

Task 2: Ingest the relevant data (card_member and member_score) from AWS RDS to Hadoop. {this task is done and given to you in form of CSV files named "card_member" and "member_score"}.

Task 2: Ingest the relevant data (card_member and member_score) from AWS RDS to Hadoop.

This task is done and given to you in form of CSV files named "card_member" and "member_score".

Task 3: Create a look-up table in MongoDB {NoSQL database} with columns specified earlier in the problem statement.

```
In [2]: ▶ df_cardMember = pd.read_csv('card_member.csv')
df_member_score = pd.read_csv('member_score.csv')
df_card_transactions = pd.read_csv('card_transactions.csv')
```

```
In [3]: ▶ print("Number of Records in df_cardMember : ", df_cardMember.shape[0])
print("Number of Records in df_member_score is :", df_member_score.shape[0])
print("Number of Records in df_card_transactions is :", df_card_transactions.shape[0])
```

```
Number of Records in df_cardMember : 999
Number of Records in df_member_score is : 999
Number of Records in df_card_transactions is : 53292
```

Task 3: Create a look-up table in MongoDB {NoSQL database} with columns specified earlier in the problem statement.

Pre-Analysis is done with Pandas and lookupTable is created with required columns before loading into Mongo.

```
In [23]: final_lookup_df = lookup_df[['member_id', 'score', 'ucl', 'card_id', 'postcode', 'transaction_dt2']]
         final_lookup_df

Out[23]:
```

	member_id	score	ucl	card_id	postcode	transaction_dt2
0	37495066290	339	1.301153e+07	348702330256514	33946	2018-11-02 00:00:00
1	117826301530	289	1.230049e+07	5189563368503974	30234	2018-06-01 09:15:04
2	1147922084344	393	1.189271e+07	5407073344486464	63770	2018-01-18 10:55:30
3	1314074991813	225	1.279424e+07	378303738095292	41778	2018-06-01 14:31:05
4	1739553947511	642	1.504144e+07	348413196172048	92258	2018-12-01 21:29:58
...
994	992104321998571	658	1.349084e+07	5115273575078211	42082	2018-08-01 15:40:29
995	992552823055811	694	1.512892e+07	374878208630443	12747	2018-04-01 22:08:21
996	994983851226493	687	1.363332e+07	5399715069800093	77432	2018-01-30 09:05:15
997	996411635289270	335	1.513406e+07	4981548424137027	92694	2017-12-18 00:01:04
998	997128952368160	683	1.425606e+07	5391723993945313	42323	2018-08-01 12:09:49

999 rows × 6 columns

```
In [24]: #determining the name of the file
         file_name = 'lookupTable4.csv'

         #saving the excel
         final_lookup_df.to_csv(file_name)
         print('DataFrame lookup_df - is written to Excel File successfully.')
```

DataFrame lookup_df - is written to Excel File successfully.

```
C:\Program Files\MongoDB\Server\5.0\bin>mongoimport --db CREDIT_CARD_DB --collection lookup_table --type csv --file "C:\Users\Jagad\MS Program\Assignments\CapstoreProject_1stYearProject\CreditCardFr
aud_Template\lookupTable4.csv" --headerline
2022-12-03T23:08:31.680-0500    connected to: mongodb://localhost/
2022-12-03T23:08:31.737-0500    999 document(s) imported successfully. 0 document(s) failed to import.

C:\Program Files\MongoDB\Server\5.0\bin>
```

Task 4: After creating the table, you need to load the relevant data in the lookup table.

```
> db.lookup_table.count()
999
>
```

```
> show collections
card_transactions
lookup_table
> db.lookup_table.find()
{ "_id" : ObjectId("638c1d3f9b314ecef494171a"), "" : 0, "member_id" : NumberLong("37495066290"), "score" : 339, "ucl" : 13011526.339291777, "card_id" : NumberLong("348702330256514"), "postcode" : 33946, "transaction_dt2" : "2018-11-02 00:00:00" }
{ "_id" : ObjectId("638c1d3f9b314ecef494171b"), "" : 3, "member_id" : NumberLong("1314074991813"), "score" : 225, "ucl" : 12794239.889645452, "card_id" : NumberLong("378303738095292"), "postcode" : 41778, "transaction_dt2" : "2018-06-01 14:31:05" }
{ "_id" : ObjectId("638c1d3f9b314ecef494171c"), "" : 2, "member_id" : NumberLong("1147922084344"), "score" : 393, "ucl" : 11892709.546673544, "card_id" : NumberLong("5407073344486464"), "postcode" : 63770, "transaction_dt2" : "2018-01-18 10:55:30" }
{ "_id" : ObjectId("638c1d3f9b314ecef494171d"), "" : 4, "member_id" : NumberLong("1739553947511"), "score" : 642, "ucl" : 15041443.076209597, "card_id" : NumberLong("348413196172048"), "postcode" : 92258, "transaction_dt2" : "2018-12-01 21:29:58" }
{ "_id" : ObjectId("638c1d3f9b314ecef494171e"), "" : 5, "member_id" : NumberLong("3761426295463"), "score" : 413, "ucl" : 13928702.895311527, "card_id" : NumberLong("348536585266345"), "postcode" : 38863, "transaction_dt2" : "2018-11-01 17:24:36" }
{ "_id" : ObjectId("638c1d3f9b314ecef494171f"), "" : 6, "member_id" : NumberLong("4494068832701"), "score" : 217, "ucl" : 10937142.461807987, "card_id" : NumberLong("5515987071565183"), "postcode" : 30668, "transaction_dt2" : "2017-10-21 15:37:42" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941720"), "" : 7, "member_id" : NumberLong("6836124210484"), "score" : 504, "ucl" : 10009917.54804042, "card_id" : NumberLong("5400251558458125"), "postcode" : 17749, "transaction_dt2" : "2018-08-01 15:40:29" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941721"), "" : 8, "member_id" : NumberLong("6991872634058"), "score" : 697, "ucl" : 12461184.53195883, "card_id" : NumberLong("4573337022888445"), "postcode" : 14126, "transaction_dt2" : "2018-01-26 20:49:08" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941722"), "" : 10, "member_id" : NumberLong("8732267588672"), "score" : 213, "ucl" : 13091644.454693366, "card_id" : NumberLong("5342400571435088"), "postcode" : 13155, "transaction_dt2" : "2018-10-01 01:01:51" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941723"), "" : 9, "member_id" : NumberLong("7955566230397"), "score" : 372, "ucl" : 12875679.909525491, "card_id" : NumberLong("4708912758619517"), "postcode" : 31650, "transaction_dt2" : "2018-11-01 00:00:00" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941724"), "" : 1, "member_id" : NumberLong("117826301530"), "score" : 289, "ucl" : 12300487.018047417, "card_id" : NumberLong("5189563368503974"), "postcode" : 30234, "transaction_dt2" : "2018-06-01 09:15:04" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941725"), "" : 13, "member_id" : NumberLong("9190444424572"), "score" : 559, "ucl" : 13042904.354232315, "card_id" : NumberLong("341363858179050"), "postcode" : 43967, "transaction_dt2" : "2018-01-21 23:11:37" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941726"), "" : 14, "member_id" : NumberLong("9250698176266"), "score" : 233, "ucl" : 16765515.132200925, "card_id" : NumberLong("340028465709212"), "postcode" : 24658, "transaction_dt2" : "2018-02-01 03:25:35" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941727"), "" : 15, "member_id" : NumberLong("9873334520465"), "score" : 298, "ucl" : 14507890.127987808, "card_id" : NumberLong("5495445301620991"), "postcode" : 93543, "transaction_dt2" : "2018-06-01 12:45:45" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941728"), "" : 17, "member_id" : NumberLong("11877954983420"), "score" : 497, "ucl" : 12542974.715323305, "card_id" : NumberLong("5164771396791995"), "postcode" : 61375, "transaction_dt2" : "2018-10-01 04:32:32" }
{ "_id" : ObjectId("638c1d3f9b314ecef4941729"), "" : 16, "member_id" : NumberLong("11716573646690"), "score" : 249, "ucl" : 12755932.16131873, "card_id" : NumberLong("4795844193055110"), "postcode" : 16874, "transaction_dt2" : "2018-09-01 00:27:10" }
{ "_id" : ObjectId("638c1d3f9b314ecef494172a"), "" : 18, "member_id" : NumberLong("12390918683920"), "score" : 407, "ucl" : 14644725.444534216, "card_id" : NumberLong("5423921058459194"), "postcode" : 47465, "transaction_dt2" : "2018-11-01 03:21:54" }
{ "_id" : ObjectId("638c1d3f9b314ecef494172b"), "" : 19, "member_id" : NumberLong("12731668664932"), "score" : 612, "ucl" : 14430228.668971352, "card_id" : NumberLong("5379610024035907"), "postcode" : 33827, "transaction_dt2" : "2018-01-19 06:14:11" }
{ "_id" : ObjectId("638c1d3f9b314ecef494172c"), "" : 21, "member_id" : NumberLong("15235737327750"), "score" : 244, "ucl" : 15159145.91846099, "card_id" : NumberLong("4877961603808028"), "postcode" : 49637, "transaction_dt2" : "2018-07-01 01:03:05" }
{ "_id" : ObjectId("638c1d3f9b314ecef494172d"), "" : 20, "member_id" : NumberLong("12871894714576"), "score" : 689, "ucl" : 14362904.575688194, "card_id" : NumberLong("4380357644954006"), "postcode" : 98953, "transaction_dt2" : "2018-11-01 15:39:15" }
Type "it" for more
>
```