

# PROYECTO SQL

---

TRABAJO CON BASES DE DATOS



# SQL

---

1. Se trata de instalar en Docker diferentes bases de datos
  1. MYSQL, SQLSERVER, ORACLE, POSTRGRES
2. Carga de datos de la base Northwidn
3. Hacer un servidor web que acceda a las distintas bases de datos

```
docker run --name curso-mysql -p 3306:3306 -e  
MYSQL_ROOT_PASSWORD=my-secret-pw -d mysql:8.0.28
```

<https://dbeaver.io/download/>

<https://www.dbvis.com/download/>

Docker [https://hub.docker.com/\\_/phpmyadmin](https://hub.docker.com/_/phpmyadmin)

Instalar mysql

Download bveaver

Download dbvisualizer

Instalar phpadmin

[https://hub.docker.com/\\_/postgres](https://hub.docker.com/_/postgres)

```
docker run --name curso-pg-14 -e POSTGRES_PASSWORD=my-secret-pw -p  
5437:5432 -d postgres:13
```

[https://hub.docker.com/\\_/microsoft-mssql-server](https://hub.docker.com/_/microsoft-mssql-server)

```
docker run -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=my-secret-pw" -p 1433:1433 -d  
mcr.microsoft.com/mssql/server:2019-CU15-ubuntu-20.04
```

carga de datos mysql

<https://www.aspsnippets.com/Handlers/DownloadFile.ashx?File=9cb579c6-86db-4596-84c3-d549428fdcf5.zip>

carga de datos pg

[https://raw.githubusercontent.com/pthom/northwind\\_psql/master/northwind.sql](https://raw.githubusercontent.com/pthom/northwind_psql/master/northwind.sql)

carga de datos sql server

<https://raw.githubusercontent.com/microsoft/sql-server-samples/master/samples/databases/northwind-pubs/instnwnd.sql>

# MYSQL

```
const mysql = require("mysql8")

var pool = mysql.createPool({
  connectionLimit: 10,
  host: 'localhost',
  user: 'root',
  password: 'my-secret-pw',
  database: 'northwind'
});

function q(sql) {
  return new Promise((resolver, reject) => {
    pool.query(sql, function (error, results, fields) {
      if (error) reject(error);
      return resolver(results);
    });
  });
}
```

# SQL SERVER

```
const mssql = require('mssql')
const sqlConfig = {
  user: process.env.SQLSERVER_USER,
  password: process.env.SQLSERVER_PASSWORD,
  database: process.env.SQLSERVER_DATABASE,
  server: process.env.SQLSERVER_HOST,
  pool: {
    max: 10,
    min: 0,
    idleTimeoutMillis: 30000
  },
  options: {
    encrypt: true,
    trustServerCertificate: true
  }
}
```

```
async function q(sql, params) {
  try {

    await mssql.connect(sqlConfig)
    const result = await
mssql.query(`select * from Customers`)
    return result;
  } catch (err) {
    return {err:JSON.stringify(err)}
  }
}
```



```
const { Pool } = require('pg');
const util = require("util")

const poolPg = new Pool({
  user: process.env.PG_USER,
  host: process.env.PG_HOST,
  database: 'postgres',
  password: process.env.PG_PASSWORD,
  port: process.env.PG_PORT,
})
```

```
async function q(sql, parametros) {
  return new Promise(async (resolve, reject) => {
    poolPg.connect((err, client, done) => {
      if (err) reject(err)
      client.query(sql, parametros, (err, result) => {
        done()
        if (err) {
          reject(err)
        } else {
          resolve(result.rows)
        }
      })
    })
  });
}

module.exports = {
  q
}
```

# POSTGRES