Project 2

Intermediate report

Submitted by,

1. Kaushik Pulivendula Balasubramanyam 1001529713
2. Subrahmanya Basavapatna Nagaraja Rao 1001540208

**PseudoCode of converting csv files into JSON, Inserting them into MongoDB and then creating two documents.**

import csv

Import json

From pymongo, import MongoClient

Import system, getoutput, pprint

// Database connection

Connection = MongoClient(‘local host’, 27017)

Print db connected success message

// Create JSON file from given CSV input

Create JSON

Assign Team, stadium, Players, game, Starting lineup’s, Own goals to files

Specify directory Address

For file in files

Assign csv files to input file

Specify JSON file as output file

Open csv input file ‘r’

Open JSON outfile, w and assign it to JSON file

Declare fieldnames = ‘ ‘

If file == ‘Team’

Assign Teamid, Team, Continent, league, population to fieldnames

If file == ‘stadium’

Assign SID, SName, SCity, SCapacity to fieldnames

If file == ‘players’

Assign player attributes to fieldnames

If file == ‘Game’

Assign Game attributes to field names

If file == ‘Starting\_lineups’

Assign Starting-lineups attributes to fieldnames

If file == ‘Goals’

Assign Goal attributes to field names

If file==’Own\_Goals’

Assign Own\_Goals attributes to field names

Read csv files with fieldnames and assign it to reader

Parse each row of csv file and convert it to JSON file

//Insert JSON file into Mongo DB

insertIntoDB();

Declare files with ith its attributes

Assign correct location in directory

For file in files

Input file is .csv file and open input file

If file == ‘Team’

Soccerdata is connected to mongo db

Declare Team attributes and assign it to fieldnames

Insert attribute values to each row

If file == ‘Stadium’

Soccerdata is connected to mongo db

Declare Stadium attributes and assign it to fieldnames

Insert attribute values to each row

If file == ‘Game’

Soccerdata is connected to mongo db

Declare Game attributes and assign it to fieldnames

Insert attribute values to each row

If file == ‘Stadium’

Soccerdata is connected to mongo db

Declare Stadium attributes and assign it to fieldnames

Insert attribute values to each row

If file == ‘Goal’

Soccerdata is connected to mongo db

Declare Stadium attributes and assign it to fieldnames

Insert attribute values to each row

If file == ‘Starting\_Lineups”

Soccerdata is connected to mongo db

Declare Starting\_Lineups attributes and assign it to fieldnames

Insert attribute values to each row

If file == ‘Players’

Soccerdata is connected to mongo db

Declare Player attributes and assign it to fieldnames

Insert attribute values to each row

If file == ‘Goals”

Soccerdata is connected to mongo db

Declare Goals attributes and assign it to fieldnames

Insert attribute values to each row

Print Mongo db success message

//Create Player\_data Document

createDocumentPlayer\_Data();

Connect to soccer file

Find country data through database and assign it to getAllCountries

For country in all countries

Get players playing for same country

For player in players

// Get card details of players

Retreive player id of player and assign it to playerCards

Declare and assign no of red and yellow cards to 0

Get no of yellow and red cards of players

//Get Goal assists of Player

Retreive player id of player and assign it to playerCards

Declare and assign no of goals and assists to 0

Get no of goals and assists of players

Declare PlayersList with PlayerId, fname, lname, height, DOB, no of yellow cards, no of red cards, no of goals, no of assists as attributes

Append PlayerList

//Get worldcup history for that country

Declare empty array and assign it to WorldcupHistory array

if (Worldcup won by country is 0 )

WorldcupHistory array is empty

Else

Retreive country name, year won and host of the worldcup

Assign Country Name to WorldCupswon and Year and host to worldCupList

Append worldCupHistoryArray by passing worldCupList as parameter

Declare population and no of Worldcupwon

Into the database, insert

Country name, capital, population, manager, Players, worldcup history

//Create Team\_Scores Document

createDocumentTeam\_Scores()

Fetch all stadium inform and assign it to getAllStadium

For stadium in getAllStadium

Fetch stadium name and hast city and assign it to stadiumCity

Fetch match results and assign it to stadiumDetails

Declare match list and have array assigned to it with attributes team1, team2, team1 score. Team2 score, date

Append matchList

Insert into stadium

Stadium,city,match history

If\_name\_ = ‘\_main\_’;

createJson();

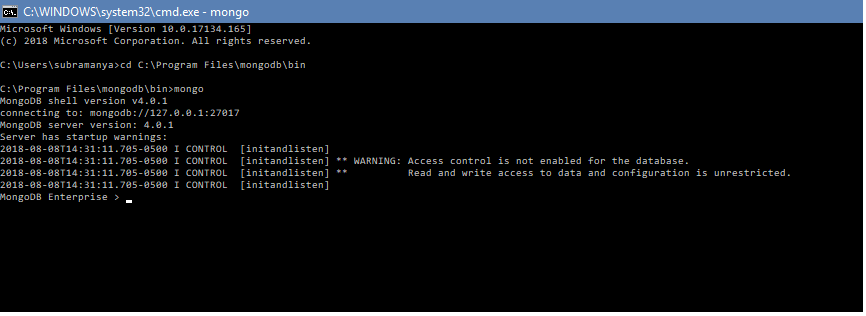
insertIntoDB();

createDocumentCountry();

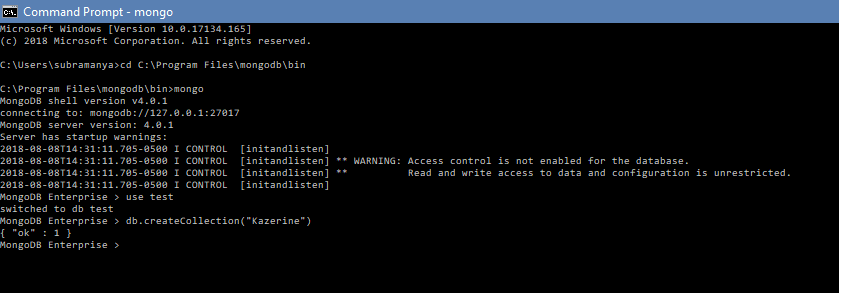
createDocumentStadium();

**Demonstration of MongoDb :-**

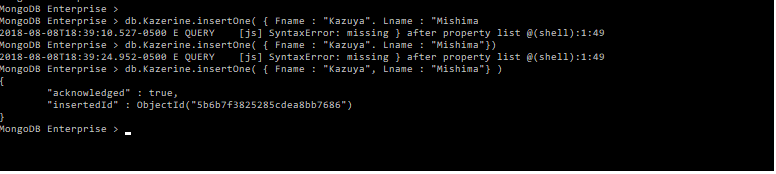
1. First we need to install mongoDB into our system, then open the command window and change directory to C:\Program Files\mongodb\bin . Then type mongo as follows :-



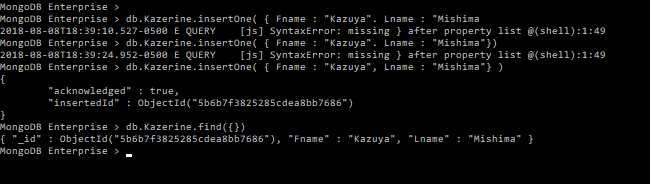
2. Then create a database using “use test”. Then create a collection in the database created by typing “db.createCollection(“Kazerine”).



3. Insert a record using insertOne method in the created collection “Kazerine” using db.Kazerine.insertOne( { Fname : “Kazuya”, Lname : “Mishima”} )



4. To display the record created inside the collection, type db.Kazerine.find({ })



**Data Structures used :-**

**1. CSV** and **JSON** modules are imported.

**2.**  Various **file operations** are used to extract data and to write data.

**3.**  **File** data structure is used to hold the field names of the files.

**4.**  **Reader** object is used to read data from CSV files.

**5.**  **Connection** object is used to connect to MongoDb.

**NOTE :** The provided input files are converted from .txt to .csv and then converted into a JSON file.