

Loading Historical Transactions Data into NoSQL Database

Commands to load the past transactions data into NoSQL database

```
import happybase

# To create connection
connection = happybase.Connection('localhost', port=9090, autoconnect=False)

# To open connection and perform operations
def open_connection():
    connection.open()

# To close the opened connection
def close_connection():
    connection.close()

# To list all tables in HBase
def list_tables():
    print("Fetching all tables...")
    open_connection()
    tables = connection.tables()
    close_connection()
    print("All tables fetched.")
    return tables

# To create the required table
def create_table(name, cf):
    print("Creating table " + name)
    tables = list_tables()
    if name.encode() not in tables: # Ensure consistent data type comparison
        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 batch insert data
def batch_insert_data(filename, tableName):
    print("Starting batch insert of events...")
    file = open(filename, "r")
    table = get_table(tableName)
    open_connection()
    i = 0
    for line in file:
        temp = line.strip().split(",")
        # To skip the first row
        if temp[0] != 'card_id':
            table.put(
                str(i).encode(), # Convert to bytes for consistency
                {
                    'info:card_id': temp[0].encode(),
                    'info:member_id': temp[1].encode(),
                    'info:amount': temp[2].encode(),
                    'info:postcode': temp[3].encode(),
                    'info:pos_id': temp[4].encode(),
                    'info:transaction_dt': temp[5].encode(),
                    'info:status': temp[6].encode()
                }
            )
            i += 1
    file.close()
    print("Batch insert done.")
    close_connection()

# To batch insert data of card_transactions.csv
create_table('card_transactions', {'info': dict(max_versions=5)})
batch_insert_data('card_transactions.csv', 'card_transactions')
```

<Command to list the table in which the data is loaded and the command to get the count of the rows of the table>

Command used to view the created table: *list*

Command used to get the count of the rows in table: *count 'card_transactions'*

Screenshot of the table created

```
HBase Shell
Use "help" to get list of supported commands.
Use "exit" to quit this interactive shell.
Version 1.4.13, rUnknown, Fri Apr 17 15:18:24 UTC 2020
```

```
count 'card_transactions'
Current count: 1000, row: 10897
Current count: 2000, row: 11797
Current count: 3000, row: 12697
Current count: 4000, row: 13597
Current count: 5000, row: 14497
Current count: 6000, row: 15397
Current count: 7000, row: 16297
Current count: 8000, row: 17197
Current count: 9000, row: 18097
Current count: 10000, row: 18998
Current count: 11000, row: 19898
Current count: 12000, row: 20797
Current count: 13000, row: 21697
Current count: 14000, row: 22597
Current count: 15000, row: 23497
Current count: 16000, row: 24397
Current count: 17000, row: 25297
Current count: 18000, row: 26197
Current count: 19000, row: 27097
Current count: 20000, row: 27998
Current count: 21000, row: 28898
Current count: 22000, row: 29798
Current count: 23000, row: 30697
Current count: 24000, row: 31597
Current count: 25000, row: 32497
Current count: 26000, row: 33397
Current count: 27000, row: 34297
Current count: 28000, row: 35197
Current count: 29000, row: 36097
Current count: 30000, row: 36998
Current count: 31000, row: 37898
Current count: 32000, row: 38798
Current count: 33000, row: 39698
Current count: 34000, row: 40597
Current count: 35000, row: 41497
Current count: 36000, row: 42397
Current count: 37000, row: 43297
Current count: 38000, row: 44197
Current count: 39000, row: 45097
Current count: 40000, row: 45998
Current count: 41000, row: 46898
Current count: 42000, row: 47798
Current count: 43000, row: 48698
Current count: 44000, row: 49598
Current count: 45000, row: 50497
Current count: 46000, row: 51397
Current count: 47000, row: 52297
Current count: 48000, row: 53197
Current count: 49000, row: 6134
Current count: 50000, row: 7034
Current count: 51000, row: 7935
Current count: 52000, row: 8835
Current count: 53000, row: 9735
53292 row(s) in 3.9470 seconds
```

```
53292
[hadoop@ip-172-31-48-237 ~]$ |
```