TASK 3

A PDF document **(CreateNoSQL.pdf)** to create a look-up table with columns specified earlier in the problem statement.

Creating Lookup Table

Command to create the Lookup Table

```
import happybase
import pandas as pd

# Initialize HBase connection
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")
```

© Copyright 2020. upGrad Education Pvt. Ltd. All rights reserved

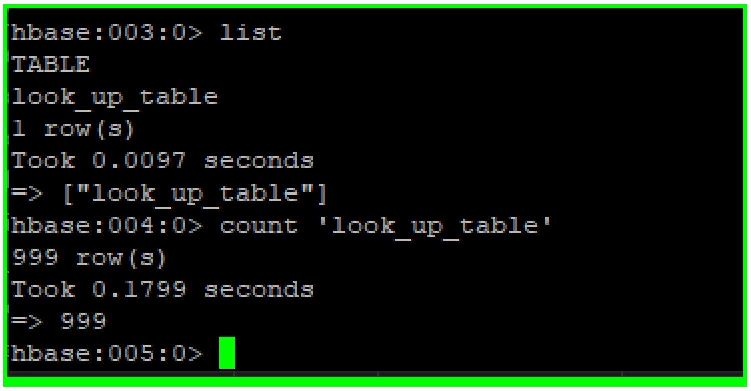
```
    open_connection()
    tables = connection.tables()
    close_connection()
    print("All tables fetched")
    return tables

def create_table(name, cf):
    print("Creating table " + name)
    tables = list_tables()
    if name.encode('utf-8') not in tables:
        open_connection()
        connection.create_table(name, cf)
        close_connection()
        print("Table created")
    else:
        print("Table already present")

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

# Create the lookup table
create_table('look_up_table', {'info': dict(max_versions=5)})

# Load data from the CSV file into a DataFrame
def load_csv_to_dataframe(file_path):
    print(f"Loading data from {file_path}")
    df = pd.read_csv(file_path)
    return df
```

© Copyright 2020. upGrad Education Pvt. Ltd. All rights reserved

```
# To batch insert data from the DataFrame into HBase
def batch_insert_data(df, tableName):
    print("Starting batch insert of events")
    table = get_table(tableName)

    open_connection()
    with table.batch(batch_size=4) as b:
        for index, row in df.iterrows():
            b.put(
                bytes(str(row['card_id']), 'utf-8'), {
                    b'info:card_id': bytes(str(row['card_id']), 'utf-8'),
                    b'info:transaction_date': bytes(str(row['transaction_date']), 'utf-8'),
                    b'info:score': bytes(str(row['score']), 'utf-8'),
                    b'info:postcode': bytes(str(row['postcode']), 'utf-8'),
                    b'info:UCL': bytes(str(row['UCL']), 'utf-8')
                }
            )
    print("Batch insert done")
    close_connection()

# Path to the CSV file
csv_file_path = '/home/hadoop/look_up_table.csv'

# Load data and insert into HBase
df = load_csv_to_dataframe(csv_file_path)
batch_insert_data(df, 'look_up_table')
```

© Copyright 2020. upGrad Education Pvt. Ltd. All rights reserved

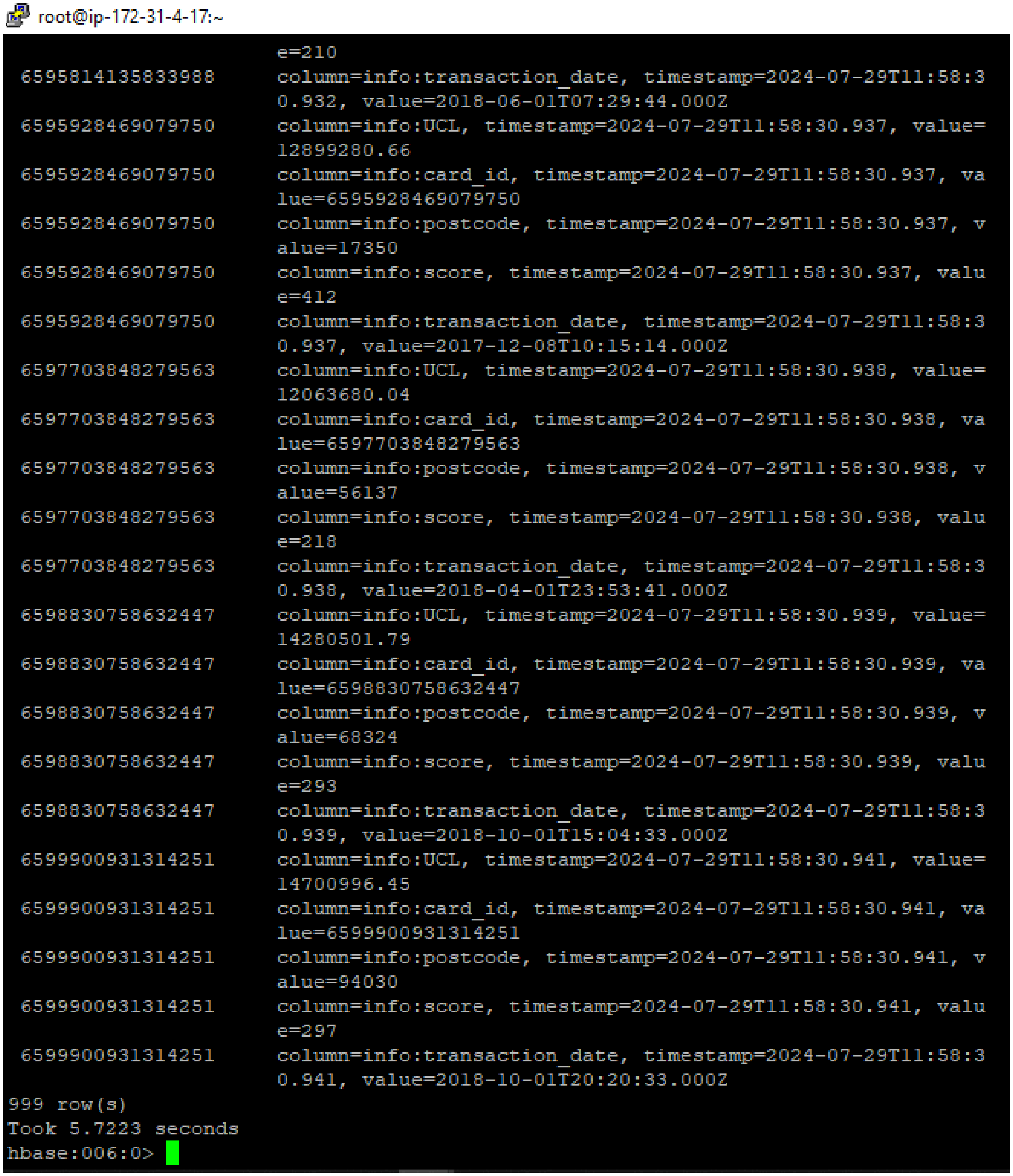
Command to see the table created :list



```
hbase:003:0> list
TABLE
look_up_table
1 row(s)
Took 0.0097 seconds
=> ["look_up_table"]
hbase:004:0> count 'look_up_table'
999 row(s)
Took 0.1799 seconds
=> 999
hbase:005:0>
```

Screenshot of the created table

© Copyright 2020. upGrad Education Pvt. Ltd. All rights reserved



```
                               e=210
 6595814135833988              column=info:transaction_date, timestamp=2024-07-29T11:58:3
                               0.932, value=2018-06-01T07:29:44.000Z
 6595928469079750              column=info:UCL, timestamp=2024-07-29T11:58:30.937, value=
                               12899280.66
 6595928469079750              column=info:card_id, timestamp=2024-07-29T11:58:30.937, va
                               lue=6595928469079750
 6595928469079750              column=info:postcode, timestamp=2024-07-29T11:58:30.937, v
                               alue=17350
 6595928469079750              column=info:score, timestamp=2024-07-29T11:58:30.937, valu
                               e=412
 6595928469079750              column=info:transaction_date, timestamp=2024-07-29T11:58:3
                               0.937, value=2017-12-08T10:15:14.000Z
 6597703848279563              column=info:UCL, timestamp=2024-07-29T11:58:30.938, value=
                               12063680.04
 6597703848279563              column=info:card_id, timestamp=2024-07-29T11:58:30.938, va
                               lue=6597703848279563
 6597703848279563              column=info:postcode, timestamp=2024-07-29T11:58:30.938, v
                               alue=56137
 6597703848279563              column=info:score, timestamp=2024-07-29T11:58:30.938, valu
                               e=218
 6597703848279563              column=info:transaction_date, timestamp=2024-07-29T11:58:3
                               0.938, value=2018-04-01T23:53:41.000Z
 6598830758632447              column=info:UCL, timestamp=2024-07-29T11:58:30.939, value=
                               14280501.79
 6598830758632447              column=info:card_id, timestamp=2024-07-29T11:58:30.939, va
                               lue=6598830758632447
 6598830758632447              column=info:postcode, timestamp=2024-07-29T11:58:30.939, v
                               alue=68324
 6598830758632447              column=info:score, timestamp=2024-07-29T11:58:30.939, valu
                               e=293
 6598830758632447              column=info:transaction_date, timestamp=2024-07-29T11:58:3
                               0.939, value=2018-10-01T15:04:33.000Z
 6599900931314251              column=info:UCL, timestamp=2024-07-29T11:58:30.941, value=
                               14700996.45
 6599900931314251              column=info:card_id, timestamp=2024-07-29T11:58:30.941, va
                               lue=6599900931314251
 6599900931314251              column=info:postcode, timestamp=2024-07-29T11:58:30.941, v
                               alue=94030
 6599900931314251              column=info:score, timestamp=2024-07-29T11:58:30.941, valu
                               e=297
 6599900931314251              column=info:transaction_date, timestamp=2024-07-29T11:58:3
                               0.941, value=2018-10-01T20:20:33.000Z
999 row(s)
Took 5.7223 seconds
hbase:006:0>
```

© Copyright 2020. upGrad Education Pvt. Ltd. All rights reserved