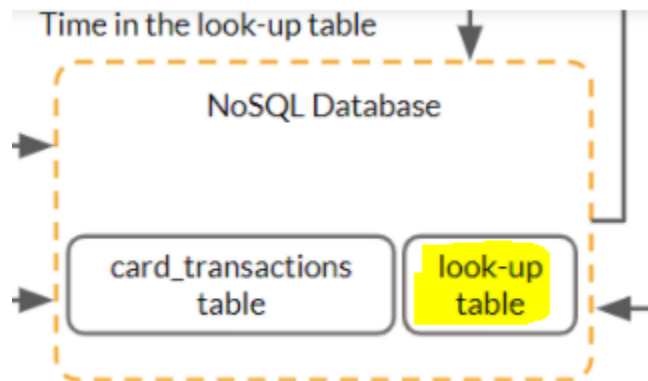


Creating Lookup Table

Command to create the Lookup Table



Used Python file for this step and created a dataframe which acts as a source file to Hbase and created a lookup table in NoSQL and then loaded the data also.

Dataframe Name: lookup_table

Hbase Table: lookup_table

#import happybase and connect to localhost

```
import happybase
```

```
connection = happybase.Connection('localhost', port=9090 ,autoconnect=False)
```

#function definition to open connection

```
def open_connection():
```

```
    connection.open()
```

#function definition to close connection

```
def close_connection():
```

```
    connection.close()
```

#function definition to list the tables

```
def list_tables():
```

```
    print "fetching all table"
```

```
    open_connection()
```

```
    tables = connection.tables()
```

```
    close_connection()
```

```
    print "all tables fetched"
```

```
    return tables
```

#check if table already exist and create the table.

```
def create_table(name,cf):
    print "creating table " + name
    tables = list_tables()
    if name not in tables:
        open_connection()
        connection.create_table(name, cf)
        close_connection()
        print "table created"
    else:
        print "table already present"
```

While loading DF into table, this function gets the create table name

```
def get_table(name):
    open_connection()
    table = connection.table(name)
    close_connection()
    return table
```

#Actual command to create lookup table

```
create_table('lookup_table', {'info' : dict(max_versions=5) })
```

#Function to load data from DF

```
def batch_insert_data(df,tableName):
    print "starting batch insert of events"
    table = get_table(tableName)
    open_connection()
    rows_count=0
    rowKey_dict={}
    with table.batch(batch_size=4) as b:
        for row in df.rdd.collect():
            b.put(bytes(row.card_id) , { 'info:card_id':bytes(row.card_id),
            'info:transaction_date':bytes(row.transaction_date),
            'info:score':bytes(row.score),
            'info:postcode':bytes(row.postcode),
            'info:UCL':bytes(row.UCL)})
```

```
print "batch insert done"  
close_connection()
```

#Call function to insert data into Hbase table

```
batch_insert_data(lookup_table,'lookup_table')
```

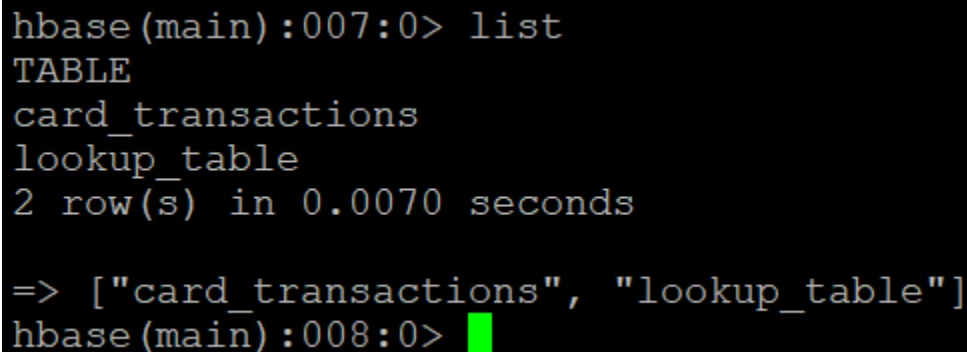
Command to see the table created

- 1) Login to putty as root user
- 2) Start thrift server using below command
/opt/cloudera/parcels/CDH/lib/hbase/bin/hbase-daemon.sh start thrift -p 9090
- 3) Give command hbase shell
- 4) Give command "list"

```
In [76]: ► create_table('lookup_table', {'info' : dict(max_versions=5) })  
  
creating table lookup_table  
fetching all table  
all tables fetched  
table created
```

```
In [79]: ► batch_insert_data(lookup_table,'lookup_table')  
  
starting batch insert of events  
batch insert done
```

Screenshot of the created table



```
hbase(main):007:0> list  
TABLE  
card_transactions  
lookup_table  
2 row(s) in 0.0070 seconds  
  
=> ["card_transactions", "lookup_table"]  
hbase(main):008:0> █
```

root@ip-10-0-0-149:~

```

6591175617713393      column=info:transaction_date, timestamp=1634375007372, value=2018-01-31 13:10:37
6592184145413632      column=info:UCL, timestamp=1634375006787, value=13734342.65
6592184145413632      column=info:card_id, timestamp=1634375006787, value=6592184145413632
6592184145413632      column=info:postcode, timestamp=1634375006787, value=53186
6592184145413632      column=info:score, timestamp=1634375006787, value=456
6592184145413632      column=info:transaction_date, timestamp=1634375006787, value=2018-01-28 00:54:30
6594248319343442      column=info:UCL, timestamp=1634375006872, value=15065362.77
6594248319343442      column=info:card_id, timestamp=1634375006872, value=6594248319343442
6594248319343442      column=info:postcode, timestamp=1634375006872, value=24927
6594248319343442      column=info:score, timestamp=1634375006872, value=350
6594248319343442      column=info:transaction_date, timestamp=1634375006872, value=2018-01-31 23:42:38
6595638658736751      column=info:UCL, timestamp=1634375007621, value=14005069.97
6595638658736751      column=info:card_id, timestamp=1634375007621, value=6595638658736751
6595638658736751      column=info:postcode, timestamp=1634375007621, value=68328
6595638658736751      column=info:score, timestamp=1634375007621, value=310
6595638658736751      column=info:transaction_date, timestamp=1634375007621, value=2018-01-30 10:50:34
6595814135833988      column=info:UCL, timestamp=1634375007288, value=14332708.84
6595814135833988      column=info:card_id, timestamp=1634375007288, value=6595814135833988
6595814135833988      column=info:postcode, timestamp=1634375007288, value=22508
6595814135833988      column=info:score, timestamp=1634375007288, value=210
6595814135833988      column=info:transaction_date, timestamp=1634375007288, value=2018-01-30 02:03:54
6595928469079750      column=info:UCL, timestamp=1634375008323, value=11824730.01
6595928469079750      column=info:card_id, timestamp=1634375008323, value=6595928469079750
6595928469079750      column=info:postcode, timestamp=1634375008323, value=98349
6595928469079750      column=info:score, timestamp=1634375008323, value=412
6595928469079750      column=info:transaction_date, timestamp=1634375008323, value=2018-01-24 12:38:22
6597703848279563      column=info:UCL, timestamp=1634375007681, value=15250624.49
6597703848279563      column=info:card_id, timestamp=1634375007681, value=6597703848279563
6597703848279563      column=info:postcode, timestamp=1634375007681, value=95699
6597703848279563      column=info:score, timestamp=1634375007681, value=218
6597703848279563      column=info:transaction_date, timestamp=1634375007681, value=2018-01-27 10:51:49
6598830758632447      column=info:UCL, timestamp=1634375007878, value=12685782.48
6598830758632447      column=info:card_id, timestamp=1634375007878, value=6598830758632447
6598830758632447      column=info:postcode, timestamp=1634375007878, value=19421
6598830758632447      column=info:score, timestamp=1634375007878, value=293
6598830758632447      column=info:transaction_date, timestamp=1634375007878, value=2018-01-30 00:18:34
6599900931314251      column=info:UCL, timestamp=1634375008288, value=12487392.07
6599900931314251      column=info:card_id, timestamp=1634375008288, value=6599900931314251
6599900931314251      column=info:postcode, timestamp=1634375008288, value=97423
6599900931314251      column=info:score, timestamp=1634375008288, value=297
6599900931314251      column=info:transaction_date, timestamp=1634375008288, value=2018-01-31 11:25:16
999 row(s) in 0.4250 seconds

hbase(main):009:0>

```