

```
postgres=# CREATE DATABASE laboratorioSQL;
CREATE DATABASE
```

```
postgres=# \c laboratoriosql
You are now connected to database "laboratoriosql" as user "postgres".
laboratoriosql=#
```

```
postgres=# CREATE TABLE profesores(
postgres(# doc_prof varchar(11) not null,
postgres(# nom_prof varchar(30) not null,
postgres(# ape_prof varchar(30) not null,
postgres(# cate_prof int not null,
postgres(# sal_prof int not null,
postgres(# primary key(doc_prof)
postgres(# );
CREATE TABLE
```

```
postgres=# CREATE TABLE cursos(
postgres(# cod_curs SERIAL PRIMARY KEY,
postgres(# nom_curs varchar(100) NOT NULL,
postgres(# horas_cur int NOT NULL,
postgres(# valor_cur int NOT NULL);
CREATE TABLE
```

```
postgres=# CREATE TABLE estudiantes(
postgres(# doc_est varchar(11) NOT NULL,
postgres(# nom_est varchar(30) NOT NULL,
postgres(# ape_est varchar(30) NOT NULL,
postgres(# edad_est INT NOT NULL,
postgres(# PRIMARY KEY(doc_est)
postgres(# );
CREATE TABLE
```

```
postgres=# CREATE TABLE estudiantesxcursos(
postgres(# cod_cur_estcur int not null,
postgres(# doc_est_estcur varchar(30) not null,
postgres(# fec_ini_estcur date not null,
postgres(# constraint fk_est_cur foreign key (cod_cur_estcur) references
cursos(cod_curs),
postgres(# constraint fk_est_est foreign key (doc_est_estcur) references
estudiantes(doc_est));
CREATE TABLE
```

Nota: la tabla estudiantes tiene doc_est de tipo varchar y estudiantesxcurso tiene doc_est_estcur de tipo int. Además la tabla estudiantexcurso le da longitud a int, entonces no es claro si los atributos de las tablas deben ser de tipo int o varchar.

```
postgres=# CREATE TABLE clientes(
postgres(# id_cli varchar(11) not null,
postgres(# nom_cli varchar(30) not null,
postgres(# ape_cli varchar(30) not null,
postgres(# dir_cli varchar(100) not null,
postgres(# dep_cli varchar(20) not null,
```

```
postgres(# mes_cum_cli varchar(10) not null,  
postgres(# primary key(id_cli)  
postgres(# );  
CREATE TABLE
```

```
postgres=# CREATE TABLE articulos(  
postgres(# id_art serial not null,  
postgres(# tit_art varchar(100) not null,  
postgres(# aut_art varchar(100) not null,  
postgres(# edi_art varchar(300) not null,  
postgres(# prec_art int not null,  
postgres(# primary key(id_art));  
CREATE TABLE
```

```
postgres=# CREATE TABLE pedidos(  
postgres(# id_ped serial not null,  
postgres(# id_cli_ped varchar(22) not null,  
postgres(# fec_ped date not null,  
postgres(# val_ped int not null,  
postgres(# constraint fk_cli_ped foreign key (id_cli_ped) references  
clientes(id_cli),  
postgres(# primary key(id_ped)  
);  
CREATE TABLE
```

```
postgres=# CREATE TABLE articulosxpedidos(  
postgres(# id_ped_artped int not null,  
postgres(# id_art_artped int not null,  
postgres(# can_art_artped int not null,  
postgres(# val_ven_art_artped int not null,  
postgres(# constraint fk_ped_art foreign key(id_ped_artped) references  
pedidos(id_ped),  
postgres(# constraint fk_art_art foreign key (id_art_artped) references  
articulos(id_art));  
CREATE TABLE
```

```
postgres=# CREATE TABLE compania(  
postgres(# comnit varchar(11) not null,  
postgres(# comnombre varchar(30) not null,  
postgres(# comañofun int not null,  
postgres(# comreplegal varchar(100) not null,  
postgres(# primary key(comnit));  
CREATE TABLE
```

```
postgres=# CREATE TABLE tiposAutomotores(  
postgres(# auttipo int not null,  
postgres(# primary key(auttipo));  
CREATE TABLE
```

```
postgres=# CREATE TABLE automotores(  
postgres(# autoplaca varchar(6) not null,  
postgres(# automarca varchar(30) not null,  
postgres(# autotipo int not null,  
postgres(# automodelo int not null,
```

```

postgres(# autopasajeros int not null,
postgres(# autocilindraje int not null,
postgres(# autonumchasis varchar(20) not null,
postgres(# constraint fk_aut_aut foreign key (autotipo) references
tiposautomotores(auttipo),
postgres(# primary key(autoplaca));
CREATE TABLE

```

```

postgres=# CREATE TABLE aseguramientos(
postgres(# asecodigo serial not null,
postgres(# asefechainicio date not null,
postgres(# asefechaexpiracion date not null,
postgres(# asevalorasegurado int not null,
postgres(# aseestado varchar(30) not null,
postgres(# asecosto int not null,
postgres(# aseplaca varchar(6) not null,
postgres(# constraint fk_aut_ase foreign key (aseplaca) references
automotores(autoplaca),
postgres(# primary key(asecodigo));
CREATE TABLE

```

```

postgres=# CREATE TABLE incidentes(
postgres(# incicodigo serial not null,
postgres(# incifecha date not null,
postgres(# inciplaca varchar(6) not null,
postgres(# incilugar varchar(40) not null,
postgres(# inciantheridos int not null,
postgres(# incicanfatalidades int not null,
postgres(# incicanautosinvolucrados int not null,
postgres(# constraint fk_aut_inci foreign key (inciplaca) references
automotores(autoplaca),
postgres(# primary key(incicodigo));
CREATE TABLE

```

```

postgres=# INSERT INTO profesores (doc_prof, nom_prof, ape_prof,
cate_prof, sal_prof)
postgres-# VALUES
postgres-# ('CC123', 'Julio', 'Castaño', 2, 3000000),
postgres-# ('CC456', 'Miguel', 'Mondragon', 1, 2000000),
postgres-# ('CC789', 'Yenifer', 'Orozco', 1, 4000000),
postgres-# ('CC098', 'Angel', 'Lopez', 1, 1000000),
postgres-# ('CC765', 'Nicolas', 'Moreno', 1, 500000),
postgres-# ('CC432', 'Marcela', 'Saavedra', 2, 350000);
INSERT 0 6

```

```

postgres=# INSERT INTO cursos (nom_curs, horas_cur, valor_cur)
postgres-# VALUES ('Gestión Bases de Datos', 2, 1500000),
postgres-# ('Introducción a la Programación', 3, 2000000),
postgres-# ('Emprendimiento', 2, 2000000),
postgres-# ('Seguridad y Salud en el Trabajo', 2, 500000),
postgres-# ('Introducción a los Lenguajes de Programación', 1, 250000),
postgres-# ('Lógica de Programación', 1, 100000);
INSERT 0 6

```

Nota: no es necesario incluir el código de curso porque es de tipo serial y se asignará e incrementará automáticamente.

```
postgres=# INSERT INTO estudiantes(doc_est, nom_est, ape_est, edad_est)
postgres=# VALUES
postgres=# ('CC123', 'Laura', 'Barona Saavedra', 25),
postgres=# ('CC456', 'Mariana', 'Gomez', 28),
postgres=# ('CC789', 'Sara', 'Lopez', 27),
postgres=# ('CC098', 'Jose', 'Daza', 24),
postgres=# ('CC765', 'Jhon', 'Arias', 25),
postgres=# ('CC432', 'Jason', 'Martinez', 30);
INSERT 0 6
```

```
postgres=# INSERT INTO estudiantesxcursos(cod_cur_estcur, doc_est_estcur,
fec_ini_estcur)
postgres=# VALUES
postgres=# (1, 'CC123', '2024-08-02');
INSERT 0 1
```

Nota: como son claves foráneas no deben sobrecribirse, deben tener los mismos datos de la clave primaria.

```
postgres=# INSERT INTO estudiantesxcursos(cod_cur_estcur, doc_est_estcur,
fec_ini_estcur)
postgres=# VALUES
postgres=# (2, 'CC456', '2024-08-06'),
postgres=# (3, 'CC789', '2024-08-06'),
postgres=# (4, 'CC098', '2024-08-10'),
postgres=# (5, 'CC765', '2024-09-04'),
postgres=# (6, 'CC432', '2024-03-25');
INSERT 0 5
```

```
postgres=# INSERT INTO clientes(id_cli, nom_cli, ape_cli, dir_cli,
dep_cli, mes_cum_cli)
postgres=# VALUES
postgres=# ('CC135', 'Margarita', 'Ramirez', 'Calle Reforma 123, Int. 2,
Colonia Juárez, C.P. 06600, Alcaldía Cuauhtémoc', 'Valle', 'Diciembre'),
postgres=# ('CC798', 'Rosa', 'Gonzalez', 'Calle Insurgentes 456, Int. 5,
Colonia Roma Norte, C.P. 06700, Alcaldía Cuauhtémoc', 'Valle', 'Enero'),
postgres=# ('CC864', 'Elena', 'Galeano', 'Calle Hidalgo 789, Int. 10,
Colonia Centro, C.P. 06000, Alcaldía Cuauhtémoc', 'Valle', 'Noviembre'),
postgres=# ('CC246', 'Gabriela', 'Murillo', 'Calle Benito Juárez 234,
Int. 3, Colonia Del Valle, C.P. 03100, Alcaldía Benito Juárez', 'Valle',
'Enero'),
postgres=# ('CC806', 'Vilma', 'Murgueitio', 'Calle Álvaro Obregón 678,
Int. 8, Colonia Condesa, C.P. 06140, Alcaldía Cuauhtémoc', 'Valle',
'Agosto'),
postgres=# ('CC423', 'Sebastián', 'Gomez', 'Calle Reforma 123, Int. 2,
Colonia Juárez, C.P. 06600, Alcaldía Cuauhtémoc', 'Valle', 'Diciembre');
INSERT 0 6
```

```
postgres=# INSERT INTO articulos(tit_art, aut_art, edi_art, prec_art)
postgres=# VALUES
postgres=# ('computador', 'asus', 'X441U', 2300000),
```

```

postgres-# ('celular', 'samsung', 'A31', 700000),
postgres-# ('iPad', 'Apple', 'Pro', 5000000),
postgres-# ('airpods', 'apple', 'Pro 2', 2999000), ('tablet', 'lenovo',
'M10', 600000),
postgres-# ('tablet', 'huawei', 'MatePad', 830000);
INSERT 0 6

```

```

postgres=# INSERT INTO pedidos(id_cli_ped, fec_ped, val_ped)
postgres-# VALUES
postgres-# ('CC135', '2023-05-15', 3000000),
postgres-# ('CC798', '2023-10-17', 5000000),
postgres-# ('CC864', '2023-12-23', 4000000),
postgres-# ('CC246', '2024-01-10', 2000000),
postgres-# ('CC806', '2024-03-25', 2000000),
postgres-# ('CC423', '2024-02-12', 2000000);
INSERT 0 6

```

```

postgres=# INSERT INTO articulosxpedidos(id_ped_artped, id_art_artped,
can_art_artped, val_ven_art_artped)
postgres-# VALUES
postgres-# (1, 1, 3, 1500000),
postgres-# (2, 2, 4, 200000),
postgres-# (3, 3, 1, 1000000),
postgres-# (4, 4, 2, 300000),
postgres-# (5, 5, 3, 900000),
postgres-# (6, 6, 1, 4000000);
INSERT 0 6

```

```

postgres=# INSERT INTO compañía(comnit, comnombre, comañofun,
comreplegal)
postgres-# VALUES
postgres-# ('899999999-5', 'compañía 1', 2002, 'legal compañía 1'),
postgres-# ('0000000001-1', 'compañía 2', 2001, 'legal compañía 2'),
postgres-# ('0000000002-1', 'compañía 3', 2003, 'legal compañía 3'),
postgres-# ('0000000003-1', 'compañía 4', 1999, 'legal compañía 4'),
postgres-# ('0000000004-1', 'compañía 5', 2004, 'legal compañía 5'),
postgres-# ('0000000005-1', 'compañía 6', 2000, 'legal compañía 6');
INSERT 0 6

```

```

INSERT INTO tiposautomotores(auttipo)
postgres-# VALUES
postgres-# (1),
postgres-# (2),
postgres-# (3),
postgres-# (4),
postgres-# (5),
postgres-# (6);
INSERT 0 6

```

```

postgres=# INSERT INTO automotores (autoplaca, automarca, autotipo,
automodelo, autopasajeros, autocilindraje, autonumchasis)
postgres-# VALUES
postgres-# ('QGX767', 'toyota', 1, 2020, 5, 2000, 'XYZ987654321'),
postgres-# ('DEF456', 'honda', 2, 2018, 4, 1800, 'ABC123987654'),

```

```

postgres=# ('GHI789', 'chevrolet', 3, 2021, 5, 2500, 'LMN456123987'),
postgres=# ('JKL012', 'ford', 4, 2019, 2, 3000, 'OPQ789654123'),
postgres=# ('MNO345', 'mazda', 5, 2022, 4, 1600, 'DEF987654321'),
postgres=# ('PQR678', 'nissan', 6, 2020, 5, 2200, 'RST321987654');
INSERT 0 6

```

```

postgres=# INSERT INTO aseguramientos (asecodigo, asefechainicio,
asefechaexpiracion, asevalorasegurado, aseestado, aseplaca)
postgres=# VALUES
postgres=# (1, '2023-01-01', '2024-01-01', 50000000, 'Activo', 1200000,
'QGX767'),
postgres=# (2, '2022-07-15', '2023-07-15', 45000000, 'Expirado', 1100000,
'DEF456'),
postgres=# (3, '2023-03-10', '2024-03-10', 60000000, 'Activo', 1300000,
'GHI789'),
postgres=# (4, '2021-11-25', '2022-11-25', 55000000, 'Expirado', 1150000,
'JKL012'),
postgres=# (5, '2023-05-20', '2024-05-20', 70000000, 'Activo', 1400000,
'MNO345'),
postgres=# (6, '2023-09-01', '2024-09-01', 48000000, 'Activo', 1250000,
'PQR678');
INSERT 0 6

```

```

postgres=# INSERT INTO incidentes (incifecha, inci placa, incilugar,
incicantheridos, incicanfatalidades, incicanautosinvolucrados)
postgres=# VALUES
postgres=# ('2023-09-15', 'QGX767', 'Avenida Siempre Viva', 2, 0, 3),
postgres=# ('2023-08-22', 'DEF456', 'Calle Falsa 123', 1, 1, 2),
postgres=# ('2023-07-10', 'GHI789', 'Boulevard del Sol', 0, 0, 1),
postgres=# ('2023-06-30', 'JKL012', 'Carretera Panamericana', 3, 1, 4),
postgres=# ('2023-05-18', 'MNO345', 'Ruta Nacional 5', 0, 0, 2),
postgres=# ('2023-04-25', 'PQR678', 'Avenida Las Palmas', 1, 0, 3);
INSERT 0 6

```

```

postgres=# SELECT sal_prof, cate_prof FROM profesores ORDER BY cate_prof;
sal_prof | cate_prof
-----+-----
2000000 | 1
4000000 | 1
1000000 | 1
500000 | 1
3000000 | 2
350000 | 2
(6 rows)

```

```

postgres=# SELECT nom_curs, valor_cur FROM cursos WHERE valor_cur >
500000;
nom_curs | valor_cur
-----+-----
Gestión Bases de Datos | 1500000
Introducción a la Programación | 2000000
Emprendimiento | 2000000
(3 rows)

```

```

postgres=# SELECT COUNT(edad_est) FROM estudiantes
WHERE edad_est > 22;
count
-----
      6
(1 row)

```

```

postgres=# SELECT nom_est, edad_est FROM estudiantes
postgres=# WHERE edad_est = (
postgres=# SELECT
postgres=# MIN(edad_est)
postgres=# FROM
postgres=# estudiantes);
 nom_est | edad_est
-----+-----
    Jose |        24
(1 row)

```

```

postgres=# UPDATE cursos
postgres=# SET horas_cur = 40
postgres=# WHERE cod_curs = 1;
UPDATE 1
postgres=# UPDATE cursos
postgres=# SET horas_cur = horas_cur * 10;
UPDATE 6
postgres=# UPDATE cursos
postgres=# SET horas_cur = 40
postgres=# WHERE cod_curs = 1;
UPDATE 1
postgres=# UPDATE cursos
postgres=# SET horas_cur = horas_cur * 2;
UPDATE 6

```

Nota: actualización de valores para realizar promedio

```

postgres=# SELECT AVG(valor_cur) FROM cursos
postgres=# WHERE horas_cur > 40;
avg
-----
1750000.000000000000
(1 row)

```

```

postgres=# SELECT AVG(sal_prof) FROM profesores
WHERE cate_prof = 1;
avg
-----
1875000.000000000000
(1 row)

```

```

postgres=# SELECT * FROM cursos
postgres=# ORDER BY valor_cur;

```

```

cod_curs | nom_curs | horas_cur |
valor_cur

```

```

-----+-----+-----+-----
-----
        6 | Lógica de Programación | 20 |
100000
        5 | Introducción a los Lenguajes de Programación | 20 |
250000
        4 | Seguridad y Salud en el Trabajo | 40 |
500000
        1 | Gestión Bases de Datos | 80 |
1500000
        2 | Introducción a la Programación | 60 |
2000000
        3 | Emprendimiento | 40 |
2000000
(6 rows)

```

Nota: ASC es opcional ya que la ordenación por defecto es ascendente.

```

postgres=# SELECT nom_prof, ape_prof FROM profesores
postgres=# WHERE sal_prof in(SELECT MIN(sal_prof) FROM profesores);
 nom_prof | ape_prof
-----+-----
Marcela  | Saavedra
(1 row)

```

```

postgres=# SELECT * FROM profesores
postgres=# WHERE (sal_prof >= 500000) AND (sal_prof < 700000);
 doc_prof | nom_prof | ape_prof | cate_prof | sal_prof
-----+-----+-----+-----+-----
CC765    | Nicolas | Moreno  |           | 500000
(1 row)

```

```

postgres=# SELECT id_ped, id_cli_ped, fec_ped, tit_art FROM articulos,
pedidos, articulosxpedidos
postgres=# WHERE id_art_artped = id_art;

```

```

 id_ped | id_cli_ped | fec_ped | tit_art
-----+-----+-----+-----
1 | CC135 | 2023-05-15 | computador
2 | CC798 | 2023-10-17 | computador
3 | CC864 | 2023-12-23 | computador
4 | CC246 | 2024-01-10 | computador
5 | CC806 | 2024-03-25 | computador
6 | CC423 | 2024-02-12 | computador
1 | CC135 | 2023-05-15 | celular
2 | CC798 | 2023-10-17 | celular
3 | CC864 | 2023-12-23 | celular
4 | CC246 | 2024-01-10 | celular
5 | CC806 | 2024-03-25 | celular
6 | CC423 | 2024-02-12 | celular
1 | CC135 | 2023-05-15 | iPad

```



```

2 | CC798      | 2023-10-17 | iPad
3 | CC864      | 2023-12-23 | iPad
4 | CC246      | 2024-01-10 | iPad
5 | CC806      | 2024-03-25 | iPad
6 | CC423      | 2024-02-12 | iPad
1 | CC135      | 2023-05-15 | airpods
2 | CC798      | 2023-10-17 | airpods
:

```

```

postgres=# SELECT nom_cli FROM clientes
postgres=# WHERE mes_cum_cli = 'marzo';
nom_cli

```

```
-----
```

```
(0 rows)
```

Nota: no hay clientes que cumplan en marzo.

```

postgres=# SELECT id_ped, val_ped, dir_cli, nom_cli
postgres=# FROM pedidos, clientes
postgres=# WHERE id_cli = id_cli_ped AND id_ped = 1;

```

```

id_ped | val_ped | dir_cli
| nom_cli

```

```
-----+-----+-----
```

```

1 | 3000000 | Calle Reforma 123, Int. 2, Colonia Juárez, C.P.
06600, Alcaldía Cuauhtémoc | Margarita
(1 row)

```

```

postgres=# SELECT nom_cli, fec_ped, val_ped
postgres=# FROM pedidos, clientes
postgres=# WHERE val_ped = (
postgres=# SELECT MAX(val_ped) FROM pedidos)
postgres=# AND id_cli = id_cli_ped;
nom_cli | fec_ped | val_ped

```

```
-----+-----+-----
```

```

Rosa | 2023-10-17 | 5000000
(1 row)

```

(otra manera de hacerlo)

```

postgres=# SELECT nom_cli, fec_ped, val_ped
postgres=# FROM pedidos, clientes
postgres=# WHERE id_cli = id_cli_ped AND val_ped IN(SELECT MAX(val_ped)
FROM pedidos);

```

```

postgres=# SELECT * FROM articulos
postgres=# ORDER BY aut_art;

```

```

id_art | tit_art | aut_art | edi_art | prec_art
-----+-----+-----+-----+-----
4 | airpods | apple | Pro 2 | 2999000
3 | iPad | Apple | Pro | 5000000
1 | computador | asus | X441U | 2300000
6 | tablet | huawei | MatePad | 830000

```

5	tablet	lenovo	M10	600000
2	celular	samsung	A31	700000

(6 rows)

```
postgres=# SELECT id_ped, val_ped, tit_art, can_art_artped
postgres=# FROM pedidos, articulos, articulosxpeditos
postgres=# WHERE id_art = id_art_artped AND id_ped = id_ped_artped;
```

id_ped	val_ped	tit_art	can_art_artped
1	3000000	computador	3
2	5000000	celular	4
3	4000000	iPad	1
4	2000000	airpods	2
5	2000000	tablet	3
6	2000000	tablet	1

(6 rows)

```
postgres=# SELECT * FROM clientes
postgres=# ORDER BY ape_cli DESC;
```

id_cli	nom_cli	ape_cli	dep_cli	mes_cum_cli
CC135	Margarita	Ramirez	Calle Reforma 123, Int. 2, Colonia Juárez, C.P. 06600, Alcaldía Cuauhtémoc	Valle Diciembre
CC246	Gabriela	Murillo	Calle Benito Juárez 234, Int. 3, Colonia Del Valle, C.P. 03100, Alcaldía Benito Juárez	Valle Enero
CC806	Vilma	Murgueitio	Calle Álvaro Obregón 678, Int. 8, Colonia Condesa, C.P. 06140, Alcaldía Cuauhtémoc	Valle Agosto
CC798	Rosa	Gonzalez	Calle Insurgentes 456, Int. 5, Colonia Roma Norte, C.P. 06700, Alcaldía Cuauhtémoc	Valle Enero
CC423	Sebastián	Gomez	Calle Reforma 123, Int. 2, Colonia Juárez, C.P. 06600, Alcaldía Cuauhtémoc	Valle Diciembre
CC864	Elena	Galeano	Calle Hidalgo 789, Int. 10, Colonia Centro, C.P. 06000, Alcaldía Cuauhtémoc	Valle Noviembre

(6 rows)

```
postgres=# SELECT * FROM articulos
postgres=# ORDER BY aut_art;
```

id_art	tit_art	aut_art	edi_art	prec_art
4	airpods	apple	Pro 2	2999000
3	iPad	Apple	Pro	5000000
1	computador	asus	X441U	2300000
6	tablet	huawei	MatePad	830000
5	tablet	lenovo	M10	600000
2	celular	samsung	A31	700000

(6 rows)

```
postgres=# SELECT nom_cli, dir_cli, id_ped, can_art_artped
postgres=# FROM pedidos, clientes, articulosxpeditos
postgres=# WHERE id_ped = id_ped_artped AND id_cli_ped = id_cli AND
id_art_artped = 2;
```

nom_cli	dir_cli
id_ped can_art_artped	
Rosa Calle Insurgentes 456, Int. 5, Colonia Roma Norte, C.P. 06700,	
Alcaldía Cuauhtémoc 2	4

(1 row)

```

postgres=# SELECT * FROM compañía
postgres=# WHERE comañofun > 2001 AND comañofun < 2008;
   comnit   | comnombre | comañofun | comreplegal
-----+-----+-----+-----
899999999-5 | compañía 1 | 2002 | legal compañía 1
000000002-1 | compañía 3 | 2003 | legal compañía 3
000000004-1 | compañía 5 | 2004 | legal compañía 5
(3 rows)

```

```

postgres=# SELECT autoplaca, automodelo, automarca, autopasajeros,
autocilindraje, asevalorasegurado, asecosto
postgres=# FROM aseguramientos, automotores, tiposautomotores
postgres=# WHERE autoplaca = aseplaca AND asefechaexpiracion <=
'20231031' AND asefechaexpiracion >= '20231001';

```

autoplaca automodelo automarca autopasajeros autocilindraje	asevalorasegurado asecosto
---	------------------------------

(0 rows)

Nota: no hay ningún registro correspondiente a esa fecha.

```

postgres=# SELECT inciplace, incifecha, incilugar, incicodigo,
asefechainicio, asecosto, aseestado, asevalorasegurado
postgres=# FROM aseguramientos, incidentes
postgres=# WHERE inciplace = aseplaca AND incifecha = '2022-09-30';

```

inciplace incifecha incilugar incicodigo asefechainicio	asecosto aseestado asevalorasegurado
---	--

(0 rows)

```

postgres=# SELECT incicantheridos, autoplaca, incilugar, incifecha,
asefechainicio, aseestado, asecosto, asevalorasegurado
postgres=# FROM incidentes, aseguramientos, automotores
postgres=# WHERE inciplace = aseplaca AND autoplaca = inciplace AND
incicantheridos = 1;

```

incicantheridos autoplaca	incilugar	incifecha	asefechainicio aseestado asecosto asevalorasegurado
-----------------------------	-----------	-----------	---

15	1	DEF456	Calle Falsa 123	2023-08-22	2022-07-
	Expirado	1100000	45000000		
01	1	PQR678	Avenida Las Palmas	2023-04-25	2023-09-
	Activo	1250000	48000000		

```
postgres=# SELECT * FROM aseguramientos
postgres=# WHERE asecosto IN(SELECT MAX(asecosto) FROM aseguramientos);
```

asecodigo	asefechainicio	asefechaexpiracion	asevalorasegurado	aseestado	asecosto	aseplaca
5	2023-05-20	2024-05-20	70000000	Activo	1400000	MNO345

(1 row)

```
postgres=# SELECT aseestado, asevalorasegurado
postgres=# FROM aseguramientos
postgres=# WHERE aseplaca IN (SELECT inci placa
postgres=# FROM incidentes
postgres=# WHERE incicanautosinvolucrados = (SELECT
postgres=# MIN(incicanautosinvolucrados)
postgres=# FROM incidentes));
```

aseestado	asevalorasegurado
Activo	60000000

(1 row)

```
postgres=# SELECT inci placa, incifecha, incilugar, incicantheridos,
postgres=# asefechainicio, asefechaexpiracion, asevalorasegurado
postgres=# FROM incidentes, aseguramientos
postgres=# WHERE inci placa = aseplaca AND inci placa = 'QGX767';
```

inci placa	incifecha	incilugar	incicantheridos
QGX767	2023-09-15	Avenida Siempre Viva	2
01-01	2024-01-01	50000000	

(1 row)

```
postgres=# SELECT * FROM compañía
postgres=# WHERE comnit = '899999999-5';
```

comnit	comnombre	comaño fun	comreplegal
899999999-5	compañía 1	2002	legal compañía 1

(1 row)

```
postgres=# SELECT * FROM aseguramientos, automotores
postgres=# WHERE aseplaca = autoplaca AND asevalorasegurado IN(SELECT
postgres=# MAX(asevalorasegurado) FROM aseguramientos);
```



```
postgres=# SELECT * FROM insertar_profesor('0987654321', 'Daniel',
'Guzmán', 1, 5000);
```

doc_prof	nom_prof	ape_prof	cate_prof	sal_prof
CC123	Julio	Castaña	2	3000000
CC456	Miguel	Mondragon	1	2000000
CC789	Yenifer	Orozco	1	4000000
CC098	Angel	Lopez	1	1000000
CC765	Nicolas	Moreno	1	500000
CC432	Marcela	Saavedra	2	350000
12345678901	Carlos	Gómez	1	3000
0987654321	Daniel	Guzmán	1	5000

(8 rows)

```
postgres=# CREATE OR REPLACE FUNCTION insertar_cursos(
postgres(# c_cod_curs INT,
postgres(# c_nom_cur VARCHAR,
postgres(# c_horas_cur INT,
postgres(# c_valor_cur INT
postgres(# )
postgres-# RETURNS TABLE(
postgres(# cod_curs INT,
postgres(# nom_curs VARCHAR,
postgres(# horas_cur INT,
postgres(# valor_cur INT
postgres(# ) AS $$
postgres$$ BEGIN
postgres$$ INSERT INTO cursos (cod_curs, nom_curs, horas_cur, valor_cur)
postgres$$ VALUES (c_cod_curs, c_nom_cur, c_horas_cur, c_valor_cur);
postgres$$ RETURN QUERY SELECT cursos.cod_curs, cursos.nom_curs,
cursos.horas_cur, cursos.valor_cur
postgres$$ FROM cursos;
postgres$$ END;
postgres$$ $$ LANGUAGE plpgsql;
CREATE FUNCTION
```

```
postgres=# SELECT * FROM insertar_cursos(8, 'ciencias', 50, 100000);
```

cod_curs	nom_curs	horas_cur	valor_cur
2	Introducción a la Programación	60	2000000
3	Emprendimiento	40	2000000
4	Seguridad y Salud en el Trabajo	40	500000
5	Introducción a los Lenguajes de Programación	20	250000
6	Lógica de Programación	20	100000
1	Gestión Bases de Datos	80	1500000

```

      8 | ciencias | 50 |
100000
(7 rows)

```

```

postgres=# SELECT * FROM insertar_cursos(7, 'JS', 50, 100000);

```

cod_curs	nom_curs	horas_cur	valor_cur
2	Introducción a la Programación	60	2000000
3	Emprendimiento	40	2000000
4	Seguridad y Salud en el Trabajo	40	500000
5	Introducción a los Lenguajes de Programación	20	250000
6	Lógica de Programación	20	100000
1	Gestión Bases de Datos	80	1500000
8	ciencias	50	100000
7	JS	50	100000

```

(8 rows)

```

```

postgres=# CREATE OR REPLACE FUNCTION insertar_estudiantes(
postgres(# e_doc_est VARCHAR,
postgres(# e_nom_est VARCHAR,
postgres(# e_ape_est VARCHAR,
postgres(# e_edad_est INT
postgres(# )
postgres-# RETURNS TABLE(
postgres(# doc_est VARCHAR,
postgres(# nom_est VARCHAR,
postgres(# ape_est VARCHAR,
postgres(# edad_est INT
postgres(# ) AS $$
postgres$# BEGIN
postgres$# INSERT INTO estudiantes (doc_est, nom_est, ape_est, edad_est)
postgres$# VALUES (e_doc_est, e_nom_est, e_ape_est, e_edad_est);
postgres$# RETURN QUERY SELECT estudiantes.doc_est, estudiantes.nom_est,
postgres$# estudiantes.ape_est, estudiantes.edad_est
postgres$# FROM estudiantes;
postgres$# END;
postgres$# $$ LANGUAGE plpgsql;
CREATE FUNCTION

```

```

postgres=# SELECT
postgres-# * FROM insertar_estudiantes ('123X', 'Martin', 'Tabarez', 15);
doc_est | nom_est | ape_est | edad_est
-----+-----+-----+-----

```

CC123		Laura		Barona Saavedra		25
CC456		Mariana		Gomez		28
CC789		Sara		Lopez		27
CC098		Jose		Daza		24
CC765		Jhon		Arias		25
CC432		Jason		Martinez		30
123X		Martin		Tabarez		15

(7 rows)

```

postgres=# CREATE OR REPLACE FUNCTION insertar_estudiantesxcursos(
postgres(# ec_cod_cur_estcur INT,
postgres(# ec_doc_est_estcur VARCHAR,
postgres(# ec_fec_ini_estcur DATE
postgres(# )
postgres-# RETURNS TABLE(
postgres(# cod_cur_estcur INT,
postgres(# doc_est_estcur VARCHAR,
postgres(# fec_ini_estcur DATE
postgres(# ) AS $$
postgres$$ BEGIN
postgres$$ INSERT INTO estudiantesxcursos (cod_cur_estcur,
doc_est_estcur, fec_ini_estcur)
postgres$$ VALUES (ec_cod_cur_estcur, ec_doc_est_estcur,
ec_fec_ini_estcur);
postgres$$ RETURN QUERY SELECT estudiantesxcursos.cod_cur_estcur,
estudiantesxcursos.doc_est_estcur, estudiantesxcursos.fec_ini_estcur
postgres$$ FROM estudiantesxcursos;
postgres$$ END;
postgres$$ $$ LANGUAGE plpgsql;
CREATE FUNCTION

```

```

postgres=# SELECT * FROM insertar_estudiantesxcursos (1, 'CC123', '20224-08-12');

```

cod_cur_estcur		doc_est_estcur		fec_ini_estcur
-----+-----+-----				
1		CC123		2024-08-02
2		CC456		2024-08-06
3		CC789		2024-08-06
4		CC098		2024-08-10
5		CC765		2024-09-04
6		CC432		2024-03-25
1		CC123		20224-08-12

(7 rows)

```

postgres=# SELECT * FROM insertar_estudiantesxcursos (2, 'CC456', '2024-08-12');

```

cod_cur_estcur		doc_est_estcur		fec_ini_estcur
-----+-----+-----				
1		CC123		2024-08-02
2		CC456		2024-08-06
3		CC789		2024-08-06
4		CC098		2024-08-10
5		CC765		2024-09-04
6		CC432		2024-03-25

1	CC123	20224-08-12
2	CC456	2024-08-12

(8 rows)

```
postgres=# CREATE OR REPLACE FUNCTION mostrar_salarios_por_categoria()
postgres=# RETURNS TABLE(
postgres=# doc_prof VARCHAR,
postgres=# nom_prof VARCHAR,
postgres=# ape_prof VARCHAR,
postgres=# cate_prof INT,
postgres=# sal_prof INT
postgres=# ) AS $$
postgres$# BEGIN
postgres$# RETURN QUERY
postgres$# SELECT profesores.doc_prof, profesores.nom_prof,
profesores.ape_prof, profesores.cate_prof, profesores.sal_prof
postgres$# FROM profesores
postgres$# ORDER BY profesores.cate_prof;
postgres$# END;
postgres$# $$ LANGUAGE plpgsql;
CREATE FUNCTION
```

```
postgres=# SELECT * FROM mostrar_salarios_por_categoria();
 doc_prof | nom_prof | ape_prof | cate_prof | sal_prof
-----+-----+-----+-----+-----
 0987654321 | Daniel | Guzmán | 1 | 5000
 CC456 | Miguel | Mondragon | 1 | 2000000
 CC789 | Yenifer | Orozco | 1 | 4000000
 CC098 | Angel | Lopez | 1 | 1000000
 CC765 | Nicolas | Moreno | 1 | 500000
 12345678901 | Carlos | Gómez | 1 | 3000
 CC123 | Julio | Castaño | 2 | 3000000
 CC432 | Marcela | Saavedra | 2 | 350000
```

(8 rows)

```
postgres=# CREATE OR REPLACE FUNCTION mostrar_cursos_valor_mayor_500k()
postgres=# RETURNS TABLE(
postgres=# cod_curs INT,
postgres=# nom_curs VARCHAR,
postgres=# horas_cur INT,
postgres=# valor_cur INT
postgres=# ) AS $$
postgres$# BEGIN
postgres$# RETURN QUERY
postgres$# SELECT cursos.cod_curs, cursos.nom_curs, cursos.horas_cur,
cursos.valor_cur
postgres$# FROM cursos
postgres$# WHERE cursos.valor_cur > 500000;
postgres$# END;
postgres$# $$ LANGUAGE plpgsql;
CREATE FUNCTION
```

```
postgres=# SELECT * FROM mostrar_cursos_valor_mayor_500k();
 cod_curs | nom_curs | horas_cur | valor_cur
```

```

-----+-----+-----+-----
      2 | Introducción a la Programación |      60 | 2000000
      3 | Emprendimiento                    |      40 | 2000000
      1 | Gestión Bases de Datos            |      80 | 1500000
(3 rows)

```

```

postgres=# CREATE OR REPLACE FUNCTION
mostrar_cursos_orden_ascendente_por_valor()
postgres=# RETURNS TABLE(
postgres=#   cod_curs INT,
postgres=#   nom_curs VARCHAR,
postgres=#   horas_cur INT,
postgres=#   valor_cur INT
postgres=# ) AS $$
postgres$$ BEGIN
postgres$$ RETURN QUERY
postgres$$ SELECT cursos.cod_curs, cursos.nom_curs, cursos.horas_cur,
cursos.valor_cur
postgres$$ FROM cursos
postgres$$ ORDER BY cursos.valor_cur ASC;
postgres$$ END;
postgres$$ $$ LANGUAGE plpgsql;
CREATE FUNCTION

```

```

postgres=# SELECT * FROM mostrar_cursos_orden_ascendente_por_valor();

```

```

   cod_curs |          nom_curs          | horas_cur |
-----+-----+-----+-----
-----+-----+-----+-----
      7 | JS                          |      50 |
100000
      6 | Lógica de Programación     |      20 |
100000
      8 | ciencias                    |      50 |
100000
      5 | Introducción a los Lenguajes de Programación |      20 |
250000
      4 | Seguridad y Salud en el Trabajo |      40 |
500000
      1 | Gestión Bases de Datos      |      80 |
1500000
      2 | Introducción a la Programación |      60 |
2000000
      3 | Emprendimiento              |      40 |
2000000
(8 rows)

```

