===================================================================================================================================

PRACTICA 4 Procedimientos Almacenados

MOISES ZABALETA CRUZ 1033810224

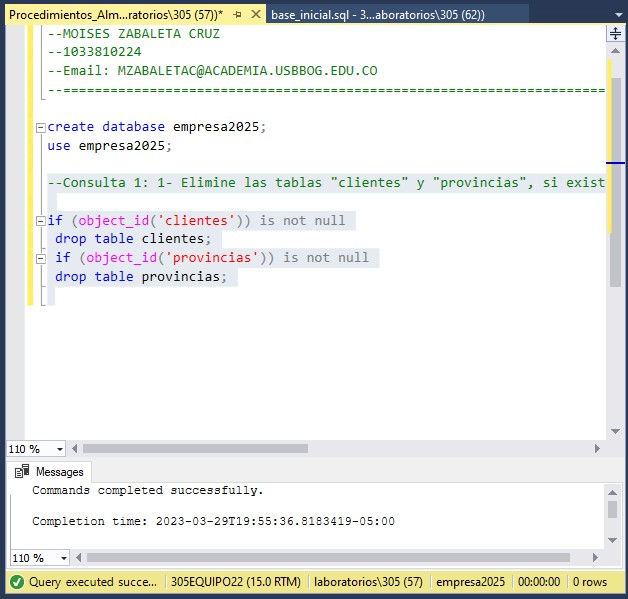
Email: [MZABALETAC@ACADEMIA.USBBOG.EDU.CO](mailto:MZABALETAC@ACADEMIA.USBBOG.EDU.CO)

================================================================================================================================

--Consulta 1: 1- Elimine las tablas "clientes" y "provincias", si existen:

if (object\_id('clientes')) is not null drop table clientes;

if (object\_id('provincias')) is not null drop table provincias;



--Consulta 2: 2- Créelas con las siguientes estructuras:

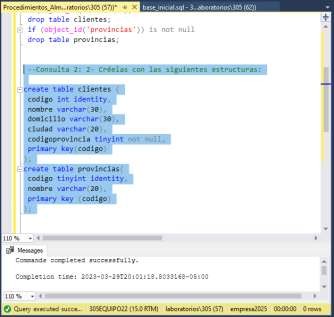
--Consulta 2: 2- Créelas con las siguientes estructuras: create table clientes (

codigo int identity,

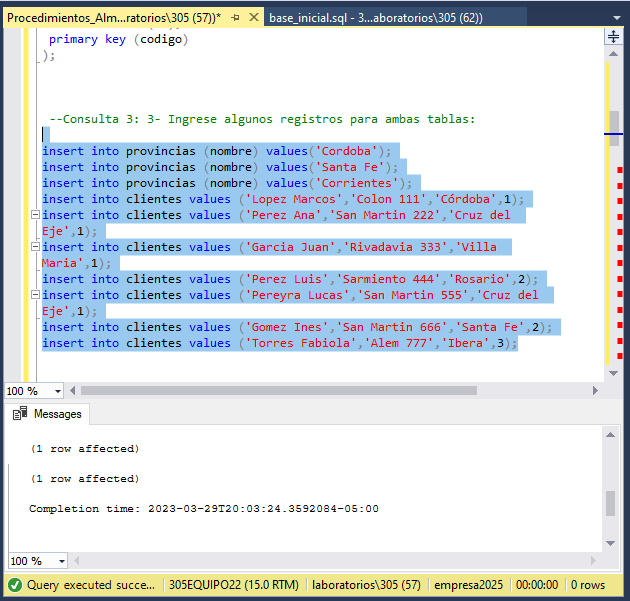
nombre varchar(30), domicilio varchar(30), ciudad varchar(20),

codigoprovincia tinyint not null, primary key(codigo)

);

create table provincias( codigo tinyint identity, nombre varchar(20), primary key (codigo)

);

--Consulta 3: 3- Ingrese algunos registros para ambas tablas:

insert into provincias (nombre) values('Cordoba'); insert into provincias (nombre) values('Santa Fe'); insert into provincias (nombre) values('Corrientes'); insert into clientes values ('Lopez Marcos','Colon 111','Córdoba',1);

insert into clientes values ('Perez Ana','San Martin 222','Cruz del

Eje',1);

insert into clientes values ('Garcia Juan','Rivadavia 333','Villa

Maria',1);

insert into clientes values ('Perez Luis','Sarmiento 444','Rosario',2);

insert into clientes values ('Pereyra Lucas','San Martin 555','Cruz del

Eje',1);

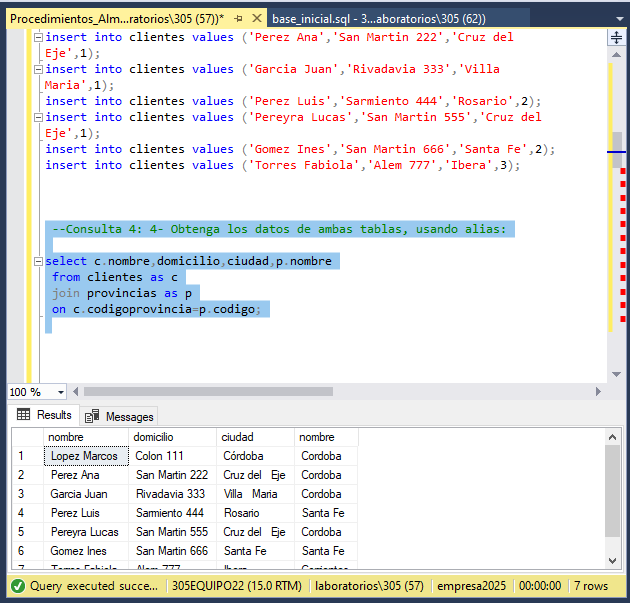
insert into clientes values ('Gomez Ines','San Martin 666','Santa Fe',2);

insert into clientes values ('Torres Fabiola','Alem 777','Ibera',3);

--Consulta 4: 4- Obtenga los datos de ambas tablas, usando alias:

select c.nombre,domicilio,ciudad,p.nombre from clientes as c

join provincias as p

on c.codigoprovincia=p.codigo;

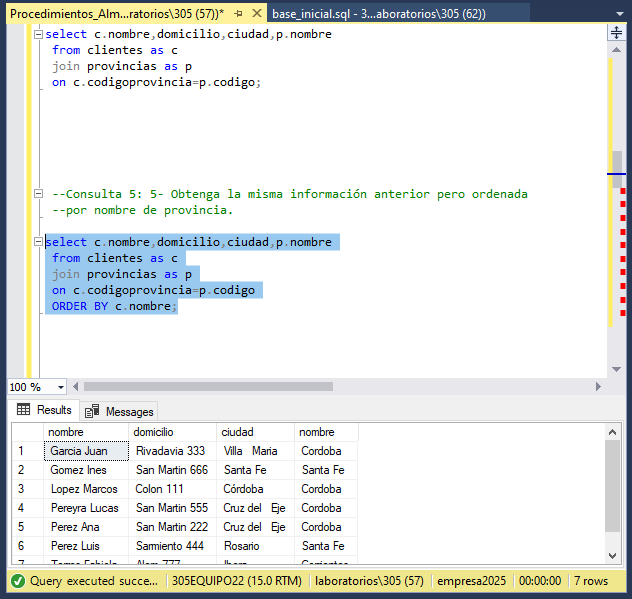
--Consulta 5: 5- Obtenga la misma información anterior pero ordenada

--por nombre de provincia.

select c.nombre,domicilio,ciudad,p.nombre from clientes as c

join provincias as p

on c.codigoprovincia=p.codigo ORDER BY c.nombre;

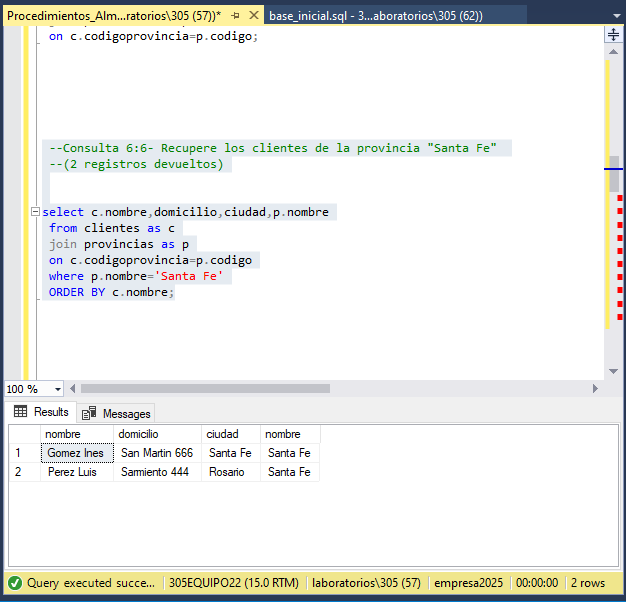


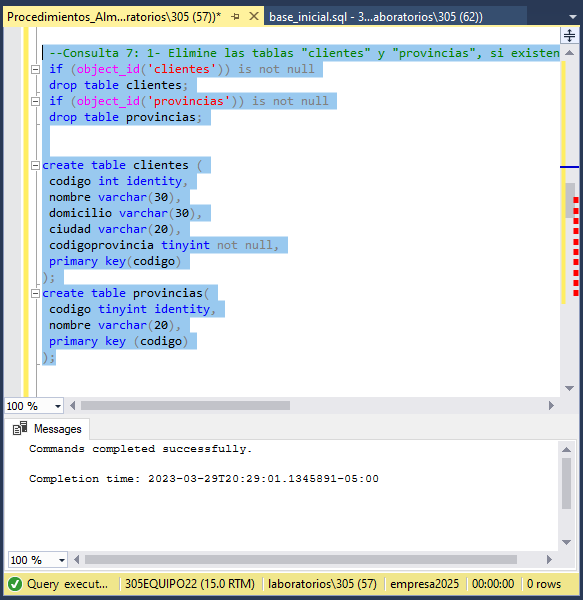
--Consulta 6:6- Recupere los clientes de la provincia "Santa Fe" (2 registros devueltos)

select c.nombre,domicilio,ciudad,p.nombre from clientes as c

join provincias as p

on c.codigoprovincia=p.codigo where p.nombre='Santa Fe' ORDER BY c.nombre;



--Consulta 7: 1- Elimine las tablas "clientes" y "provincias", si existen y cree las

if (object\_id('clientes')) is not null drop table clientes;

if (object\_id('provincias')) is not null drop table provincias;

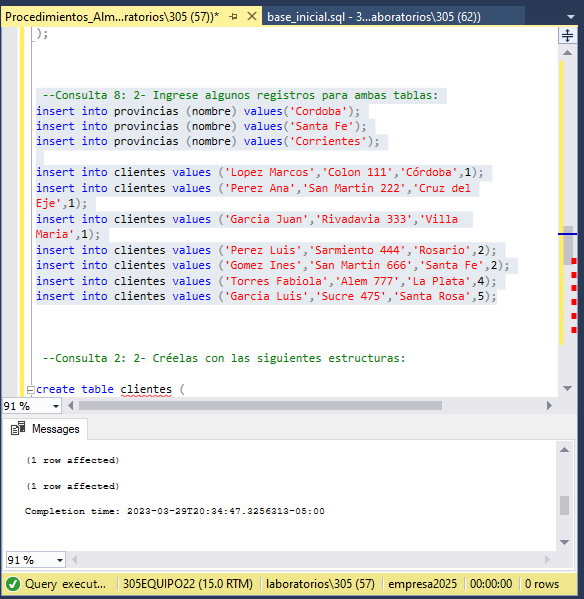
create table clientes ( codigo int identity, nombre varchar(30), domicilio varchar(30), ciudad varchar(20),

codigoprovincia tinyint not null, primary key(codigo)

);

create table provincias( codigo tinyint identity, nombre varchar(20), primary key (codigo)

);

--Consulta 8: 2- Ingrese algunos registros para ambas tablas:

insert into provincias (nombre) values('Cordoba'); insert into provincias (nombre) values('Santa Fe'); insert into provincias (nombre) values('Corrientes');

insert into clientes values ('Lopez Marcos','Colon 111','Córdoba',1);

insert into clientes values ('Perez Ana','San Martin 222','Cruz del

Eje',1);

insert into clientes values ('Garcia Juan','Rivadavia 333','Villa

Maria',1);

insert into clientes values ('Perez Luis','Sarmiento 444','Rosario',2);

insert into clientes values ('Gomez Ines','San Martin 666','Santa Fe',2);

insert into clientes values ('Torres Fabiola','Alem 777','La Plata',4);

insert into clientes values ('Garcia Luis','Sucre 475','Santa Rosa',5);

--Consulta 9: 3- Muestre todos los datos de los clientes, incluido el nombre de la

provincia:

select c.nombre,domicilio,ciudad, p.nombre from clientes as c

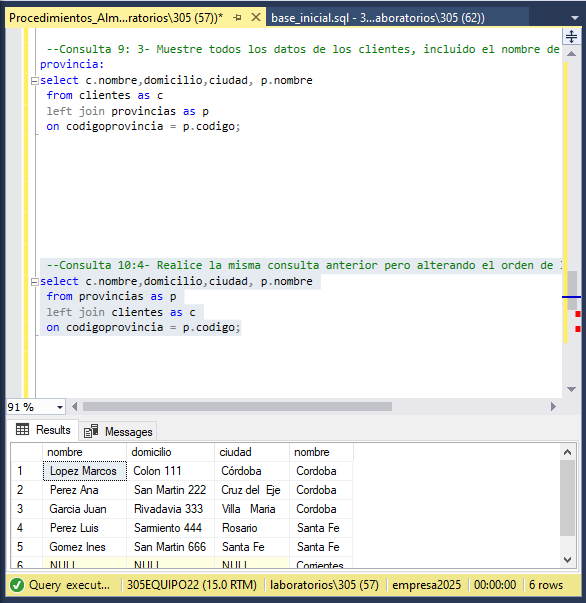
left join provincias as p

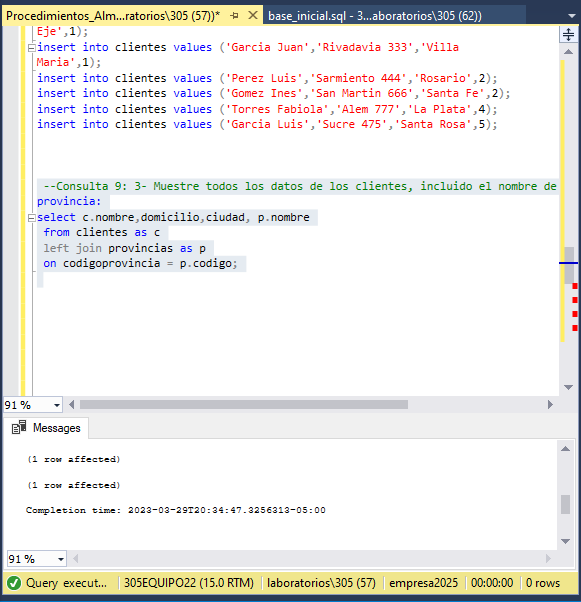
on codigoprovincia = p.codigo;

--Consulta 10:4- Realice la misma consulta anterior pero alterando el orden de las tablas:

select c.nombre,domicilio,ciudad, p.nombre from provincias as p

left join clientes as c

on codigoprovincia = p.codigo;

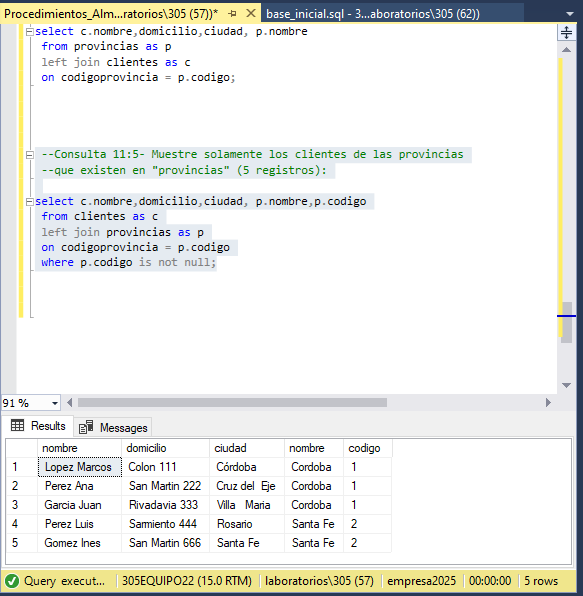


--Consulta 11:5- Muestre solamente los clientes de las provincias

--que existen en "provincias" (5 registros):

select c.nombre,domicilio,ciudad, p.nombre,p.codigo from clientes as c

left join provincias as p

on codigoprovincia = p.codigo where p.codigo is not null;

--Consulta 12:6- Muestre todos los clientes cuyo código de provincia

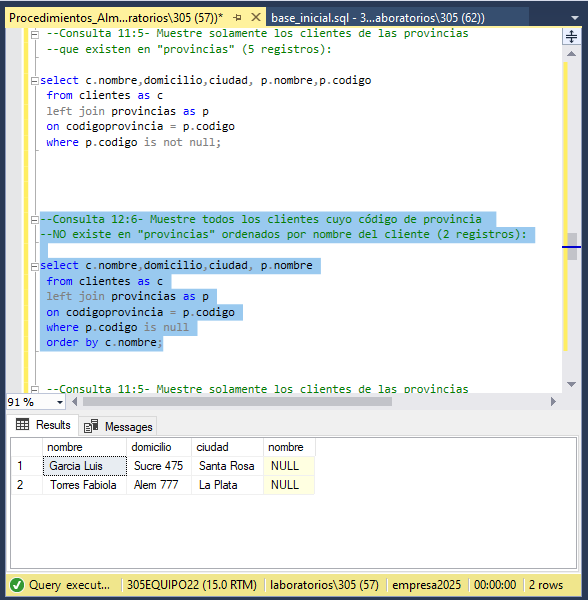
--NO existe en "provincias" ordenados por nombre del cliente (2 registros):

select c.nombre,domicilio,ciudad, p.nombre from clientes as c

left join provincias as p

on codigoprovincia = p.codigo where p.codigo is null

order by c.nombre;

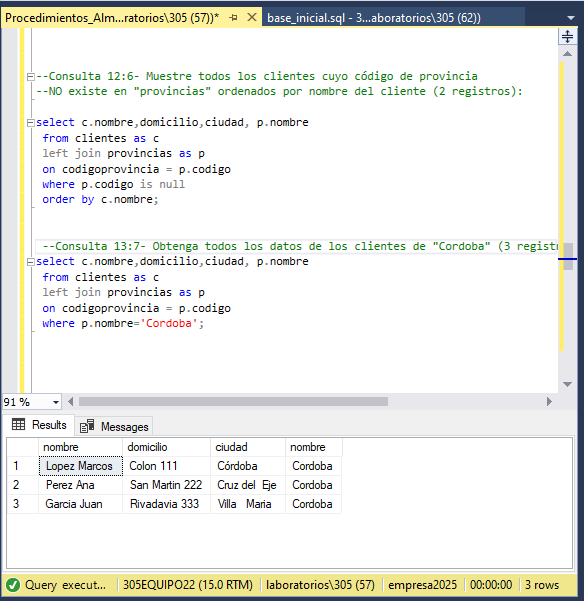


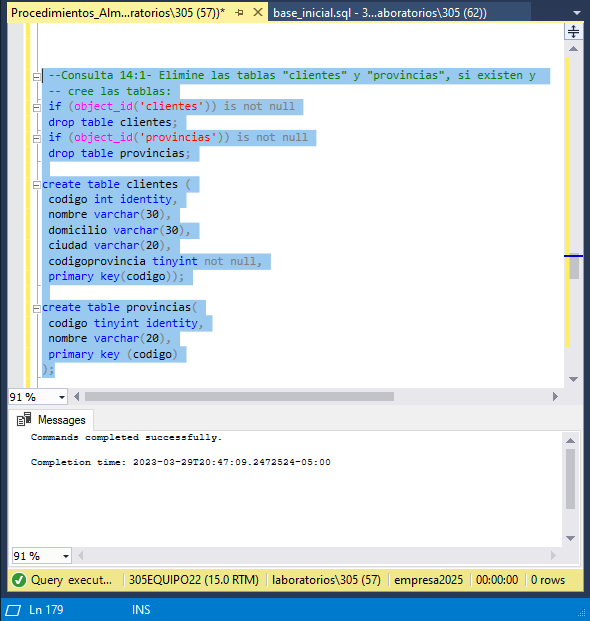
--Consulta 13:7- Obtenga todos los datos de los clientes de "Cordoba" (3 registros):

select c.nombre,domicilio,ciudad, p.nombre from clientes as c

left join provincias as p

on codigoprovincia = p.codigo where p.nombre='Cordoba';



--Consulta 14:1- Elimine las tablas "clientes" y "provincias", si existen y

-- cree las tablas:

if (object\_id('clientes')) is not null drop table clientes;

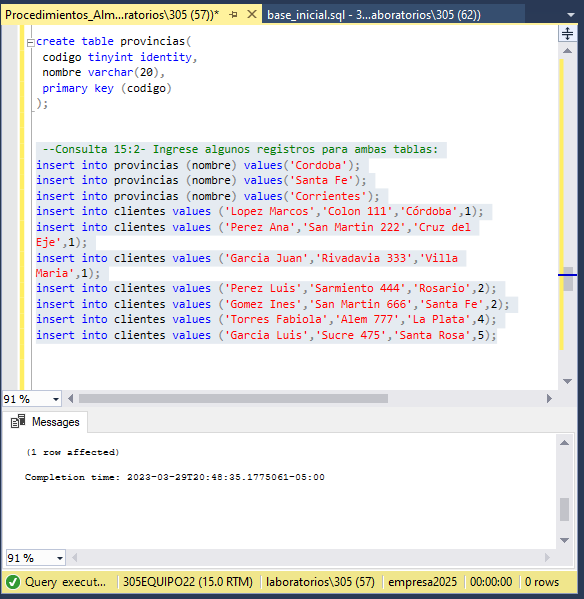
if (object\_id('provincias')) is not null drop table provincias;

create table clientes ( codigo int identity, nombre varchar(30), domicilio varchar(30), ciudad varchar(20),

codigoprovincia tinyint not null, primary key(codigo));

create table provincias( codigo tinyint identity, nombre varchar(20), primary key (codigo)

);

--Consulta 15:2- Ingrese algunos registros para ambas tablas:

insert into provincias (nombre) values('Cordoba'); insert into provincias (nombre) values('Santa Fe'); insert into provincias (nombre) values('Corrientes'); insert into clientes values ('Lopez Marcos','Colon 111','Córdoba',1);

insert into clientes values ('Perez Ana','San Martin 222','Cruz del

Eje',1);

insert into clientes values ('Garcia Juan','Rivadavia 333','Villa

Maria',1);

insert into clientes values ('Perez Luis','Sarmiento 444','Rosario',2);

insert into clientes values ('Gomez Ines','San Martin 666','Santa Fe',2);

insert into clientes values ('Torres Fabiola','Alem 777','La Plata',4);

insert into clientes values ('Garcia Luis','Sucre 475','Santa Rosa',5);

--Consulta 16:3- Muestre todos los datos de los clientes, incluido

--el nombre de la provincia empleando un "right join".

select c.nombre,c.codigo,c.domicilio,c.ciudad,c.codigoprovincia,

p.nombre,p.codigo from clientes as c

right join provincias as p on codigoprovincia = p.codigo

--Consulta 17:4- Obtenga la misma salida que la consulta anterior

--pero empleando un "left join".

select c.nombre,c.codigo,c.domicilio,c.ciudad,c.codigoprovincia,

p.nombre,p.codigo from clientes as c

left join provincias as p

on codigoprovincia = p.codigo

# 

--Consulta 18:5- Empleando un "right join", muestre solamente

--los clientes de las provincias que existen en"provincias" (5 registros)

select c.nombre,domicilio,ciudad, p.nombre,p.codigo from clientes as c

right join provincias as p on codigoprovincia = p.codigo where p.codigo is not null;

--Consulta 19:6- Muestre todos los clientes cuyo código de provincia

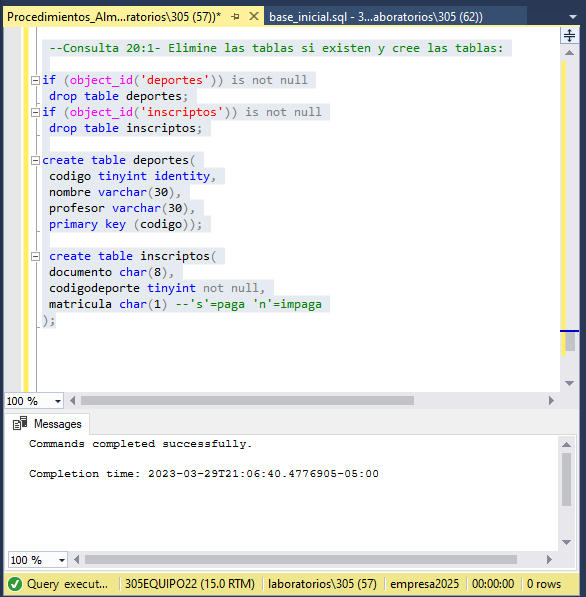
--NO existe en "provincias" ordenados por ciudad (2 registros)

select c.nombre,domicilio,ciudad, p.nombre,p.codigo from clientes as c

left join provincias as p

on codigoprovincia = p.codigo where p.codigo is not null;

# 

--Consulta 20:1- Elimine las tablas si existen y cree las tablas:

if (object\_id('deportes')) is not null drop table deportes;

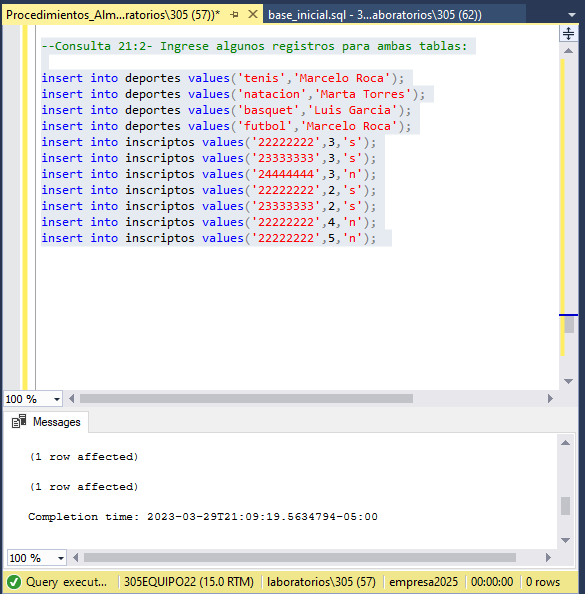
if (object\_id('inscriptos')) is not null drop table inscriptos;

create table deportes( codigo tinyint identity, nombre varchar(30), profesor varchar(30), primary key (codigo));

create table inscriptos( documento char(8), codigodeporte tinyint not null,

matricula char(1) --'s'=paga 'n'=impaga

);

--Consulta 21:2- Ingrese algunos registros para ambas tablas:

insert into deportes values('tenis','Marcelo Roca'); insert into deportes values('natacion','Marta Torres'); insert into deportes values('basquet','Luis Garcia'); insert into deportes values('futbol','Marcelo Roca'); insert into inscriptos values('22222222',3,'s');

insert into inscriptos values('23333333',3,'s'); insert into inscriptos values('24444444',3,'n'); insert into inscriptos values('22222222',2,'s'); insert into inscriptos values('23333333',2,'s'); insert into inscriptos values('22222222',4,'n'); insert into inscriptos values('22222222',5,'n');

--Consulta 22:3- Muestre todos la información de la tabla "inscriptos",

--y consulte la tabla "deportes" para obtener el nombre de cada deporte (6 registros)

select d.codigo,d.nombre,d.profesor,i.documento,i.codigodeporte,

i.matricula from deportes as d

right join inscriptos as i

on d.codigo = i.codigodeporte;

--Consulta 22:4- Empleando un "left join" con "deportes" obtenga todos los datos de los

--inscriptos (7 registros)

select d.codigo,d.nombre,d.profesor,i.documento,i.codigodeporte,

1. matricula from deportes as d

left join inscriptos as i

on d.codigo = i.codigodeporte;

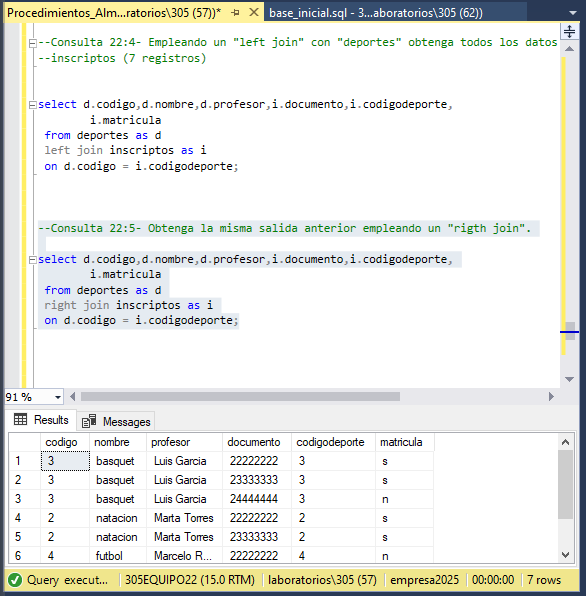
# 

--Consulta 22:5- Obtenga la misma salida anterior empleando un "rigth join".

select d.codigo,d.nombre,d.profesor,i.documento,i.codigodeporte,

* 1. matricula from deportes as d

right join inscriptos as i

on d.codigo = i.codigodeporte;

--Consulta 22:7- Muestre los deportes para los cuales no hay inscriptos, empleando un

--"left join" (1 registro)

select d.codigo,d.nombre,d.profesor,i.documento,i.codigodeporte,

1. matricula from deportes as d

left join inscriptos as i

on d.codigo = i.codigodeporte where i.codigodeporte is null;

