

Creating Lookup Table

Command to create the Lookup Table

Dataframe Name: look_up_table

Hbase Table: look_up_table

```
import happybase
```

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

```
def open_connection():
```

```
    connection.open()
```

```
def close_connection():
```

```
    connection.close()
```

```
def list_tables():
```

```
    print "fetching all tables"
```

```
    open_connection()
```

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

```
    close_connection()
```

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

```
    return tables
```

To 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"
```

To get the created table

```
def get_table(name):
```

```
    open_connection()
```

```
    table = connection.table(name)
```

```
    close_connection()
```

```
    return table
```

#To create lookup table

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

#To load data from the dataframe

```
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()
```

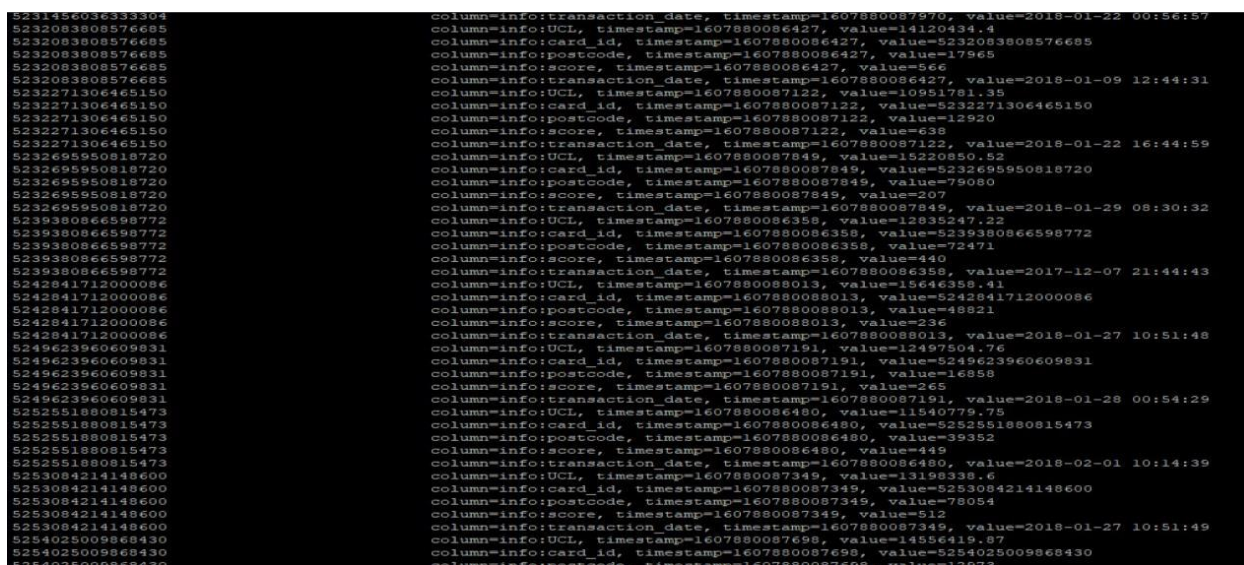
#To insert data into Hbase table

```
batch_insert_data(look_up_table,'look_up_table')
```

<Command to see the table created>

- *list* command is used to view the created table
- *scan 'look_up_table'* command to view data inserted into table.

Screenshot of the created table



```
5231456036333304 column=info:transaction_date, timestamp=1607880087970, value=2018-01-22 00:56:57
5232083808576685 column=info:UCL, timestamp=1607880086427, value=14120434.4
5232083808576685 column=info:card_id, timestamp=1607880086427, value=5232083808576685
5232083808576685 column=info:postcode, timestamp=1607880086427, value=17965
5232083808576685 column=info:score, timestamp=1607880086427, value=566
5232083808576685 column=info:transaction_date, timestamp=1607880086427, value=2018-01-09 12:44:31
5232271306465150 column=info:UCL, timestamp=1607880087122, value=10951751.35
5232271306465150 column=info:card_id, timestamp=1607880087122, value=5232271306465150
5232271306465150 column=info:postcode, timestamp=1607880087122, value=12920
5232271306465150 column=info:score, timestamp=1607880087122, value=638
5232271306465150 column=info:transaction_date, timestamp=1607880087122, value=2018-01-22 16:44:59
5232695950818720 column=info:UCL, timestamp=1607880087849, value=15220850.52
5232695950818720 column=info:card_id, timestamp=1607880087849, value=5232695950818720
5232695950818720 column=info:postcode, timestamp=1607880087849, value=79080
5232695950818720 column=info:score, timestamp=1607880087849, value=207
5232695950818720 column=info:transaction_date, timestamp=1607880087849, value=2018-01-29 08:30:32
5239380866598772 column=info:UCL, timestamp=1607880086358, value=12935247.22
5239380866598772 column=info:card_id, timestamp=1607880086358, value=5239380866598772
5239380866598772 column=info:postcode, timestamp=1607880086358, value=72471
5239380866598772 column=info:score, timestamp=1607880086358, value=440
5239380866598772 column=info:transaction_date, timestamp=1607880086358, value=2017-12-07 21:44:43
5242841712000086 column=info:UCL, timestamp=1607880088013, value=15646358.41
5242841712000086 column=info:card_id, timestamp=1607880088013, value=5242841712000086
5242841712000086 column=info:postcode, timestamp=1607880088013, value=48821
5242841712000086 column=info:score, timestamp=1607880088013, value=236
5242841712000086 column=info:transaction_date, timestamp=1607880088013, value=2018-01-27 10:51:48
5249623960609831 column=info:UCL, timestamp=1607880087191, value=12497804.76
5249623960609831 column=info:card_id, timestamp=1607880087191, value=5249623960609831
5249623960609831 column=info:postcode, timestamp=1607880087191, value=16858
5249623960609831 column=info:score, timestamp=1607880087191, value=265
5249623960609831 column=info:transaction_date, timestamp=1607880087191, value=2018-01-28 00:54:29
5252551880815473 column=info:UCL, timestamp=1607880086480, value=11540779.75
5252551880815473 column=info:card_id, timestamp=1607880086480, value=5252551880815473
5252551880815473 column=info:postcode, timestamp=1607880086480, value=39352
5252551880815473 column=info:score, timestamp=1607880086480, value=449
5252551880815473 column=info:transaction_date, timestamp=1607880086480, value=2018-02-01 10:14:39
5253084214148600 column=info:UCL, timestamp=1607880087349, value=13198338.6
5253084214148600 column=info:card_id, timestamp=1607880087349, value=5253084214148600
5253084214148600 column=info:postcode, timestamp=1607880087349, value=78054
5253084214148600 column=info:score, timestamp=1607880087349, value=512
5253084214148600 column=info:transaction_date, timestamp=1607880087349, value=2018-01-27 10:51:49
5254025009868430 column=info:UCL, timestamp=1607880087698, value=14556419.87
5254025009868430 column=info:card_id, timestamp=1607880087698, value=5254025009868430
5254025009868430 column=info:postcode, timestamp=1607880087698, value=12973
```