Wiem CHOUCHANE

M2 APP LS1

Lab 1 - Hbase

1. Hbase CLI

(Voir doc : https://hbase.apache.org/book.html)

1.1.1 Commandes de base :

- Connexion SSH au cluster Hadoop :

```
ssh wiem.chouchane@hadoop-edge01.efrei.online
```

- Initialisation ticket Kerberos :

```
kinit wiem.chouchane
```

- Lancement de l'outil shell de HBase :

```
hbase shell
```

- Commandes et output :
    - Commande permettant d'afficher les infos sur le cluster HBase :

```
hbase(main):001:0> status
1 active master, 1 backup masters, 3 servers, 0 dead, 2.3333 average load
Took 0.3439 seconds
hbase(main):002:0>
```

    - Commande permettant d'afficher la version de Hbase utilisée :

```
hbase(main):002:0> version
2.2.4.1.0.3.0-223, rUnknown, Wed Jul 28 00:29:09 CEST 2021
Took 0.0007 seconds
hbase(main):003:0>
```

- Commande permettant d'afficher les infos sur l'utilisateur courant :

```
hbase(main):003:0> whoami
wiem.chouchane@EFREI.ONLINE (auth:KERBEROS)
    groups: wiem.chouchane
Took 0.0350 seconds
hbase(main):004:0>
```

- Liste des tables du cluster Hbase :

```
hbase(main):004:0> list
TABLE
ns_dany_sonethavy:my_table
ns_lucas_bakalian:table_example
2 row(s)
Took 0.0593 seconds
=> ["ns_dany_sonethavy:my_table", "ns_lucas_bakalian:table_example"]
hbase(main):005:0>
```

- Déconexion du HBase shell :

```
hbase(main):005:0> exit
[wiem.chouchane@hadoop-edge01 ~]$
```

1.1.2 Création du namespace

```
create_namespace "ns_wiem_chouchane"
```

1.1.3 Création de table

- Création table :

```
create "ns_wiem_chouchane:library" , {NAME => "author", VERSIONS => 2},
{NAME => "book", VERSIONS => 3}
```

- Description table :

```
describe "ns_wiem_chouchane:library"
```

```
hbase(main):003:0> describe "ns_wiem_chouchane:library"
Table ns_wiem_chouchane:library is ENABLED
ns_wiem_chouchane:library
COLUMN FAMILIES DESCRIPTION
{NAME => 'author', VERSIONS => '2', EVICT_BLOCKS_ON_CLOSE => 'false', NEW_VERSION_BEHAVIOR => 'false', KEEP_DELETED_CELL
S => 'FALSE', CACHE_DATA_ON_WRITE => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', MIN_VERSIONS => '0', REPL
ICATION_SCOPE => '0', BLOOMFILTER => 'ROW', CACHE_INDEX_ON_WRITE => 'false', IN_MEMORY => 'false', CACHE_BLOOMS_ON_WRITE
 => 'false', PREFETCH_BLOCKS_ON_OPEN => 'false', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '65536'}

{NAME => 'book', VERSIONS => '3', EVICT_BLOCKS_ON_CLOSE => 'false', NEW_VERSION_BEHAVIOR => 'false', KEEP_DELETED_CELLS
=> 'FALSE', CACHE_DATA_ON_WRITE => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', MIN_VERSIONS => '0', REPLIC
ATION_SCOPE => '0', BLOOMFILTER => 'ROW', CACHE_INDEX_ON_WRITE => 'false', IN_MEMORY => 'false', CACHE_BLOOMS_ON_WRITE =
> 'false', PREFETCH_BLOCKS_ON_OPEN => 'false', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '65536'}

2 row(s)

QUOTAS
0 row(s)
Took 0.3519 seconds
hbase(main):004:0>
```

1.1.4 Ajout de valeurs

```
put "ns_wiem_chouchane:library", "vhugo", "author:lastname", "Hugo"
```

```
hbase(main):004:0> put "ns_wiem_chouchane:library", "vhugo", "author:lastname", "Hugo"
Took 0.0808 seconds
hbase(main):005:0>
```

```
put "ns_wiem_chouchane:library", "vhugo", "author:firstname", "Victor"
```

Les autres output de PUT sont similaires à celui du dessus donc ne seront pas pris en screen.

```
put "ns_wiem_chouchane:library", "vhugo", "book:title", "La légende des siècles"
```

```
put "ns_wiem_chouchane:library", "vhugo", "book:category", "Poemes"
```

```
put "ns_wiem_chouchane:library", "vhugo", "book:year", "1855"
```

```
put "ns_wiem_chouchane:library", "vhugo", "book:year", "1877"
```

```
put "ns_wiem_chouchane:library", "vhugo", "book:year", "1883"
```

```
put "ns_wiem_chouchane:library", "jverne", "author:lastname", "Jules"
```

```
put "ns_wiem_chouchane:library", "jverne", "author:firstname", "Verne"
```

```
put "ns_wiem_chouchane:library", "jverne", "book:publisher", "Hetzel"
```

```
put "ns_wiem_chouchane:library", "jverne", "book:title", "Face au drapeau"
```

```
put "ns_wiem_chouchane:library", "jverne", "book:year", "1896"
```

1.1.5 Comptage des tuples de la table library :

```
count "ns_wiem_chouchane:library"
```

```
hbase(main):002:0> count "ns_wiem_chouchane:library"
2 row(s)
Took 0.0906 seconds
=> 2
hbase(main):003:0>
```

Ici, il n'y a que 2 tuples donc pas besoin de configurer de cache.

1.1.6 Récupération des valeurs

- Récupération des valeurs de toutes les colonnes identifiées par la clé "vhugo" :

```
get "ns_wiem_chouchane:library", "vhugo"
```

```
hbase(main):003:0> get "ns_wiem_chouchane:library", "vhugo"
COLUMN                          CELL
 author:firstname               timestamp=1635938399800, value=Victor
 author:lastname                timestamp=1635938268927, value=Hugo
 book:category                  timestamp=1635938611672, value=Poemes
 book:title                     timestamp=1635938573689, value=La legende des siecles
 book:year                      timestamp=1635938675684, value=1883
1 row(s)
Took 0.0548 seconds
```

- Récupération des valeurs des colonnes appartenant à la column family "author" et identifiées par la clé "vhugo" :

```
get "ns_wiem_chouchane:library", "vhugo", "author"
```

```
hbase(main):004:0> get "ns_wiem_chouchane:library", "vhugo", "author"
COLUMN                          CELL
 author:firstname               timestamp=1635938399800, value=Victor
 author:lastname                timestamp=1635938268927, value=Hugo
1 row(s)
Took 0.0227 seconds
```

- Récupération des valeurs de la colonne "firstname" appartenant à la column family "author" et identifiée par la clé "vhugo" :

```
get "ns_wiem_chouchane:library", "vhugo", "author:firstname"
```

```
hbase(main):005:0> get "ns_wiem_chouchane:library", "vhugo", "author:firstname"
COLUMN                          CELL
 author:firstname               timestamp=1635938399800, value=Victor
1 row(s)
Took 0.0121 seconds
```

- Récupération des valeurs des colonnes appartenant à la column family "book" et identifiées par la clé "jverne" :

```
get "ns_wiem_chouchane:library", "jverne", COLUMN => "book"
```

```
hbase(main):006:0> get "ns_wiem_chouchane:library", "jverne", COLUMN => "book"
COLUMN                          CELL
 book:publisher                 timestamp=1635938798726, value=Hetzel
 book:title                     timestamp=1635938884118, value=Face au drapeau
 book:year                      timestamp=1635938914037, value=1896
1 row(s)
Took 0.0245 seconds
```

- Récupération des valeurs des colonnes "title", "year" et "publisher" appartenant à la column family "book" et identifiées par la clé "jverne" :

```
get "ns_wiem_chouchane:library", "jverne", COLUMN => ["book:title", "book:year",
"book:publisher"]
```

```
hbase(main):007:0> get "ns_wiem_chouchane:library", "jverne", COLUMN => ["book:title", "book:year", "book:publisher"]
COLUMN                          CELL
 book:publisher                 timestamp=1635938798726, value=Hetzel
 book:title                     timestamp=1635938884118, value=Face au drapeau
 book:year                      timestamp=1635938914037, value=1896
1 row(s)
Took 0.0242 seconds
```

- Récupération des valeurs correspondant au filtre par valeur appliqué (c'est-à-dire valant ici "Jules") et identifiées par la clé "jverne" :

```
get "ns_wiem_chouchane:library", "jverne", FILTER => "ValueFilter(=,
'binary:Jules')"
```

```
hbase(main):009:0> get "ns_wiem_chouchane:library", "jverne", FILTER => "ValueFilter(=, 'binary:Jules')"
COLUMN                          CELL
 author:lastname                timestamp=1635938718627, value=Jules
1 row(s)
Took 0.0600 seconds
```

1.1.7 Navigation dans les tuples :

- Scan de toutes les données de la table "library" :

```
scan "ns_wiem_chouchane:library"
```

```
hbase(main):010:0> scan "ns_wiem_chouchane:library"
ROW                             COLUMN+CELL
 jverne                         column=author:firstname, timestamp=1635938761462, value=Verne
 jverne                         column=author:lastname, timestamp=1635938718627, value=Jules
 jverne                         column=book:publisher, timestamp=1635938798726, value=Hetzel
 jverne                         column=book:title, timestamp=1635938884118, value=Face au drapeau
 jverne                         column=book:year, timestamp=1635938914037, value=1896
 vhugo                          column=author:firstname, timestamp=1635938399800, value=Victor
 vhugo                          column=author:lastname, timestamp=1635938268927, value=Hugo
 vhugo                          column=book:category, timestamp=1635938611672, value=Poemes
 vhugo                          column=book:title, timestamp=1635938573689, value=La legende des siecles
 vhugo                          column=book:year, timestamp=1635938675684, value=1883
2 row(s)
Took 0.0328 seconds
```

- Scan des données de la column family "book" :

```
scan "ns_wiem_chouchane:library", COLUMN => "book"
```

```
hbase(main):012:0> scan "ns_wiem_chouchane:library", COLUMN => "book"
ROW                             COLUMN+CELL
 jverne                         column=book:publisher, timestamp=1635938798726, value=Hetzel
 jverne                         column=book:title, timestamp=1635938884118, value=Face au drapeau
 jverne                         column=book:year, timestamp=1635938914037, value=1896
 vhugo                          column=book:category, timestamp=1635938611672, value=Poemes
 vhugo                          column=book:title, timestamp=1635938573689, value=La legende des siecles
 vhugo                          column=book:year, timestamp=1635938675684, value=1883
2 row(s)
Took 0.0200 seconds
```

- Scan des données de la colonne "year" appartenant à la column family "book" :

```
scan "ns_wiem_chouchane:library", COLUMN => "book:year"
```

```
hbase(main):013:0> scan "ns_wiem_chouchane:library", COLUMN => "book:year"
ROW                             COLUMN+CELL
 jverne                         column=book:year, timestamp=1635938914037, value=1896
 vhugo                          column=book:year, timestamp=1635938675684, value=1883
2 row(s)
Took 0.0070 seconds
```

- Scan des données des colonnes appartenant à la column family "author" et identifiées par une clé commençant par une lettre comprise entre a et n (sans filtre) :

```
scan "ns_wiem_chouchane:library", COLUMN => "author", STARTROW => "a", STOPROW =>
"n"
```

```
hbase(main):014:0> scan "ns_wiem_chouchane:library", COLUMN => "author", STARTROW => "a", STOPROW => "n"
ROW                             COLUMN+CELL
 jverne                         column=author:firstname, timestamp=1635938761462, value=Verne
 jverne                         column=author:lastname, timestamp=1635938718627, value=Jules
1 row(s)
Took 0.0164 seconds
```

- Scan des données des colonnes appartenant à la column family "author" et identifiées par une clé commençant par une lettre comprise entre a et n (avec filtre) :

```
scan "ns_wiem_chouchane:library", COLUMN => "author", FILTER => "RowFilter(>=,
'binary:a') AND RowFilter(<=, 'binary:n')"
```

```
hbase(main):022:0> scan "ns_wiem_chouchane:library", COLUMN => "author", FILTER => "RowFilter(>=, 'binary:a') AND RowFil
ter(<=, 'binary:n')"
ROW                             COLUMN+CELL
 jverne                         column=author:firstname, timestamp=1635938761462, value=Verne
 jverne                         column=author:lastname, timestamp=1635938718627, value=Jules
1 row(s)
Took 0.0182 seconds
```

- Scan des données de la colonne "firstname" appartenant à la column family "author" :

```
scan "ns_wiem_chouchane:library", COLUMN => "author:firstname"
```

```
hbase(main):023:0> scan "ns_wiem_chouchane:library", COLUMN => "author:firstname"
ROW                             COLUMN+CELL
 jverne                         column=author:firstname, timestamp=1635938761462, value=Verne
 vhugo                          column=author:firstname, timestamp=1635938399800, value=Victor
2 row(s)
Took 0.0148 seconds
```

- Scan des données dont la valeur de "title" correspond à la valeur paramétrée :

```
scan "ns_wiem_chouchane:library", COLUMN => "book:title", FILTER =>
"ValueFilter(=, 'binary:Face au drapeau')"
```

```
hbase(main):024:0> scan "ns_wiem_chouchane:library", COLUMN => "book:title", FILTER => "ValueFilter(=, 'binary:Face au d
rapeau')"
ROW                           COLUMN+CELL
 jverne                       column=book:title, timestamp=1635938884118, value=Face au drapeau
1 row(s)
Took 0.0215 seconds
```

- Scan des données (de version la plus récente) appartenant à la column family "book" dont la valeur de colonne "year" est inférieure ou égale à 1890 :

```
scan "ns_wiem_chouchane:library", {COLUMN => "book:year", FILTER =>
"ValueFilter(<=, 'binary:1890')", VERSIONS => 1}
```

```
hbase(main):036:0> scan "ns_wiem_chouchane:library", {COLUMN => "book:year", FILTER => "ValueFilter(<=, 'binary:1890')",
 VERSIONS => 1}
ROW                           COLUMN+CELL
 vhugo                        column=book:year, timestamp=1635938675684, value=1883
1 row(s)
Took 0.0066 seconds
```

- Scan des données des colonnes identifiées par une clé commençant par "jv" et correspondant à la regex "A-Z{2,}" :

```
scan "ns_wiem_chouchane:library", FILTER => "RowFilter(>=, 'binary:jv') AND
RowFilter(<, 'binary:jw') AND ValueFilter(=, 'regexstring:[A-Z]([a-z]+){2,}')"
```

```
hbase(main):051:0> scan "ns_wiem_chouchane:library", FILTER => "RowFilter(>=, 'binary:jv') AND RowFilter(<, 'binary:jw')
 AND ValueFilter(=, 'regexstring:[A-Z]([a-z]+){2,}')"
ROW                           COLUMN+CELL
 jverne                       column=author:firstname, timestamp=1635938761462, value=Verne
 jverne                       column=author:lastname, timestamp=1635938718627, value=Jules
 jverne                       column=book:publisher, timestamp=1635938798726, value=Hetzel
 jverne                       column=book:title, timestamp=1635938884118, value=Face au drapeau
1 row(s)
Took 0.0080 seconds
```

1.1.8 Mise à jours de valeurs

- Modification de la valeur de la colonne "lastname" appartenant à la column family "author" et identifiée par "vhugo" en "HAGO" :

```
put "ns_wiem_chouchane:library", "vhugo", "author:lastname", "HAGO"
```

- Modification de la valeur de la colonne "lastname" appartenant à la column family "author" et identifiée par "vhugo" en "HUGO" :

```
put "ns_wiem_chouchane:library", "vhugo", "author:lastname", "HUGO"
```

- Modification de la valeur de la colonne "firstname" appartenant à la column family "author" et identifiée par "vhugo" en "Victor Marie" :

```
put "ns_wiem_chouchane:library", "vhugo", "author:firstname", "Victor Marie"
```

- Modification de la valeur de la colonne "lastname" appartenant à la column family "author" et identifiée par "vhugo" en "Hugo" :

```
put "ns_wiem_chouchane:library", "vhugo", "author:lastname", "Hugo"
```

- Récupération des colonnes appartenant à la column family "author" et identifiées par "vhugo" :

```
get "ns_wiem_chouchane:library", "vhugo", "author"
```

```
hbase(main):056:0> get "ns_wiem_chouchane:library", "vhugo", "author"
COLUMN                          CELL
 author:firstname               timestamp=1636199194793, value=Victor Marie
 author:lastname                timestamp=1636199271648, value=Hugo
1 row(s)
Took 0.0364 seconds
```

- Récupération des colonnes appartenant à la column family "author" et identifiées par "vhugo" :

```
get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author"
```

```
hbase(main):057:0> get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author"
COLUMN                          CELL
 author:firstname               timestamp=1636199194793, value=Victor Marie
 author:lastname                timestamp=1636199271648, value=Hugo
1 row(s)
Took 0.0180 seconds
```

- Récupération des 10 dernières versions des colonnes appartenant à la column family "author" et identifiées par "vhugo" :

```
get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author", VERSIONS => 10
```

```
hbase(main):058:0> get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author", VERSIONS => 10
COLUMN                          CELL
 author:firstname               timestamp=1636199194793, value=Victor Marie
 author:firstname               timestamp=1635938399800, value=Victor
 author:lastname                timestamp=1636199271648, value=Hugo
 author:lastname                timestamp=1636199116726, value=HUGO
1 row(s)
Took 0.0119 seconds
```

1.1.9 Suppression de valeurs / colonnes

Le timestamp de "HUGO" dans le dernier get est 1636199116726.

- Suppression de la valeur author:name=HUGO correspondant au timestamp paramétré :

```
deleteall "ns_wiem_chouchane:library", "vhugo", "author:lastname", 1636199116726
```

```
hbase(main):062:0> deleteall "ns_wiem_chouchane:library", "vhugo", "author:lastname", 1636199116726
Took 0.0170 seconds
hbase(main):063:0> get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author", VERSIONS => 10
COLUMN                          CELL
 author:firstname               timestamp=1636199194793, value=Victor Marie
 author:firstname               timestamp=1635938399800, value=Victor
 author:lastname                timestamp=1636199271648, value=Hugo
1 row(s)
Took 0.0111 seconds
```

- Suppression de toutes les valeurs de la colonne "firstname" :

```
deleteall "ns_wiem_chouchane:library", "vhugo", "author:firstname"
```

```
hbase(main):064:0> deleteall "ns_wiem_chouchane:library", "vhugo", "author:firstname"
Took 0.0209 seconds
hbase(main):065:0> get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author", VERSIONS => 10
COLUMN                          CELL
 author:lastname                timestamp=1636199271648, value=Hugo
1 row(s)
Took 0.0080 seconds
```

- Suppression de tout le tuple identifié par "vhugo" :

```
deleteall "ns_wiem_chouchane:library", "vhugo"
```

```
hbase(main):066:0> get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author", VERSIONS => 10
COLUMN                          CELL
0 row(s)in):067:0> deleteall "ns_wiem_chouchane:library", "vhugo"
Took 0.0051 seconds
hbase(main):068:0> get "ns_wiem_chouchane:library", "vhugo", COLUMNS => "author", VERSIONS => 10
COLUMN                          CELL
0 row(s)
Took 0.0050 seconds
```

- Scan de la version 10 du tuple :

```
scan "ns_wiem_chouchane:library", COLUMNS => "author", VERSIONS => 10
```

```
hbase(main):072:0> scan "ns_wiem_chouchane:library", COLUMNS => "author", VERSIONS => 10
ROW                           COLUMN+CELL
 jverne                       column=author:firstname, timestamp=1635938761462, value=Verne
 jverne                       column=author:lastname, timestamp=1635938718627, value=Jules
1 row(s)
Took 0.0170 seconds
```

1.1.10 Suppression de table

- Désactivation de la table :

```
disable "ns_wiem_chouchane:library"
```

- Suppression de la table :

```
drop "ns_wiem_chouchane:library"
```

```
hbase(main):073:0> disable "ns_wiem_chouchane:library"
Took 1.4282 seconds
hbase(main):074:0> drop "ns_wiem_chouchane:library"
Took 0.4702 seconds
```

1.2.1 Insertion d'une table dans Hbase à partir d'un fichier CSV

- Import du fichier CSV dans HBase :

```
hdfs dfs -copyFromLocal ~/trees.csv /user/wiem.chouchane/trees.csv
```

- Vérification de la présence du fichier importé dans HBase :

```
hdfs dfs -cat /user/wiem.chouchane/trees.csv
```

```
[wiem.chouchane@hadoop-edge01 ~]$ hdfs dfs -copyFromLocal ~/trees.csv /user/wiem.chouchane/trees.csv
[wiem.chouchane@hadoop-edge01 ~]$ hdfs dfs -cat /user/wiem.chouchane/trees.csv
GEOPOINT;ARRONDISSEMENT;GENRE;ESPECE;FAMILLE;ANNEE PLANTATION;HAUTEUR;CIRCONFERENCE;ADRESSE;NOM COMMUN;VARIETE;OBJECTID;NOM_EV
(48.857140829, 2.29533455314);7;Maclura;pomifera;Moraceae;1935;13.0;;Quai Branly, avenue de La Motte-Piquet, avenue de la Bourdonnais, avenue de Suffren;Oran
er des Osages;;6;Parc du Champs de Mars
(48.8685686134, 2.31331809304);8;Calocedrus;decurrens;Cupressaceae;1854;20.0;195.0;Cours-la-Reine, avenue Franklin-D.-Roosevelt, avenue Matignon, avenue Gabr
el;Cèdre à encens;;11;Jardin des Champs Elysées
(48.8768191638, 2.33210374339);9;Pterocarya;fraxinifolia;Juglandaceae;1862;22.0;330.0;Place d'Estienne-d'Orves;Pérocarya du Caucase;;14;Square Etienne d'Orve
(48.8373323894, 2.40776275516);12;Celtis;australis;Cannabaceae;1906;16.0;295.0;27, boulevard Soult;Micocoulier de Provence;;16;Avenue 27 boulevard Soult
(48.8341842636, 2.46130493573);12;Quercus;petraea;Fagaceae;1784;30.0;430.0;route ronde des Minimes;Chêne rouvre;;19;Bois de Vincennes (lac des minimes)
(48.8325900983, 2.41116455985);12;Platanus;x acerifolia;Platanaceae;1860;45.0;405.0;Ile de Bercy;Platane commun;;21;Bois de Vincennes (Ile de Bercy)
(48.8226749117, 2.33869560229);14;Platanus;x acerifolia;Platanaceae;1840;40.0;580.0;Bd Jourdan, avenue Reille, rue Gazan, rue de la Cité,-Universitaire, rue
ansouty;Platane commun;;26;Parc Montsouris
(48.8428118006, 2.2972574926);15;Alnus;glutinosa;Betulaceae;1933;16.0;220.0;Rue Th.-ophraste-Renaudot, rue Léon-Lhermitte, rue Jean-Formig.., rue du Docteur-la
```

- Création de la table "trees" (préfixée du namespace précédent) dans HBase à partir du CSV :

```
import sys

ROW_KEY_COLUMN_NAME = 'objectid'
TABLE_NAME = 'ns_wiem_chouchane:trees'
GENDER_COLUMN_FAMILY = 'gender'
INFORMATION_COLUMN_FAMILY = 'information'
ADDRESS_COLUMN_FAMILY = 'address'


def main():
    columns = get_all_column_names()
    entry = sys.stdin
    command = get_table_creation_command()
    command = fill_command(command, entry, columns)
    sys.stdout.write(command)


def fill_command(command, entry, columns):
    row_key_column_index = columns.index(ROW_KEY_COLUMN_NAME)
    for line in entry:
        data = line.replace('\n', '').split(';')
        row_key = str(data[row_key_column_index])
        for index, element in enumerate(data):
            if index != row_key_column_index:
                command = add_new_command_line(command, index, row_key, element,
columns)
    return command


def get_all_column_names():
    columns = sys.stdin.readline().replace('\n', '').split(';')
    return list(map(lambda column: column.lower(), columns))


def get_table_creation_command():
    return 'create "' + TABLE_NAME + '" , ' \
          '{NAME => "' + GENDER_COLUMN_FAMILY + '", VERSIONS => 10},  ' \
          '{NAME => "' + INFORMATION_COLUMN_FAMILY + '", VERSIONS => 10}, ' \
          '{NAME => "' + ADDRESS_COLUMN_FAMILY + '", VERSIONS => 10};'


def add_new_command_line(command, index, row_key, element, columns):
    meta_data = get_meta_data(index, row_key, columns)
    command += get_new_command_line(meta_data, element)
    return command


def get_meta_data(index, row_key, columns):
    return {
        'row_key': row_key,
        'column_family': get_column_family(index),
        'column': get_column(index, columns)
    }
```

```
def get_new_command_line(meta_data, element):
    return 'put "' + \
           TABLE_NAME + '", "' + \
           meta_data['row_key'] + '", "' + \
           meta_data['column_family'] + ':' + \
           meta_data['column'] + '", "' + \
           element + '";'


def get_column_family(index):
    if is_gender_column_family_index(index):
        return GENDER_COLUMN_FAMILY
    elif is_information_column_family_index(index):
        return INFORMATION_COLUMN_FAMILY
    else:
        return ADDRESS_COLUMN_FAMILY


def is_gender_column_family_index(index):
    return 2 <= index <= 4 \
           or 9 <= index <= 10


def is_information_column_family_index(index):
    return 5 <= index <= 7


def get_column(index, columns):
    return columns[index]


if __name__ == "__main__":
    main()
```

- Lancement de la commande de création de la table :

```
hdfs dfs -cat /user/wiem.chouchane/trees.csv | python app.py | hbase shell
```

- Description de la table créée :

```
describe "ns_wiem_chouchane:trees"
```

```
hbase(main):001:0> describe "ns_wiem_chouchane:trees"
Table ns_wiem_chouchane:trees is ENABLED
ns_wiem_chouchane:trees
COLUMN FAMILIES DESCRIPTION
{NAME => 'address', VERSIONS => '10', EVICT_BLOCKS_ON_CLOSE => 'false', NEW_VERSION_BEHAVIOR => 'false', KEEP_DELETED_CELLS => 'FALSE', CACHE_DATA_ON_WRITE =>
 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', MIN_VERSIONS => '0', REPLICATION_SCOPE => '0', BLOOMFILTER => 'ROW', CACHE_INDEX_ON_WRITE => 'false
', IN_MEMORY => 'false', CACHE_BLOOMS_ON_WRITE => 'false', PREFETCH_BLOCKS_ON_OPEN => 'false', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '6553
6'}

{NAME => 'gender', VERSIONS => '10', EVICT_BLOCKS_ON_CLOSE => 'false', NEW_VERSION_BEHAVIOR => 'false', KEEP_DELETED_CELLS => 'FALSE', CACHE_DATA_ON_WRITE =>
'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', MIN_VERSIONS => '0', REPLICATION_SCOPE => '0', BLOOMFILTER => 'ROW', CACHE_INDEX_ON_WRITE => 'false'
, IN_MEMORY => 'false', CACHE_BLOOMS_ON_WRITE => 'false', PREFETCH_BLOCKS_ON_OPEN => 'false', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '65536
'}

{NAME => 'information', VERSIONS => '10', EVICT_BLOCKS_ON_CLOSE => 'false', NEW_VERSION_BEHAVIOR => 'false', KEEP_DELETED_CELLS => 'FALSE', CACHE_DATA_ON_WRIT
E => 'false', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', MIN_VERSIONS => '0', REPLICATION_SCOPE => '0', BLOOMFILTER => 'ROW', CACHE_INDEX_ON_WRITE => 'f
alse', IN_MEMORY => 'false', CACHE_BLOOMS_ON_WRITE => 'false', PREFETCH_BLOCKS_ON_OPEN => 'false', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '
65536'}

3 row(s)

QUOTAS
0 row(s)
Took 0.4983 seconds
```

- Comptage des tuples créés :

```
count "ns_wiem_chouchane:trees"
```

```
hbase(main):002:0> count "ns_wiem_chouchane:trees"
97 row(s)
Took 0.1090 seconds
=> 97
```

Idem ici, il n'y a pas énormément de tuples donc il n'y a pas besoin de configurer le cache. On a bien 97 tuples correspondant aux 97 lignes insérées depuis le CSV, en effet chacune avait un "OBJECTID" différent comme nous avons pu le constater sur l'exploration ci-dessous :

```
[6]  import pandas as pd

     df = pd.read_csv('trees.csv', sep=';')
     df.shape

     (97, 13)
```

```
[9]  len(df['OBJECTID'].unique())

     97
```

- Pour l'exemple, récupération des valeurs de toutes les colonnes identifiées par la clé "6" :

```
get "ns_wiem_chouchane:trees", "6"
```

```
hbase(main):003:0> get "ns_wiem_chouchane:trees", "6"
COLUMN                                  CELL
 address:adresse                        timestamp=1636220250112, value=Quai Branly, avenue de La Motte-Piquet, avenue de la Bourdonnais, avenue de Suffren
 address:arrondissement                 timestamp=1636220250053, value=7
 address:geopoint                       timestamp=1636220250040, value=(48.857140829, 2.29533455314)
 address:nom_ev                         timestamp=1636220250134, value=Parc du Champs de Mars
 gender:espece                          timestamp=1636220250070, value=pomifera
 gender:famille                         timestamp=1636220250077, value=Moraceae
 gender:genre                           timestamp=1636220250061, value=Maclura
 gender:nom commun                      timestamp=1636220250119, value=Oranger des Osages
 gender:variete                         timestamp=1636220250126, value=
 information:annee plantation           timestamp=1636220250089, value=1935
 information:circonference              timestamp=1636220250105, value=
 information:hauteur                    timestamp=1636220250097, value=13.0
1 row(s)
Took 0.0591 seconds
```