CREATE TABLE pasajero(

id serial,

nombre character varying(100),

direccion\_residencia character varying,

fecha\_nacimiento date,

CONSTRAINT pasajero\_pkey PRIMARY KEY (id)

)

CREATE TABLE viaje(

id serial,

id\_pasajero integer,

id\_trayecto integer,

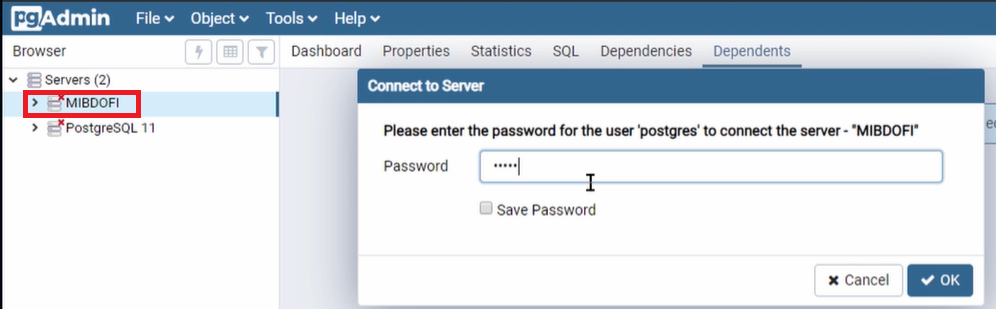
inicio date,

fin date,

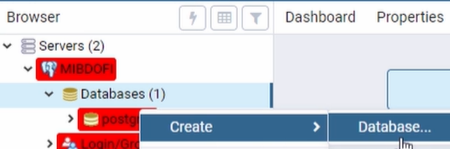
CONSTRAINT viaje\_pkey PRIMARY KEY (id),

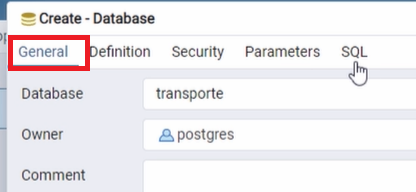
FOREIGN KEY (id\_pasajero) REFERENCES pasajero (id)

)

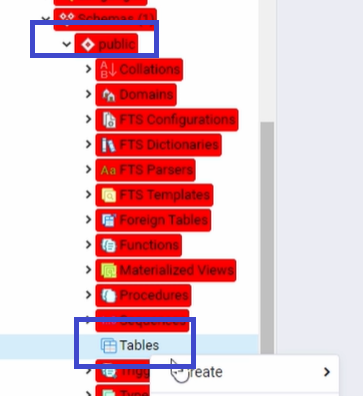


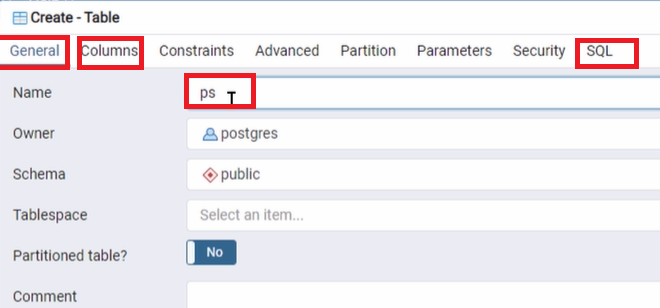
Cree base de datos

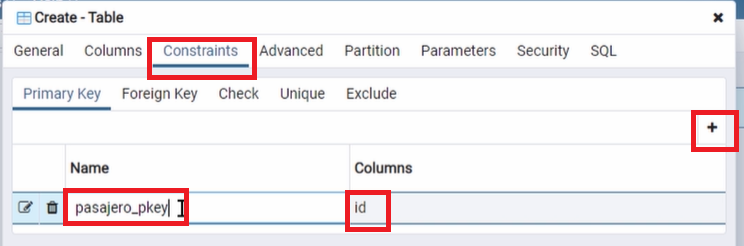




Escoge “Public” / Tables / Create

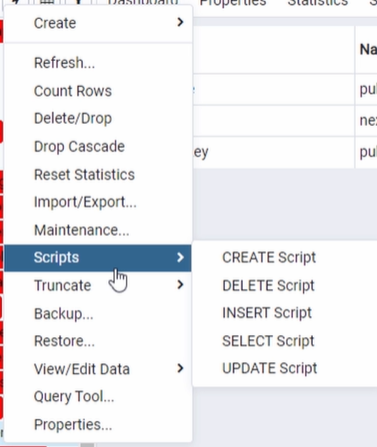




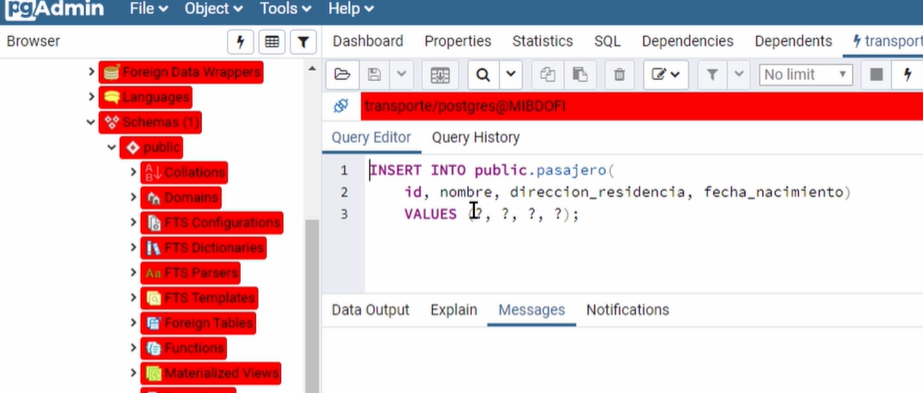




O por script



Script de insersion



**Particiones**

- Separacion fisica de datos

- Mantiene estructura logica

-- Tabla bitacora\_viaje

CREATE TABLE **public.bitacora\_viaje**(

id serial,

id\_viaje integer,

fecha date

) PARTITION BY RANGE (fecha)

WITH (OIDS = FALSE);

ALTER TABLE public.bitacora\_viaje OWNER to postgres;

-- Vamos a crear la particion

CREATE TABLE bitacora\_viaje201001

PARTITION OF **public.bitacora\_viaje**

FOR VALUES FROM ('2010-01-01') TO ('2020-04-24');

-- Insertamos un dato --> ERROR antes de crear la particions

INSERT INTO public.bitacora\_viaje(id\_viaje, fecha)

VALUES(1, '2010-01-01');

-- Mostramos los valores de la tabla bitacora\_viaje (toma todas las particiones)

SELECT \* FROM bitacora\_viaje;

--Si solo queremos ENERO

SELECT \* FROM bitacora\_viaje\_201001;

Ventaja de las tablas particionadas es que puedes utilizar la sentencia TRUNCATE, la cual elimina toda la información de una tabla, pero a nivel partición.

Es decir, si tienes una tabla con 12 particiones (1 para cada mes del año) y deseas eliminar toda la información del mes de Enero; con la sentencia ALTER TABLE tabla TRUNCATE PARTITION enero; podrías eliminar dicha información sin afectar el resto.