FINAL DRILLING: ARRIENDO DE DVDS

Javier Medina Muñoz

- 1. Construye las siguientes consultas:
 - Aquellas usadas para insertar, modificar y eliminar un Customer, Staff y Actor.

Customer:

Insertar:

INSERT INTO customer(store_id,first_name, last_name, email, address_id, activebool, create_date, last_update, active) VALUES (1,'Jon','Snow','jsnow@gmail.com',5,TRUE,'2010-11-04', CURRENT_DATE, 1);



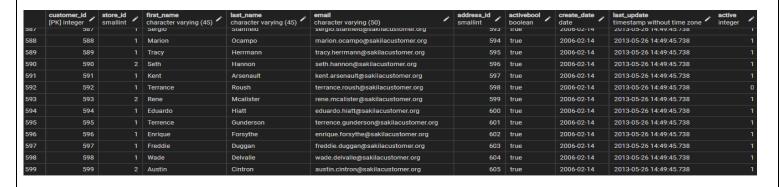
Modificar:

UPDATE customer SET email = 'esto_es_una_prueba@gmail.com' WHERE customer_id = 20;

	customer_id [PK] integer	store_id smallint	first_name character varying (45)	last_name character varying (45)	email character varying (50)	address_id smallint	activebool boolean	create_date /		active nteger	1
17	17		Donna	Thompson	donna.thompson@sakilacustomer.org	21	true	2006-02-14	2013-05-26 14:49:45.738		1
18	18		Carol	Garcia	carol.garcia@sakilacustomer.org	22	true	2006-02-14	2013-05-26 14:49:45.738		1
19	19		Ruth	Martinez	ruth.martinez@sakilacustomer.org	23	true	2006-02-14	2013-05-26 14:49:45.738		1
20	20	2	Sharon	Robinson	esto_es_una_prueba@gmail.com	24	true	2006-02-14	2024-10-16 16:20:45.414019		1
21	21		Michelle	Clark	michelle.clark@sakilacustomer.org	25	true	2006-02-14	2013-05-26 14:49:45.738		1
22	22		Laura	Rodriguez	laura.rodriguez@sakilacustomer.org	26	true	2006-02-14	2013-05-26 14:49:45.738		1

Eliminar:

DELETE FROM customer WHERE first_name = 'Jon' AND last_name = 'Snow';



Staff:

Insertar:

INSERT INTO staff(first_name, last_name, address_id, email, store_id, active, username, password, last_update) VALUES ('Rob','Stark',504,'rstark@gmail.com',2,TRUE,'Rob','asdfghjklñlkj623482842392342', '2022-05-29');



Modificar:

UPDATE staff SET first_name = 'Ana', username = 'Ana' WHERE first_name = 'Mike';



Eliminar:

DELETE FROM staff WHERE staff_id = 3;



Actor:

Insertar:

INSERT INTO actor(first_name, last_name, last_update) VALUES ('Robert', 'Downey', '2023-10-17');

	actor_id [PK] integer	first_name character varying (45)	last_name character varying (45)	last_update timestamp without time zone
196	196	Bela	Walken	2013-05-26 14:47:57.62
197	197	Reese	West	2013-05-26 14:47:57.62
198	198	Mary	Keitel	2013-05-26 14:47:57.62
199	199	Julia	Fawcett	2013-05-26 14:47:57.62
200	200	Thora	Temple	2013-05-26 14:47:57.62
201	201	Robert	Downey	2023-10-17 00:00:00

Modificar:

UPDATE actor SET first_name = 'Javier', last_name = 'Medina' WHERE actor_id = 100;

	actor_id [PK] integer	first_name character varying (45)	last_name character varying (45)	last_update timestamp without time zone
97	97	Meg	Hawke	2013-05-26 14:47:57.62
98	98	Chris	Bridges	2013-05-26 14:47:57.62
99	99	Jim	Mostel	2013-05-26 14:47:57.62
100	100	Javier	Medina	2024-10-16 16:35:34.771232
101	101	Susan	Davis	2013-05-26 14:47:57.62
102	102	Walter	Torn	2013-05-26 14:47:57.62
103	103	Matthew	Leigh	2013-05-26 14:47:57.62
104	104	Penelope	Cronyn	2013-05-26 14:47:57.62

Eliminar:

DELETE FROM actor WHERE first_name = 'Robert' AND last_name = 'Downey';

	actor_id [PK] integer	first_name character varying (45)	last_name character varying (45)	last_update timestamp without time zone
เชช	188	коск	Dukakis	ZU13-U0-Z0 14.47.07.0Z
189	189	Cuba	Birch	2013-05-26 14:47:57.62
190	190	Audrey	Bailey	2013-05-26 14:47:57.62
191	191	Gregory	Gooding	2013-05-26 14:47:57.62
192	192	John	Suvari	2013-05-26 14:47:57.62
193	193	Burt	Temple	2013-05-26 14:47:57.62
194	194	Meryl	Allen	2013-05-26 14:47:57.62
195	195	Jayne	Silverstone	2013-05-26 14:47:57.62
196	196	Bela	Walken	2013-05-26 14:47:57.62
197	197	Reese	West	2013-05-26 14:47:57.62
198	198	Mary	Keitel	2013-05-26 14:47:57.62
199	199	Julia	Fawcett	2013-05-26 14:47:57.62
200	200	Javier	Medina	2024-10-16 16:34:29.781148

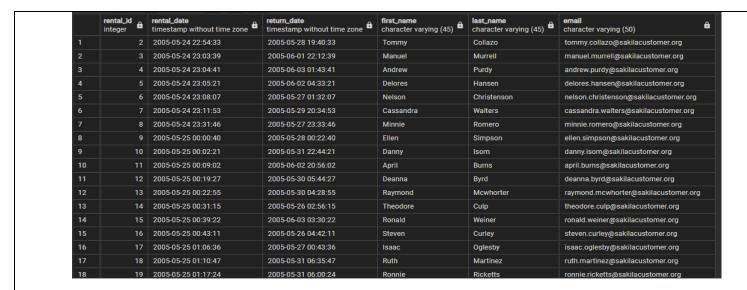
b. Listar todas las "rental" con los datos del "customer" dado un año y mes.

SELECT r.rental_id, r.rental_date, r.return_date, c.first_name, c.last_name, c.email FROM rental r

JOIN customer c ON r.customer_id = c.customer_id

WHERE EXTRACT(YEAR FROM r.rental_date) = 2005

AND EXTRACT(MONTH FROM r.rental_date) = 05;



Para datos completes revisar "rental.csv"

 c. Listar Número, Fecha (payment_date) y Total (amount) de todas las "payment".

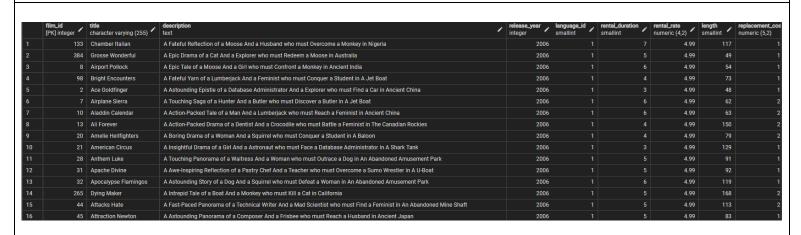
SELECT payment_id, payment_date, amount FROM payment;

	payment_id [PK] integer	payment_date timestamp without time zone	amount numeric (5,2)
1	17503	2007-02-15 22:25:46.996577	7.99
2	17504	2007-02-16 17:23:14.996577	1.99
3	17505	2007-02-16 22:41:45.996577	7.99
4	17506	2007-02-19 19:39:56.996577	2.99
5	17507	2007-02-20 17:31:48.996577	7.99
6	17508	2007-02-21 12:33:49.996577	5.99
7	17509	2007-02-17 23:58:17.996577	5.99
8	17510	2007-02-20 02:11:44.996577	5.99
9	17511	2007-02-20 13:57:39.996577	2.99
10	17512	2007-02-16 00:10:50.996577	4.99
11	17513	2007-02-16 01:15:33.996577	6.99
12	17514	2007-02-17 01:26:00.996577	0.99
13	17515	2007-02-17 04:32:51.996577	0.99
14	17516	2007-02-18 18:26:38.996577	6.99
15	17517	2007-02-20 07:03:29.996577	8.99
16	17518	2007-02-21 14:42:28.996577	0.99
17	17519	2007-02-15 10:54:44.996577	3.99
18	17520	2007-02-15 19:36:27.996577	4.99

Para datos completes revisar "payment.csv"

d. Listar todas las "film" del año 2006 que contengan un (rental_rate) mayor a 4.0.

SELECT *
FROM film
WHERE release_year = 2006
AND rental_rate > 4.0;



Para datos completes revisar "film.csv"

2. Realiza un Diccionario de datos que contenga el nombre de las tablas y columnas, si éstas pueden ser nulas, y su tipo de dato correspondiente

SELECT

table_name AS "Tabla", column_name AS "Columna", data_type AS "Tipo de Dato", is_nullable AS "¿Puede ser nulo?"

FROM

information_schema.columns

WHERE

table_schema = 'public'

AND table_name IN('actor','store','address','category','city','country','customer', 'film_actor','film_category','inventory','language','rental','staff','payment','film')

ORDER BY table_name ASC;

	Tabla name	Columna name	Tipo de Dato character varying	¿Puede ser nulo? character varying (3)
9	address	address	character varying	NO
10	address	city_id	smallint	NO
11	address	address_id	integer	NO
12	address	phone	character varying	NO
13	category	name	character varying	NO
14	category	last_update	timestamp without time zone	NO
15	category	category_id	integer	NO
16	city	city_id	integer	NO
17	city	city	character varying	NO
18	city	last_update	timestamp without time zone	NO
19	city	country_id	smallint	NO
20	country	last_update	timestamp without time zone	NO
21	country	country	character varying	NO
22	country	country_id	integer	NO
23	customer	last_name	character varying	NO
24	customer	customer_id	integer	NO
25	customer	store_id	smallint	NO
26	customer	address_id	smallint	NO

Para datos completes revisar "diccionario.csv"