-01-02
13	GONZALES SURITA ROBERTO	RICO CAT SALMON	7	2022-01-02
14	TORRES ROCHA CLAUDIA	RICO CRACK CARNE	22	2022-01-02
15	TORRES ROCHA CLAUDIA	BEBEDERO PORTATIL	35	2022-01-02

### Vista de los clientes y sus mascotas

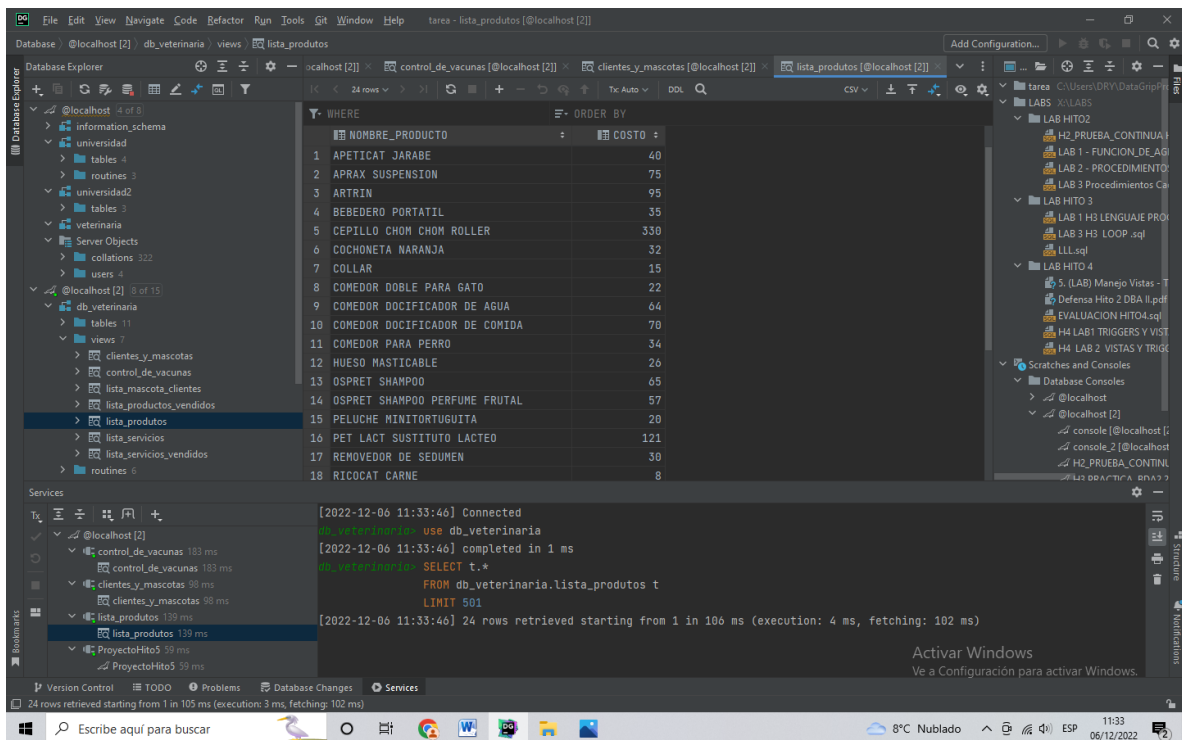
Database Explorer: @localhost [2] db\_veterinaria > views > clientes\_y\_mascotas

Query: `SELECT t.* FROM db_veterinaria.clientes_y_mascotas t LIMIT 501`

Results: 8 rows retrieved starting from 1 in 58 ms (execution: 4 ms, fetching: 54 ms)

ID_CLIENTE	DUEÑO	MASCOTA	ANIMAL	GENERO	ID_MASCOTA
1	FERNANDEZ OBRADOR RODRIGO	TOTO	PERRO	MACHO	6
2	GONZALES SURITA ROBERTO	PELUSA	GATO	HEMBRA	2
3	LUARTE PEREZ KARLA	FROGGY	PERRO	MACHO	7
4	SAMUDIO LOPEZ LOREANA	SASI	GATO	HEMBRA	5
5	SAMUDIO LOPEZ LOREANA	TITO	PERRO	MACHO	1
6	TORRES ROCHA CLAUDIA	SNOWY	GATO	HEMBRA	4
7	TORRES ROCHA CLAUDIA	MISHA	GATO	HEMBRA	8
8	VILLCA IMACA LEONARDO	WOOPER	PERRO	MACHO	3

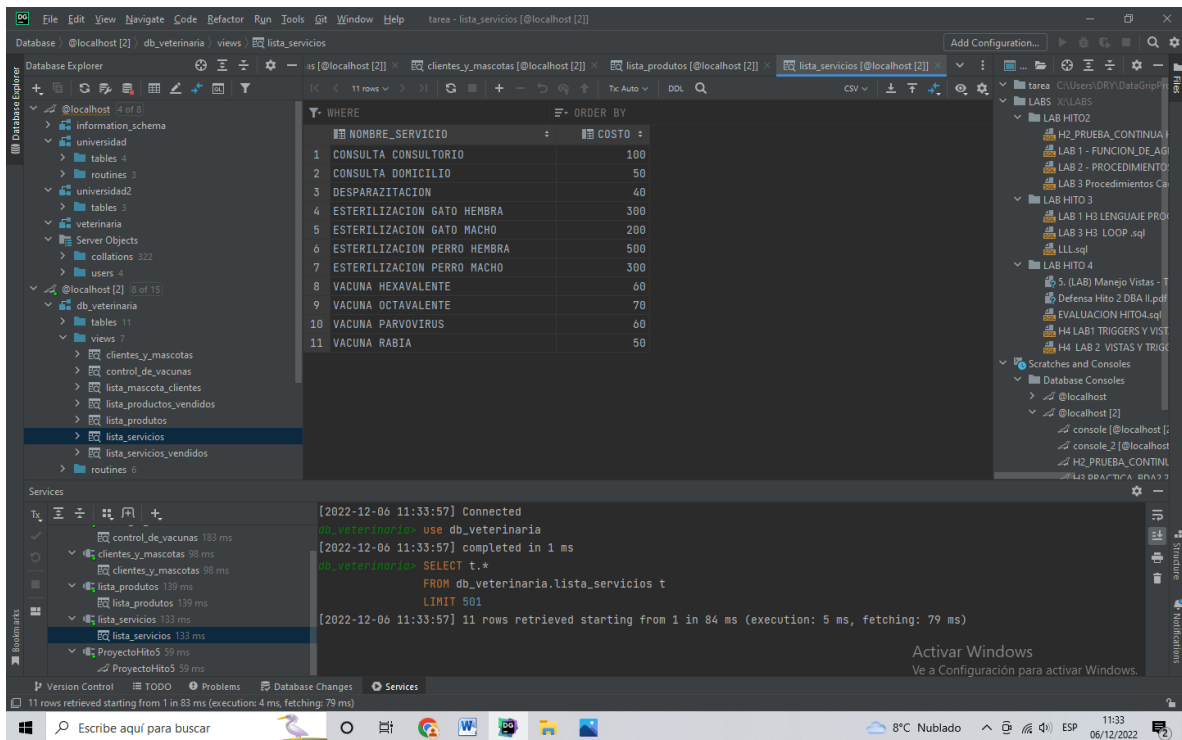
## Vista de los productos



The screenshot shows the SQL Server Enterprise Manager interface. The 'Database Explorer' pane on the left shows the 'db\_veterinaria' database structure. The 'Views' pane shows the 'lista\_productos' view. The 'Query Results' pane displays the following data:

NOMBRE_PRODUCTO	COSTO
1 APETICAT JARABE	40
2 APRAX SUSPENSION	75
3 ARTRIN	95
4 BEBEDERO PORTATIL	35
5 CEPILLO CHOM CHOM ROLLER	330
6 COCHONETA NARANJA	32
7 COLLAR	15
8 COMEDOR DOBLE PARA GATO	22
9 COMEDOR DOCEFICADOR DE AGUA	64
10 COMEDOR DOCEFICADOR DE COMIDA	70
11 COMEDOR PARA PERRO	34
12 HUESO MASTICABLE	26
13 OSPRET SHAMPOO	65
14 OSPRET SHAMPOO PERFUME FRUTAL	57
15 PELUCHE MINITORTUGUITA	20
16 PET LACT SUSTITUTO LACTEO	121
17 REMOVEDOR DE SEDUNEN	30
18 RICOCAT CARNE	8

## Vista de los servicios

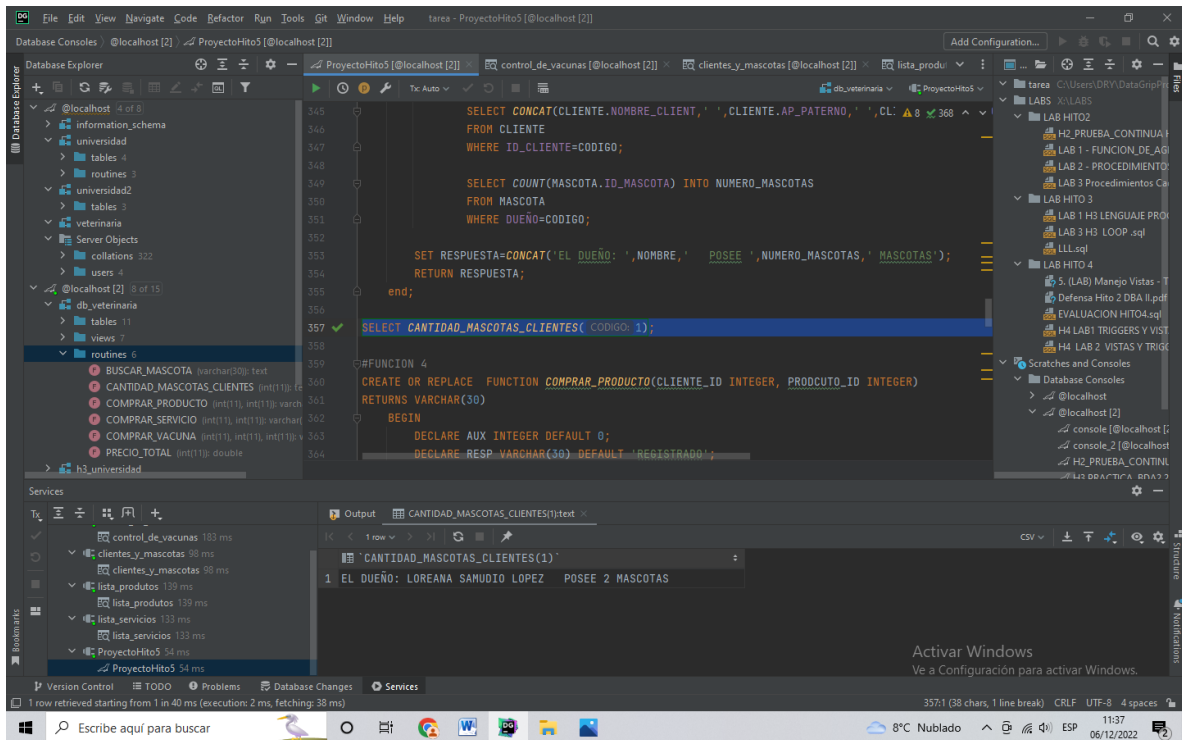


The screenshot shows the SQL Server Enterprise Manager interface. The 'Database Explorer' pane on the left shows the 'db\_veterinaria' database structure. The 'Views' pane shows the 'lista\_servicios' view. The 'Query Results' pane displays the following data:

NOMBRE_SERVICIO	COSTO
1 CONSULTA CONSULTORIO	100
2 CONSULTA DOMICILIO	50
3 DESPARAZITACION	40
4 ESTERILIZACION GATO HEMBRA	300
5 ESTERILIZACION GATO MACHO	200
6 ESTERILIZACION PERRO HEMBRA	500
7 ESTERILIZACION PERRO MACHO	300
8 VACUNA HEXAVALENTE	60
9 VACUNA OCTAVALENTE	70
10 VACUNA PARVOVIRUS	60
11 VACUNA RABIA	50

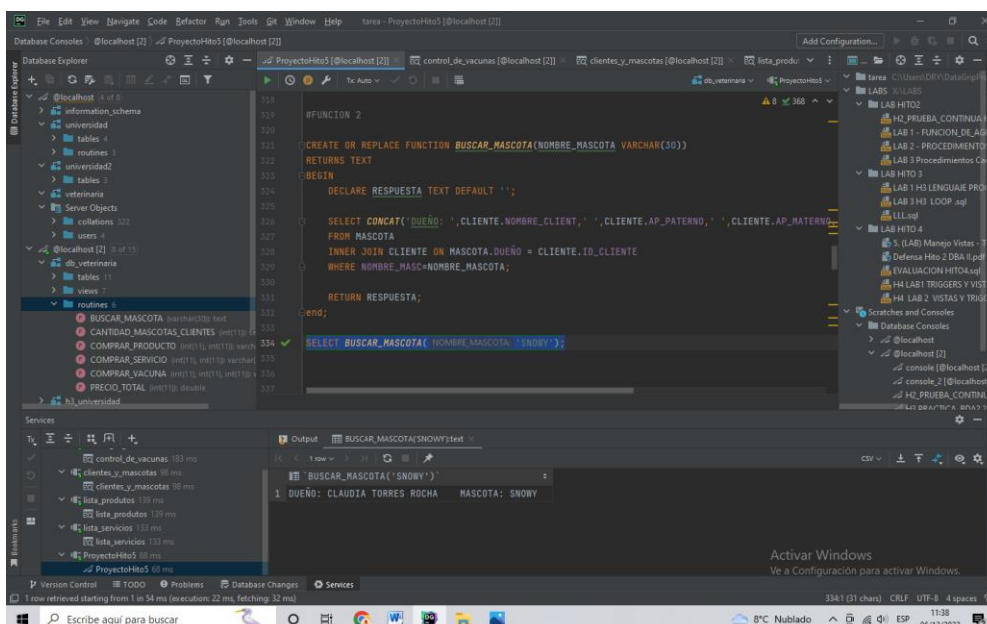
## IMAGEN 3

Correcto funcionamiento de la función CANTIDAD\_MASCOTAS\_CLIENTES la función recibe el ID del cliente y devuelve la cantidad de mascotas de ese cliente



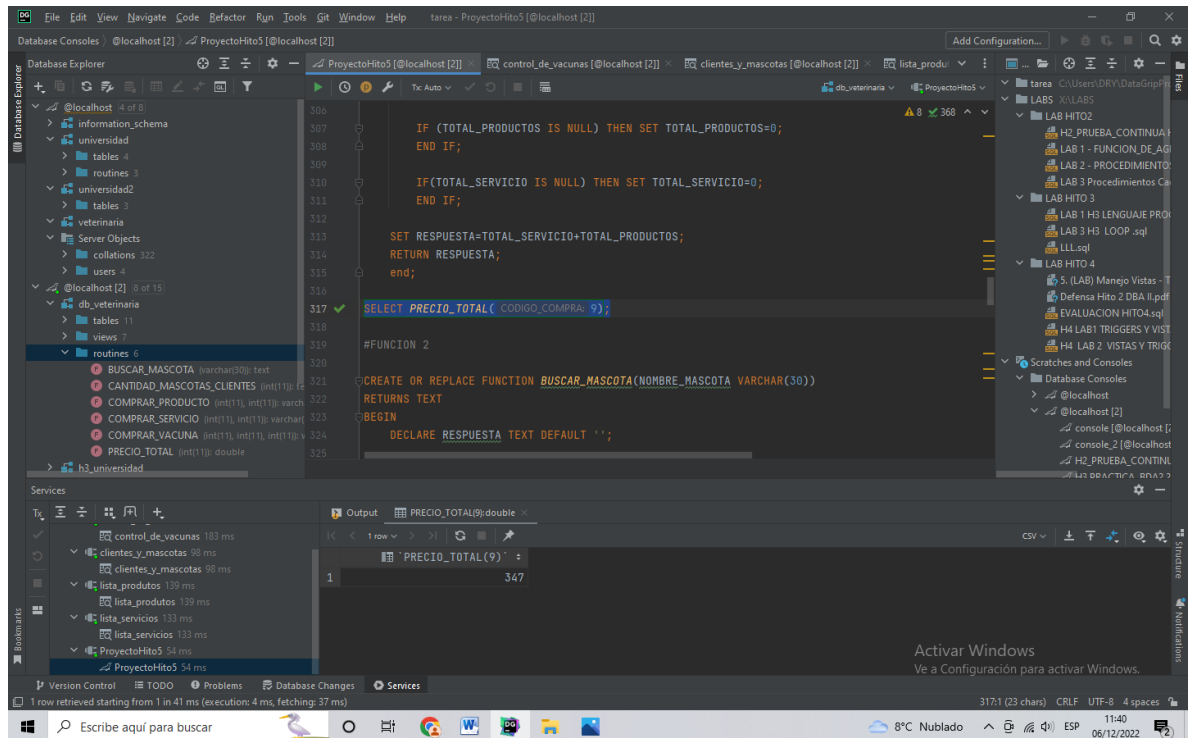
## IMAGEN 4

Correcto funcionamiento de la función BUSCAR\_MASCOTA la función recibe el nombre de la mascota, y si la mascota existe muestra los datos correspondientes



## IMAGEN 4

Correcto funcionamiento de la función PRECIO\_TOTAL, esta función recibe el código de la compra y devuelve el precio total de los productos y servicios que corresponden a esa venta



## VIDEO

ENLACE DEL VIDEO DRIVE: [https://drive.google.com/drive/folders/1YoLy\\_-9Xh\\_kAqubL-hdoW1vPBKoPFdKU?usp=share\\_link](https://drive.google.com/drive/folders/1YoLy_-9Xh_kAqubL-hdoW1vPBKoPFdKU?usp=share_link)

#### 4. CONCLUSIONES

- Para una buena práctica, se debe particionar la tabla CLIENTE y la tabla MASCOTA las cuales tienen muchos datos en otras como por ejemplo: DIRECCIÓN, CARACTERÍSTICAS\_MASCOTA, etc.
- se debe buscar y analizar el nombre que tendrán las tablas, vistas y funciones para comprender mejor la base de datos.
- Cuando hay muchas relaciones M:N y por ende muchas tablas de unión con varios IDs como consecuencia la inserción de datos desde 'consola' es tremendamente complicada (como ocurre en el caso de comprar una vacuna para la mascota) por lo que es mejor crear una **función** para facilitar la inserción de datos
- Cuando hay varias tablas relacionadas, las **vistas** son de gran ayuda al momento de comprobar que los trigger y funciones funcionen correctamente