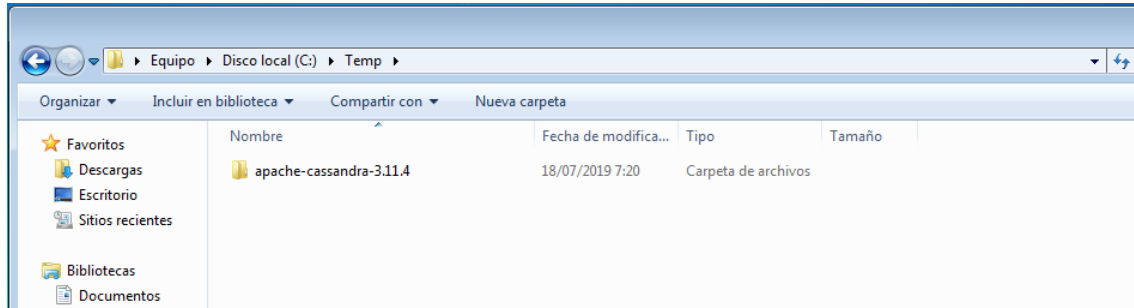# 2.- NoSQL - Cassandra

Descargamos el comprimido de Cassandra y lo descomprimimos en un directorio

Accedemos al directorio *bin* de la instalación y ejecutamos cassandra –f (foreground), para arrancar el servidor

Procedemos a arrancar el cliente, cqlsh.py

```
E:\Temp\apache-cassandra-3.11.4\bin>cqlsh.py

WARNING: console codepage must be set to cp65001 to support utf-8 encoding on Wi
ndows platforms.
If you experience encoding problems, change your console codepage with 'chcp 650
01' before starting cqlsh.

Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.11.4 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
WARNING: pyreadline dependency missing.  Install to enable tab completion.
cqlsh>
```

Creamos un KEYSPACE llamado 'berenjena', que es el contenedor de la información almacenada

```
cqlsh> CREATE KEYSPACE berenjena WITH REPLICATION = { 'class' : 'SimpleStrategy'
, 'replication_factor' : 1 };
```

Y entramos a él con *use*

```
cqlsh> use berenjena;
cqlsh:berenjena>
```

Una vez dentro procedemos a crear una *column family*, que es semejante a una tabla, en ete caso con los campos *'nombre', 'apodo', 'apellido' y 'organizacion'*

```
cqlsh:berenjena>   CREATE TABLE alumnos (
           ...      nombre text,
           ...      apodo text,
           ...      apellido text,
           ...      organizacion text,
           ...      PRIMARY KEY (nombre));
cqlsh:berenjena>
```

Y procedmeos a insertar valores adicha column family

```
cqlsh:berenjena> INSERT INTO alumnos (nombre, apodo,apellido, organizacion) VALU
ES ('Dan', '"Programitas"', 'Grigore', 'FEI');
cqlsh:berenjena>
```

```
cqlsh:berenjena> SELECT * FROM alumnos;

 nombre   | apellido | apodo          | organizacion
----------+----------+----------------+--------------
 Svetlana | Makarova |        "Rusa"  |          FEI
   Ismael |  Velasco |      "Soopas"  |          FEI
      Dan |  Grigore | "Programitas"  |          FEI

(3 rows)
cqlsh:berenjena>
```

Diferencias entre Cassanadra, MongoDB y Hadoop

Editorial information provided by DB-Engines

| Name	Cassandra X	HBase X	MongoDB X
Description	Wide-column store based on ideas of BigTable and DynamoDB ℹ	Wide-column store based on Apache Hadoop and on concepts of BigTable	One of the most popular document stores
Primary database model	Wide column store	Wide column store	Document store
DB-Engines Ranking ℹ Trend Chart	Score 127.00 Rank #10 Overall #1 Wide column stores	Score 57.54 Rank #18 Overall #2 Wide column stores	Score 409.93 Rank #5 Overall #1 Document stores
Website	cassandra.apache.org	hbase.apache.org	www.mongodb.com
Technical documentation	cassandra.apache.org/doc/latest	hbase.apache.org	docs.mongodb.com/manual
Developer	Apache Software Foundation ℹ	Apache Software Foundation ℹ	MongoDB, Inc
Initial release	2008	2008	2009
Current release	3.11.4, February 2019	2.2.0, June 2019	4.0.10, May 2019
License ℹ	Open Source ℹ	Open Source ℹ	Open Source ℹ
Cloud-based only ℹ	no	no	no ℹ
DBaaS offerings (sponsored links) ℹ			MongoDB Atlas: Deploy a fully managed cloud database in minutes.
Implementation language	Java	Java	C++
Server operating systems	BSD Linux OS X Windows	Linux Unix Windows ℹ	Linux OS X Solaris Windows
Data scheme	schema-free	schema-free	schema-free ℹ
Typing ℹ	yes	no	yes ℹ
XML support ℹ	no	no	
Secondary indexes	restricted ℹ	no	yes
SQL ℹ	SQL-like SELECT, DML and DDL statements (CQL)	no	Read-only SQL queries via the MongoDB Connector for BI
APIs and other access methods	Proprietary protocol ℹ Thrift	Java API RESTful HTTP API Thrift	proprietary protocol using JSON
Supported programming languages	C# C++ Clojure Erlang Go Haskell Java JavaScript ℹ Perl PHP Python Ruby Scala	C C# C++ Groovy Java PHP Python Scala	Actionscript ℹ C C# C++ Clojure ℹ ColdFusion ℹ D ℹ Dart ℹ Delphi ℹ Erlang Go ℹ Groovy ℹ Haskell Java JavaScript Lisp ℹ Lua ℹ MatLab ℹ Perl PHP PowerShell ℹ Prolog ℹ R ℹ Ruby Scala Smalltalk ℹ
Server-side scripts ℹ	no	yes ℹ	JavaScript
Triggers	yes	yes	no
Partitioning methods ℹ	Sharding ℹ	Sharding	Sharding
Replication methods ℹ	selectable replication factor ℹ	selectable replication factor	Master-slave replication
MapReduce ℹ	yes	yes	yes
Consistency concepts ℹ	Eventual Consistency Immediate Consistency ℹ	Immediate Consistency	Eventual Consistency Immediate Consistency ℹ
Foreign keys ℹ	no	no	no ℹ
Transaction concepts ℹ	no ℹ	no	Multi-document ACID Transactions with snapshot isolation
Concurrency ℹ	yes	yes	yes
Durability ℹ	yes	yes	yes ℹ
In-memory capabilities ℹ	no	no	yes ℹ
User concepts ℹ	Access rights for users can be defined per object	Access Control Lists (ACL) ℹ	Access rights for users and roles