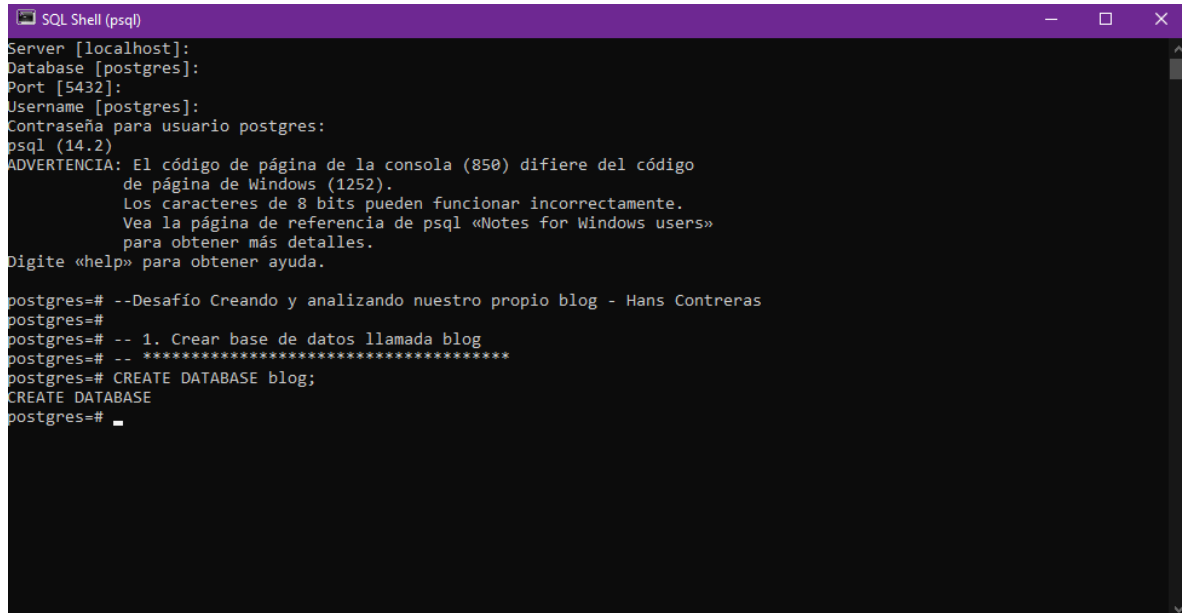


Desafío Creando y analizando nuestro propio blog - Hans Contreras Robledo

Requerimientos:

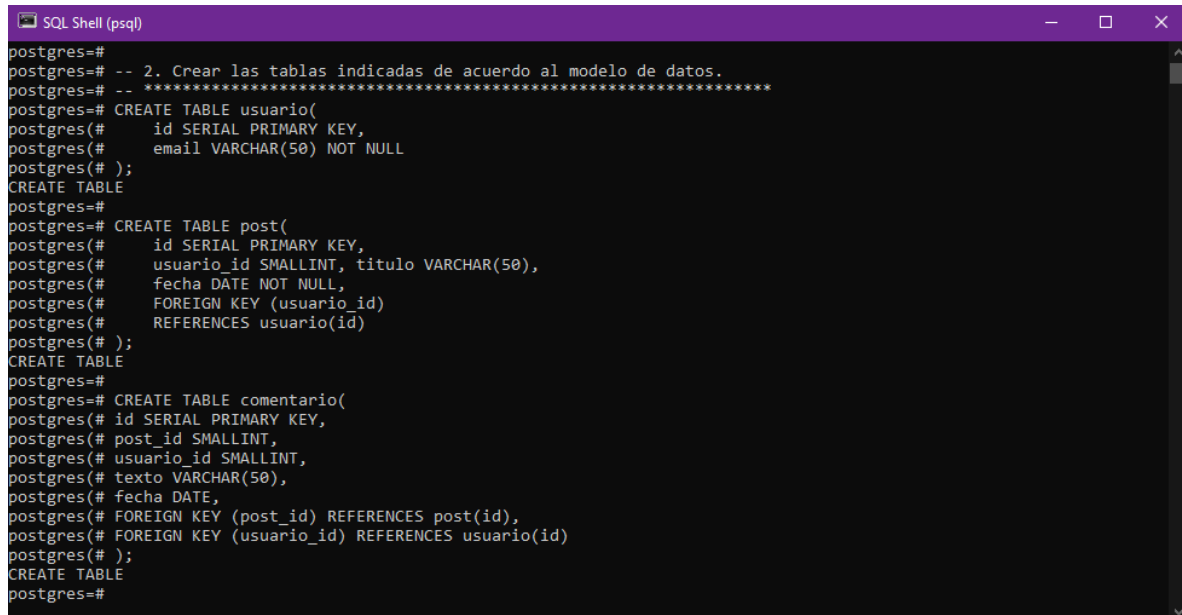
1. Crear base de datos llamada blog.



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Contraseña para usuario postgres:
psql (14.2)
ADVERTENCIA: El código de página de la consola (850) difiere del código
de página de Windows (1252).
Los caracteres de 8 bits pueden funcionar incorrectamente.
Vea la página de referencia de psql «Notes for Windows users»
para obtener más detalles.
Digite «help» para obtener ayuda.

postgres=# --Desafío Creando y analizando nuestro propio blog - Hans Contreras
postgres=#
postgres=# -- 1. Crear base de datos llamada blog
postgres=# -- *****
postgres=# CREATE DATABASE blog;
CREATE DATABASE
postgres=#
```

2. Crear las tablas indicadas de acuerdo al modelo de datos.



```
SQL Shell (psql)
postgres=#
postgres=# -- 2. Crear las tablas indicadas de acuerdo al modelo de datos.
postgres=# -- *****
postgres=# CREATE TABLE usuario(
postgres(#   id SERIAL PRIMARY KEY,
postgres(#   email VARCHAR(50) NOT NULL
postgres(# );
CREATE TABLE
postgres=#
postgres=# CREATE TABLE post(
postgres(#   id SERIAL PRIMARY KEY,
postgres(#   usuario_id SMALLINT, titulo VARCHAR(50),
postgres(#   fecha DATE NOT NULL,
postgres(#   FOREIGN KEY (usuario_id)
postgres(#   REFERENCES usuario(id)
postgres(# );
CREATE TABLE
postgres=#
postgres=# CREATE TABLE comentario(
postgres(#   id SERIAL PRIMARY KEY,
postgres(#   post_id SMALLINT,
postgres(#   usuario_id SMALLINT,
postgres(#   texto VARCHAR(50),
postgres(#   fecha DATE,
postgres(#   FOREIGN KEY (post_id) REFERENCES post(id),
postgres(#   FOREIGN KEY (usuario_id) REFERENCES usuario(id)
postgres(# );
CREATE TABLE
postgres=#
```

3. Insertar los registros listados en el desafío, para las tablas creadas.

```
SQL Shell (psql)
postgres=#
postgres=# -- 3. Insertar los siguientes registros.
postgres=# -- *****
postgres=# INSERT INTO usuario (email)
postgres=# VALUES
postgres=# ('usuario01@hotmail.com'),
postgres=# ('usuario02@gmail.com'),
postgres=# ('usuario03@gmail.com'),
postgres=# ('usuario04@hotmail.com'),
postgres=# ('usuario05@yahoo.com'),
postgres=# ('usuario06@hotmail.com'),
postgres=# ('usuario07@yahoo.com'),
postgres=# ('usuario08@yahoo.com'),
postgres=# ('usuario09@yahoo.com');
INSERT 0 9
postgres=#
postgres=# SELECT * FROM usuario;
 id | email
-----+-----
  1 | usuario01@hotmail.com
  2 | usuario02@gmail.com
  3 | usuario03@gmail.com
  4 | usuario04@hotmail.com
  5 | usuario05@yahoo.com
  6 | usuario06@hotmail.com
  7 | usuario07@yahoo.com
  8 | usuario08@yahoo.com
  9 | usuario09@yahoo.com
(9 filas)
```

```
SQL Shell (psql)
postgres=#
postgres=# INSERT INTO post (usuario_id, titulo, fecha)
postgres=# VALUES
postgres=# (1,'Post 1: Esto es malo','2020-06-29'),
postgres=# (5,'Post 2: Esto es malo','2020-06-20'),
postgres=# (1,'Post 3: Esto es excelente','2020-05-30'),
postgres=# (9,'Post 4: Esto es bueno','2020-05-09'),
postgres=# (7,'Post 5: Esto es bueno','2020-07-10'),
postgres=# (5,'Post 6: Esto es excelente','2020-07-18'),
postgres=# (8,'Post 7: Esto es excelente','2020-07-07'),
postgres=# (5,'Post 8: Esto es excelente','2020-05-14'),
postgres=# (2,'Post 9: Esto es bueno','2020-05-08'),
postgres=# (6,'Post 10: Esto es bueno','2020-06-02'),
postgres=# (4,'Post 11: Esto es bueno','2020-05-05'),
postgres=# (9,'Post 12: Esto es malo','2020-07-23'),
postgres=# (5,'Post 13: Esto es excelente','2020-05-30'),
postgres=# (8,'Post 14: Esto es excelente','2020-05-01'),
postgres=# (7,'Post 15: Esto es malo','2020-06-17');
INSERT 0 15
postgres=#
postgres=# SELECT * FROM post;
 id | usuario_id | titulo | fecha
-----+-----+-----+-----
  1 |          1 | Post 1: Esto es malo | 2020-06-29
  2 |          5 | Post 2: Esto es malo | 2020-06-20
  3 |          1 | Post 3: Esto es excelente | 2020-05-30
  4 |          9 | Post 4: Esto es bueno | 2020-05-09
  5 |          7 | Post 5: Esto es bueno | 2020-07-10
  6 |          5 | Post 6: Esto es excelente | 2020-07-18
  7 |          8 | Post 7: Esto es excelente | 2020-07-07
  8 |          5 | Post 8: Esto es excelente | 2020-05-14
  9 |          2 | Post 9: Esto es bueno | 2020-05-08
 10 |          6 | Post 10: Esto es bueno | 2020-06-02
 11 |          4 | Post 11: Esto es bueno | 2020-05-05
 12 |          9 | Post 12: Esto es malo | 2020-07-23
 13 |          5 | Post 13: Esto es excelente | 2020-05-30
 14 |          8 | Post 14: Esto es excelente | 2020-05-01
 15 |          7 | Post 15: Esto es malo | 2020-06-17
(15 filas)
```

```
SQL Shell (psql)
postgres=#
postgres=# INSERT INTO comentario (post_id, usuario_id, texto, fecha)
postgres=# VALUES
postgres=# (6,3,'Este es el comentario 1','2020-07-08'),
postgres=# (2,4,'Este es el comentario 2','2020-06-07'),
postgres=# (4,6,'Este es el comentario 3','2020-06-16'),
postgres=# (13,2,'Este es el comentario 4','2020-06-15'),
postgres=# (6,6,'Este es el comentario 5','2020-05-14'),
postgres=# (3,3,'Este es el comentario 6','2020-07-08'),
postgres=# (1,6,'Este es el comentario 7','2020-05-22'),
postgres=# (7,6,'Este es el comentario 8','2020-07-09'),
postgres=# (13,8,'Este es el comentario 9','2020-06-30'),
postgres=# (6,8,'Este es el comentario 10','2020-06-19'),
postgres=# (1,5,'Este es el comentario 11','2020-05-09'),
postgres=# (15,8,'Este es el comentario 12','2020-06-17'),
postgres=# (9,1,'Este es el comentario 13','2020-05-01'),
postgres=# (5,2,'Este es el comentario 14','2020-05-31'),
postgres=# (3,4,'Este es el comentario 15','2020-06-28');
INSERT 0 15
postgres=#
postgres=# SELECT * FROM comentario;
 id | post_id | usuario_id |      texto      |      fecha
-----+-----+-----+-----+-----
  1 |      6 |          3 | Este es el comentario 1 | 2020-07-08
  2 |      2 |          4 | Este es el comentario 2 | 2020-06-07
  3 |      4 |          6 | Este es el comentario 3 | 2020-06-16
  4 |     13 |          2 | Este es el comentario 4 | 2020-06-15
  5 |      6 |          6 | Este es el comentario 5 | 2020-05-14
  6 |      3 |          3 | Este es el comentario 6 | 2020-07-08
  7 |      1 |          6 | Este es el comentario 7 | 2020-05-22
  8 |      7 |          6 | Este es el comentario 8 | 2020-07-09
  9 |     13 |          8 | Este es el comentario 9 | 2020-06-30
 10 |      6 |          8 | Este es el comentario 10 | 2020-06-19
 11 |      1 |          5 | Este es el comentario 11 | 2020-05-09
 12 |     15 |          8 | Este es el comentario 12 | 2020-06-17
 13 |      9 |          1 | Este es el comentario 13 | 2020-05-01
 14 |      5 |          2 | Este es el comentario 14 | 2020-05-31
 15 |      3 |          4 | Este es el comentario 15 | 2020-06-28
(15 filas)
postgres=#
```

4. Seleccionar el correo, id y título de todos los post publicados por el usuario 05.

```
SQL Shell (psql)
postgres=# -- 4. Seleccionar el correo, id y título de todos los post publicados por el usuario 5.
postgres=# -- *****
postgres=# SELECT email,post.id,titulo FROM post INNER JOIN usuario ON post.usuario_id = usuario.id WHERE usuario.id = 5;
 email      | id |      titulo
-----+---+-----
 usuario05@yahoo.com |  2 | Post 2: Esto es malo
 usuario05@yahoo.com |  6 | Post 6: Esto es excelente
 usuario05@yahoo.com |  8 | Post 8: Esto es excelente
 usuario05@yahoo.com | 13 | Post 13: Esto es excelente
(4 filas)
postgres=#
```

5. Listar el correo, id y el detalle de todos los comentarios que no hayan sido realizados por el usuario con email: usuario06@hotmail.com.

```
SQL Shell (psql)
postgres=#
postgres=# -- 5. Listar el correo, id y el detalle de todos los comentarios que no hayan sido realizados por el usuario con
email usuario06@hotmail.com.
postgres=# SELECT email, comentario.id, post_id, texto, fecha FROM comentario INNER JOIN usuario
postgres=# ON comentario.usuario_id = usuario.id WHERE email != 'usuario06@hotmail.com';
    email      | id | post_id |      texto      |      fecha
-----+----+-----+-----+-----
 usuario03@gmail.com | 1 |      6 | Este es el comentario 1 | 2020-07-08
 usuario04@hotmail.com | 2 |      2 | Este es el comentario 2 | 2020-06-07
 usuario02@gmail.com | 4 |     13 | Este es el comentario 4 | 2020-06-15
 usuario03@gmail.com | 6 |      3 | Este es el comentario 6 | 2020-07-08
 usuario08@yahoo.com | 9 |     13 | Este es el comentario 9 | 2020-06-30
 usuario08@yahoo.com | 10 |      6 | Este es el comentario 10 | 2020-06-19
 usuario05@yahoo.com | 11 |      1 | Este es el comentario 11 | 2020-05-09
 usuario08@yahoo.com | 12 |     15 | Este es el comentario 12 | 2020-06-17
 usuario01@hotmail.com | 13 |      9 | Este es el comentario 13 | 2020-05-01
 usuario02@gmail.com | 14 |      5 | Este es el comentario 14 | 2020-05-31
 usuario04@hotmail.com | 15 |      3 | Este es el comentario 15 | 2020-06-28
(11 filas)

postgres=#
```

6. Listar los usuarios que no han publicado ningún post.

```
SQL Shell (psql)
postgres=#
postgres=# -- 6. Listar los usuarios que no han publicado ningún post.
postgres=# -- *****
postgres=# SELECT * FROM usuario LEFT JOIN post ON usuario.id = post.usuario_id WHERE post.usuario_id IS NULL;
 id | email      | id | usuario_id | titulo | fecha
-----+-----+----+-----+-----+-----
  3 | usuario03@gmail.com |   |          |      |
(1 fila)

postgres=#
```

7. Listar todos los post con sus comentarios (incluyendo aquellos que no poseen comentarios).

```
SQL Shell (psql)
postgres=# -- 7. Listar todos los post con sus comentarios (incluyendo aquellos que no poseen comentarios).
postgres=# -- *****
postgres=# SELECT * FROM post FULL OUTER JOIN comentario ON post.id = comentario.post_id;
 id | usuario_id | titulo      | fecha      | id | post_id | usuario_id | texto      | fecha
-----+-----+-----+-----+----+-----+-----+-----+-----
  6 |      5 | Post 6: Esto es excelente | 2020-07-18 | 1 |      6 |      3 | Este es el comentario 1 | 2020-07-08
  2 |      5 | Post 2: Esto es malo | 2020-06-20 | 2 |      2 |      4 | Este es el comentario 2 | 2020-06-07
  9 |      9 | Post 4: Esto es bueno | 2020-05-09 | 3 |      4 |      6 | Este es el comentario 3 | 2020-06-16
 13 |      5 | Post 13: Esto es excelente | 2020-05-30 | 4 |     13 |      2 | Este es el comentario 4 | 2020-06-15
  6 |      5 | Post 6: Esto es excelente | 2020-07-18 | 5 |      6 |      6 | Este es el comentario 5 | 2020-05-14
  3 |      1 | Post 3: Esto es excelente | 2020-05-30 | 6 |      3 |      3 | Este es el comentario 6 | 2020-07-08
  1 |      1 | Post 1: Esto es malo | 2020-06-29 | 7 |      1 |      6 | Este es el comentario 7 | 2020-05-22
  7 |      8 | Post 7: Esto es excelente | 2020-07-07 | 8 |      7 |      6 | Este es el comentario 8 | 2020-07-09
 13 |      5 | Post 13: Esto es excelente | 2020-05-30 | 9 |     13 |      8 | Este es el comentario 9 | 2020-06-30
  6 |      5 | Post 6: Esto es excelente | 2020-07-18 | 10 |      6 |      8 | Este es el comentario 10 | 2020-06-19
  1 |      1 | Post 1: Esto es malo | 2020-06-29 | 11 |      1 |      5 | Este es el comentario 11 | 2020-05-09
  7 |      2 | Post 15: Esto es malo | 2020-06-17 | 12 |     15 |      8 | Este es el comentario 12 | 2020-06-17
  9 |      2 | Post 9: Esto es bueno | 2020-05-08 | 13 |      9 |      1 | Este es el comentario 13 | 2020-05-01
  5 |      7 | Post 5: Esto es bueno | 2020-07-10 | 14 |      5 |      2 | Este es el comentario 14 | 2020-05-31
  3 |      1 | Post 3: Esto es excelente | 2020-05-30 | 15 |      3 |      4 | Este es el comentario 15 | 2020-06-28
(20 filas)

postgres=#
```

8. Listar todos los usuarios que hayan publicado un post en Junio.

Usando "TEXT LIKE" en la consulta a fecha:

```
SQL Shell (psql)

postgres=# -- 8. Listar todos los usuarios que hayan publicado un post en Junio.
postgres=# -- *****
postgres=# SELECT * FROM usuario LEFT JOIN post ON usuario.id = post.usuario_id WHERE post.fecha::TEXT LIKE '2020-06-%';
 id | email | id | usuario_id | titulo | fecha
-----+-----+----+-----+-----+-----
  1 | usuario01@hotmail.com | 1 | 1 | Post 1: Esto es malo | 2020-06-29
  5 | usuario05@yahoo.com | 2 | 5 | Post 2: Esto es malo | 2020-06-20
  6 | usuario06@hotmail.com | 10 | 6 | Post 10: Esto es bueno | 2020-06-02
  7 | usuario07@yahoo.com | 15 | 7 | Post 15: Esto es malo | 2020-06-17
(4 filas)

postgres=#
```

Otra Forma usando BETWEEN en la consulta a fecha:

```
SQL Shell (psql)

postgres=# SELECT * FROM usuario LEFT JOIN post ON usuario.id = post.usuario_id WHERE post.fecha BETWEEN '2020-06-01' AND '2020-06-30';
 id | email | id | usuario_id | titulo | fecha
-----+-----+----+-----+-----+-----
  1 | usuario01@hotmail.com | 1 | 1 | Post 1: Esto es malo | 2020-06-29
  5 | usuario05@yahoo.com | 2 | 5 | Post 2: Esto es malo | 2020-06-20
  6 | usuario06@hotmail.com | 10 | 6 | Post 10: Esto es bueno | 2020-06-02
  7 | usuario07@yahoo.com | 15 | 7 | Post 15: Esto es malo | 2020-06-17
(4 filas)

postgres=#
```