

## Practica Sqoop

Atención: por defecto, el puerto de postgres es 5432, pero puede variar de acuerdo a la configuración hecha.

Para comprobar el puerto de postgres, ejecutar `docker inspect edvai_postgres` y verificar el puerto, otra manera es ingresar al container de Docker de postgres (`docker exec -it edvai_postgres bash`). En el file `/var/lib/postgresql/pgdata/postgresql.conf` está definido el puerto.

Verificar también que el container de Docker de postgres esté ejecutándose.

Contraseña postgres: edvai

### Comandos:

Listar databases:

```
sqoop list-databases \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--username postgres -P
```

Listar tablas:

```
sqoop list-tables \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--username postgres -P
```

Ejecutar queries

```
sqoop eval \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--username postgres \  
--P \  
--query "select * from region limit 10"
```

Importar tablas:

```
sqoop import \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--username postgres \  
--table region \  
--m 1 \  
--P \  
--target-dir /sqoop/ingest \  
--as-parquetfile \  
--delete-target-dir
```

Importar tablas con filtro:

```
sqoop import \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--where "region = 1"
```

```
--username postgres\  
--table region\  
--m 1 \  
--P \  
--target-dir /sqoop/ingest/southern \  
--as-parquetfile \  
--where "region_description = 'Southern' " \  
--delete-target-dir
```

Ejercicios:

- 1) Mostrar las tablas de la base de datos northwind
- 2) Mostrar los clientes de Argentina
- 3) Importar un archivo .parquet que contenga toda la tabla orders. Luego ingestar el archivo a HDFS (carpeta /sqoop/ingest)
- 4) Importar un archivo .parquet que contenga solo los productos con mas 20 unidades en stock, de la tabla Products . Luego ingestar el archivo a HDFS (carpeta ingest)

## Practica Nifi

<https://localhost:8443/nifi>

- 1) En el shell de Nifi, crear un script .sh que descargue el archivo starwars.csv al directorio /home/nifi/ingest (crearlo si es necesario). Ejecutarlo con `./home/nifi/ingest/ingest.sh`  
  
<https://github.com/fpineyro/homework-0/blob/master/starwars.csv>
- 2) Usando procesos en Nifi:
  - a) tomar el archivo starwars.csv desde el directorio /home/nifi/ingest.
  - b) Mover el archivo starwars.csv desde el directorio anterior, a /home/nifi/bucket (crear el directorio si es necesario)
  - c) Tomar nuevamente el archivo, ahora desde /home/nifi/bucket
  - d) Ingestarlo en HDFS/nifi (si es necesario, crear el directorio con `hdfs dfs -mkdir /nifi`)

Atencion:

- Para que Nifi pueda ingestar el archivo a HDFS, debe asignársele el permiso desde la consola de Hadoop con el comando `hdfs dfs -chmod 777 /nifi`
- Desde la consola de nifi, es necesario agregar dos archivos de configuración llamados `core-site.xml` y `hdfs-site.xml` al directorio `/home/nifi/hadoop` (crearlo si es necesario). Al final de este archivo está detallado cuál debe ser el contenido de ambos archivos
- Luego desde la interfaz gráfica de Nifi, al crear el proceso de ingesta de HDFS se debe definir en 'Properties/Hadoop Configuration Resources' la ruta a los archivos de configuración: `/home/nifi/hadoop/core-site.xml`, `/home/nifi/hadoop/hdfs-site.xml`

### Processor Details

▶ Running⚙ STOP & CONFIGURE

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Hadoop Configuration Resources	<input type="text" value="/home/nifi/hadoop/core-site.xml, /home/nifi/hadoop/hdf..."/>
Kerberos Credentials Service	<input type="text" value="No value set"/>
Kerberos User Service	<input type="text" value="No value set"/>
Kerberos Principal	<input type="text" value="No value set"/>
Kerberos Keytab	<input type="text" value="No value set"/>

## Core-site.xml de ejemplo

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>
  <property>
    <name>fs.defaultFS</name>
    <value>hdfs://172.17.0.2:9000</value>
  </property>
</configuration>
```

## Hdfs-site.xml de ejemplo

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

<!-- Put site-specific property overrides in this file. -->

<configuration>

  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>

  <property>
    <name>dfs.name.dir</name>
    <value>file:///home/hadoop/hadoopdata/hdfs/namenode</value>
  </property>

  <property>
    <name>dfs.data.dir</name>
    <value>file:///home/hadoop/hadoopdata/hdfs/datanode</value>
  </property>
</configuration>
```