

# Requisitos SQL y DW



# Herramientas necesarias

- Navegador web, preferiblemente Chrome, ya que usaremos Google Cloud Platform y en otros navegadores podremos encontrar alguna incompatibilidad.  
<https://www.google.com/chrome/>
- Durante las clases usaré Visual Studio Code para editar algunas partes del desarrollo de queries. Sois libres de usar este mismo editor o cualquier otro con el que os sintáis cómodos.  
<https://code.visualstudio.com/Download>
- Para la iniciación a bases de datos vamos a usar PostgreSQL. Recomendando crear una instancia en <https://www.tembo.io/>, pero sentíos libres de levantar un docker y tenerlo en local.
- Para la gestión de la base de datos voy a usar TablePlus <https://tableplus.com/>, pero igual, sentíos libres de usar otros como DBeaver...

# Postgre SQL



- Crear una cuenta en [tembo.io](https://tembo.io) TRY FREE

# Postgre SQL



- Selecciona Stack Standard – Deploy Free

**Select a Stack**

Stacks are specialized Postgres instances that are designed for your workload. Click any stack to learn more.

[Deploy Pro](#) [Deploy Free](#)

**OLTP**  
A low latency transactional processing stack with high I/O performance, concurrency, and real-time metrics.  
7 extensions: transaction-processing, real-time-operations, operational data

**VectorDB**  
An alternative to Pinecone, Weaviate, and Chroma, built on Postgres. Store and search vectors with pgvector. Manage the lifecycle of embeddings with pgvectorize.  
4 extensions: vector-search, similarity-search, pgvector, embeddings

**ParadeDB**  
Postgres for Search and Analytics  
7 extensions: elasticsearch, full-text-search, bm25, hybrid-search

**Standard**

**Overview**  
sql-database, balanced, getting started, build your own

**Configuration**  
Our default configuration of Postgres. The following configurations scale based on the size of cpu, memory, and storage that is selected for the instance; shared\_buffers, max\_connections, work\_mem, bgwriter\_delay, effective\_cache\_size, maintenance\_work\_mem, and max\_wal\_size.

**Extensions**  
pg\_stat\_statements comes pre-installed and enabled. Extensions in pgt.dev can be installed on-demand.

# Postgre SQL



- En Show Connections puedes ver los detalles para la conexión con base de datos

The screenshot shows a dark-themed 'Connections' window. At the top, it says 'Select database' with three dropdown menus: 'Instance' (instance-1), 'Database' (postgres), and 'Role' (superuser). Below these is a 'URI' section with a 'PSQL' label and a text area containing the connection string: 'postgresql://postgres:.....@effusively-democratic-chimp.data-1.use1.tembo.io:5432/postgres'. To the right of the text area are download and copy icons. Below the URI section is a 'Certificate Validation: OFF' toggle. At the bottom, there is a 'Details' section with a table of connection parameters.

Details	
Hostname	Port
effusively-democratic-chimp.data-1.use1.tembo.io	5432
Username	Password

There is a 'Change Password' link next to the Password field.

# Postgre SQL



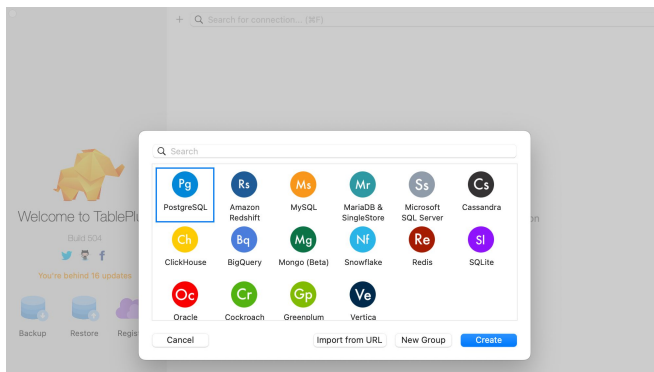
- Instala TablePlus <https://tableplus.com/download>



# Postgre SQL



- Crea una nueva conexión a nuestra base de datos en PostgreSQL



# Postgre SQL



- Introduce los datos de la conexión y los testeamos. Los datos están en la pestaña Show Connection Strings (Tembo). Estos son mis datos:

Details

Hostname	Port
effusively-democratic-chimp.data-1.use1.tembo.io	5432
Username	Password
postgres	..... <a href="#">Change Password</a>

PostgreSQL Connection

Name: tembo

Status color: ☐ ☐ ☐ ☐ ☐ ☐ Tag: local

Host/Socket: effusively-democratic-chimp.data- Port: 5432

User: postgres Other options: ▾

Password: ..... Store in keychain: ▾

Database: postgres Bootstrap commands: ▾

SSL mode: PREFERRED

SSL keys: Key... Cert... CA Cert... -

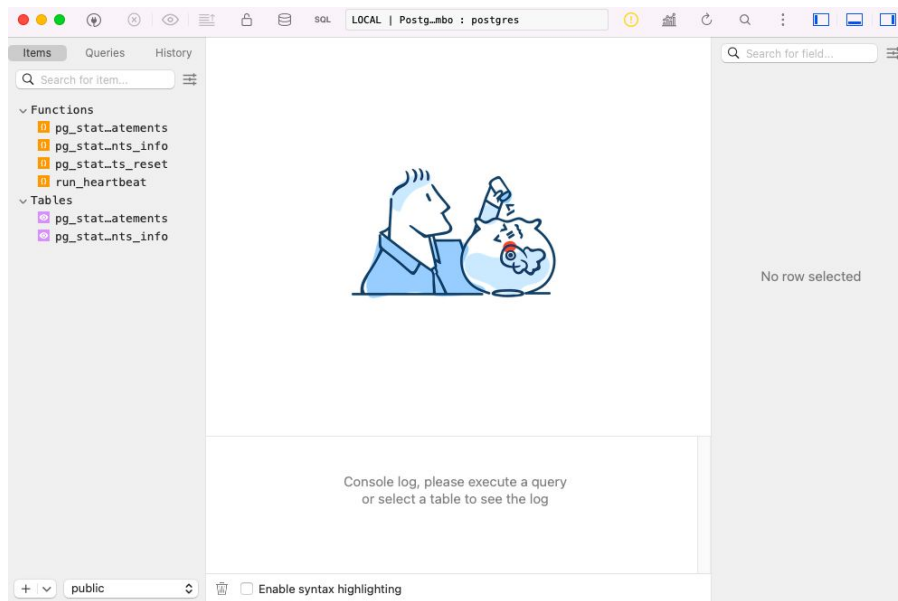
Over SSH Save Test Connect



# Postgre SQL



- Una vez hecha la conexión debemos ver una UI como esta:



# BigQuery



También usaremos la base de datos analítica de Google Cloud Platform, BigQuery. Es un almacén de datos completamente administrado, es decir, sin servidores que gestionar.

- Permite usar SQL para las consultas.
- A escala de Petabytes.
- Dispone de un editor gráfico para realizar las consultas.

# BigQuery



Usaremos la zona de pruebas de BigQuery, que aunque tiene algunas limitaciones para nuestro uso será suficiente.

<https://cloud.google.com/bigquery/docs/introduction>

Para habilitar la zona de pruebas de BigQuery:

<https://cloud.google.com/bigquery/docs/sandbox>

En el anterior enlace encontrareis una pequeña guía introductoria para comenzar a usar la zona de pruebas, vídeo incluido.



# BigQuery

En el siguiente enlace podréis entrar a la zona de pruebas de BigQuery, tenéis la posibilidad de usar vuestra cuenta de Gmail o crear una nueva. Lo dejo a vuestra elección, yo crearé una nueva para aislar el trabajo que hagamos en esta asignatura y para asegurar que ese correo no tiene ninguna cuenta de facturación asociada en Google Cloud en la que podamos incurrir en gastos innecesarios.

<https://console.cloud.google.com/bigquery>

# BigQuery

Una vez estéis logueados deberéis aceptar las condiciones del servicio de Google Cloud



## Te damos la bienvenida

Crea y administra tus instancias, discos, redes y otros recursos de Google Cloud desde un solo lugar.

País

España

## Condiciones del Servicio

☒ Acepto las [Condiciones del Servicio de Google Cloud Platform](#) y de [las API y los servicios aplicables](#).

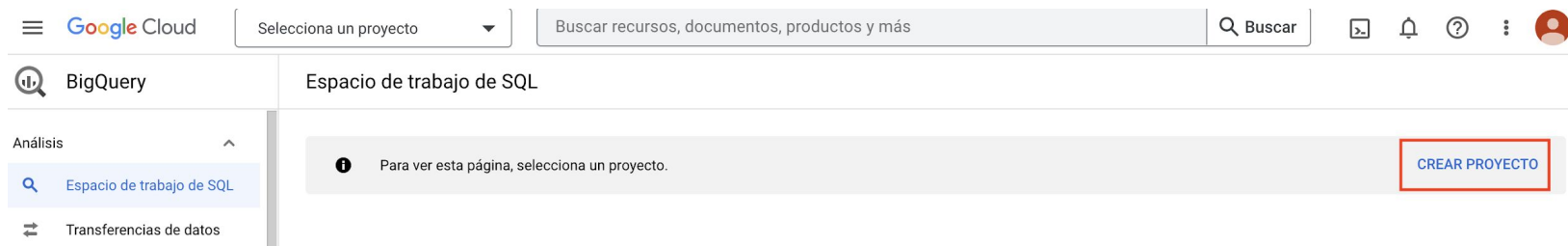
## Actualizaciones por correo electrónico

☐ Quiero recibir correos electrónicos periódicos sobre novedades, actualizaciones de productos y ofertas especiales de Google Cloud y Google Cloud Partners.

[ACEPTAR Y CONTINUAR](#)

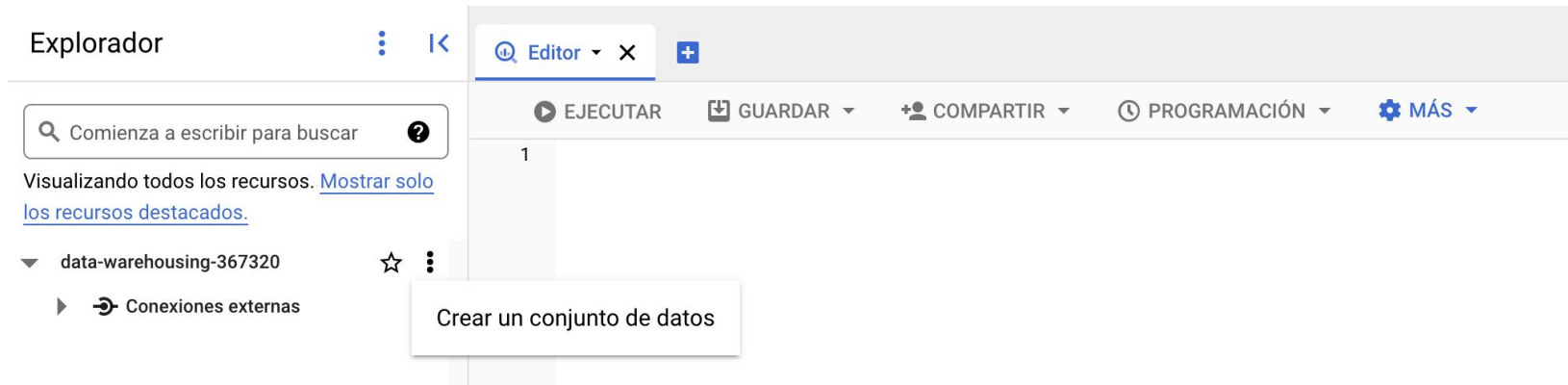
# BigQuery

Una vez dentro del servicio tendréis que crear un proyecto, éste debe tener un nombre único.



# BigQuery

Veremos en la parte izquierda de la interfaz el nombre de nuestro proyecto y pulsando sobre los 3 puntitos la opción de crear un conjunto de datos (dataset), pulsamos sobre ésta.



# BigQuery

Crearemos un conjunto de datos con nombre keepcoding, es importante que uséis este nombre para que el código de ejemplo os funcione correctamente. Podéis elegir la región que esté más cerca de vuestra ubicación.

## Crea un conjunto de datos

ID del proyecto  
data-warehousing-367320 [CAMBIO](#)

ID del conjunto de datos \*  
keepcoding

Puede incluir letras, números y guiones bajos

Ubicación de los datos  
europe-southwest1 (Madrid) ▼ ⓘ

### Vencimiento predeterminado de la tabla

☐ Habilitar el vencimiento de la tabla ⓘ

Máxima antigüedad predeterminada de la tabla Days

### Opciones avanzadas ▼

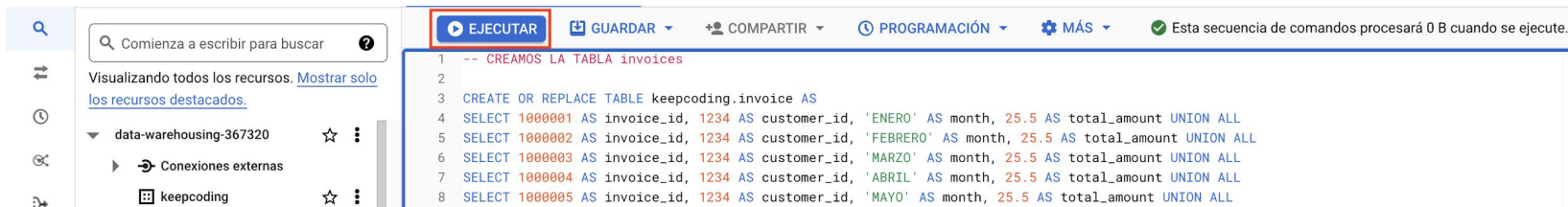
CREAR CONJUNTO DE DATOS

CANCELAR



# BigQuery

Copiaremos todo el contenido del fichero tablas.sql, que está en Github, en el editor de BigQuery y pulsaremos ejecutar.



# BigQuery

Una vez ejecutado este código correctamente aparecerán dos nuevas tablas dentro el dataset keepcoding.

Explorador

Visualizando todos los recursos. [Mostrar solo los recursos destacados.](#)

- data-warehousing-367320
  - Conexiones externas
  - keepcoding
    - customer
    - invoice

\*Consulta no guardada

EJECUTAR GUARDAR COMPARTIR PROGRAMACIÓN MÁS

Se completó la consulta.

```
41 -- CREAMOS LA TABLA CUSTOMER
42
43 CREATE OR REPLACE TABLE keepcoding.customer AS
44 SELECT 9999 customer_id, 'Estela Muñoz' AS name, 31 AS age
```

Presiona Alt + F1 para ver las opciones de accesibilidad

Todos los resultados

Tiempo transcurrido	Declaraciones procesadas	Estado del trabajo
4 s	2	✓ SUCCESS

Estado	Hora de finalización	SQL	Etapas finalizadas	Bytes procesados	Acción
✓	16:36 [3:1]	CREATE OR REPLACE TABLE keepcoding.invoice AS SELECT ...	3	0 B	VER RESULTADOS
✓	16:36 [43:1]	CREATE OR REPLACE TABLE keepcoding.customer AS SELEC...	3	0 B	VER RESULTADOS



# BigQuery

Con estos pasos ya tenemos el entorno listo para iniciar la asignatura, espero que os haya resultado útil y ante cualquier dificultad comentamos por Discord.



# KEEPCODING

## Tech School

Madrid | Barcelona | Bogotá

**Datos de contacto**