CSE 5331/4331 **Summer 2017**

Project 2 – Intermediate Report

Team Members:

Harikrishna Bathala (1001415489)

Rohit Katta (1001512896)

Programming Language: Python 2.7

Data Structures:

Python dictionary (which is a JSON)

Project:

{‘pname’ –project name, ‘pnumber’ -project number, ‘dname’- department number and ‘employees’ which is again a JSON which has list of employees with {‘fname’-first name,’lname’- last name, ‘hours’= no of hours worked} }

Department:

{‘Dname’- department name, ‘lname’- manager’s last name and locations which is again a JSON which includes list of ‘locations’ {(‘dlocations’- department locations) } }

Design/PseudoCode:

1.) Creating tables of employee, department, dept_locations, project, works_on in SQL and load the respective table data into them.
2.) Writing a SQL query by joining project, department, employee and works_on and loading those tuples into the mongodb document ‘project’.
3.) Writing a SQL query by joining dept_locations, department and employee and loading those tuples into the mongodb document ‘department’.
4.) Now, we can write mongodb queries on the both documents ‘project’ and ‘department’ to retrieve data based on our requirements.

Create tables in MySQL:

```
CREATE TABLE employee (
        fname varchar(255) NOT NULL,
        minit varchar(4),
        lname varchar(255) NOT NULL,
```

```
        Ssn char(9) NOT NULL,
        Bdate varchar(20),
        address varchar(255),
        sex char,salary decimal(10,2),
        Super_ssn char(9) NOT NULL,
        dno int NOT NULL,
        PRIMARY KEY (Ssn))

CREATE TABLE department (
        dname varchar(255) NOT NULL,
        dnumber int NOT NULL,
        Mgr_Ssn char(9) NOT NULL,
        Mgr_start_date varchar(20),
        PRIMARY KEY (dnumber))

CREATE TABLE dept_locations (
        dnumber int NOT NULL,
        dlocation varchar(255) NOT NULL,
        PRIMARY KEY (dnumber,dlocation))

CREATE TABLE project (
        pname varchar(255) NOT NULL,
        pnumber int NOT NULL,
        plocation varchar(15),
        dnum int NOT NULL,
        PRIMARY KEY (pnumber))

CREATE TABLE works_on (
        Essn char(9) NOT NULL,
        pno int NOT NULL,
        hours decimal(3,1) NOT NUll,
```

PRIMARY KEY (Essn,pno))

To Insert into MySql:

Python Modules that will be used are PyMySQL.

By pymysql.connect() method we connect to MySQL database.

Conn = pymysql.connect(host ="" ; ….[connection parameters]).

cr = conn.cursor()

Now using the following commands, we will insert all csv file data to MySql tables.

emp_insert_query = **r"LOAD DATA local INFILE 'D:\\CourseWork\\Summer2017\\Db2\\Project2\\EMPLOYEE.csv' INTO TABLE employee FIELDS TERMINATED BY ', ' OPTIONALLY ENCLOSED BY '"' LINES TERMINATED BY '\r\n'"**

dept_insert_query = **r"LOAD DATA local INFILE 'D:\\CourseWork\\Summer2017\\Db2\\Project2\\DEPARTMENT.csv' INTO TABLE department FIELDS TERMINATED BY ', ' OPTIONALLY ENCLOSED BY '"' LINES TERMINATED BY '\r\n'"**

dept_loc_insert_query = **r"LOAD DATA local INFILE 'D:\\CourseWork\\Summer2017\\Db2\\Project2\\DEPT_LOCATIONS.csv' INTO TABLE dept_locations FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"' LINES TERMINATED BY '\r\n'"**

proj_insert_query = **r"LOAD DATA local INFILE 'D:\\CourseWork\\Summer2017\\Db2\\Project2\\PROJECT.csv' INTO TABLE project FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"' LINES TERMINATED BY '\r\n'"**

works_insert_query = **r"LOAD DATA local INFILE 'D:\\CourseWork\\Summer2017\\Db2\\Project2\\WORKS_ON.csv' INTO TABLE works_on FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"' LINES TERMINATED BY '\r\n'"**

```
cur.execute(emp_insert_query)
cur.execute(dept_insert_query)
cur.execute(dept_loc_insert_query)
cur.execute(proj_insert_query)
cur.execute(works_insert_query)
```

Now we commit the data using the following command.

```
conn.commit()
```

We can now check that the data is loaded into MySQL database with the following queries.

Queries to Show MySQL Data:

```
mysql> select * from dept_locations;
+---------+----------------+
| dnumber | dlocation      |
+---------+----------------+
|       1 | 'Houston'      |
|       4 | 'Stafford'     |
|       5 | 'Bellaire'     |
|       5 | 'Houston'      |
|       5 | 'Sugarland'    |
|       6 | 'Atlanta'      |
|       6 | 'Sacramento'   |
|       7 | 'Milwaukee'    |
|       8 | 'Chicago'      |
|       8 | 'Dallas'       |
|       8 | 'Miami'        |
|       8 | 'Philadephia'  |
```

```
|       8 | 'Seattle'      |
|       9 | 'Arlington'    |
|      11 | 'Austin'       |
+---------+----------------+
15 rows in set (0.00 sec)

mysql> select * from project;
+------------------+---------+--------------+------+
| pname            | pnumber | plocation    | dnum |
+------------------+---------+--------------+------+
| 'ProductX'       |       1 | 'Bellaire'   |    5 |
| 'ProductY'       |       2 | 'Sugarland'  |    5 |
| 'ProductZ'       |       3 | 'Houston'    |    5 |
| 'EntityAnnot'    |       4 | 'Houston'    |    5 |
| 'Computerization' |      10 | 'Stafford'   |    4 |
| 'ConfigMgmt'     |      11 | 'Atlanta'    |    6 |
| 'DataMining'     |      13 | 'Sacramento' |    6 |
| 'Reorganization'  |      20 | 'Houston'    |    1 |
| 'SearchEngine'   |      22 | 'Arlington'  |    6 |
| 'MotherBoard'    |      29 | 'Milwaukee'  |    7 |
| 'Newbenefits'    |      30 | 'Stafford'   |    4 |
| 'OperatingSystem' |      61 | 'Sacramento' |    6 |
| 'DatabaseSystems' |      62 | 'Atlanta'    |    6 |
| 'Middleware'     |      63 | 'Atlanta'    |    6 |
| 'Advertizing'    |      70 | 'Arlington'  |    9 |
| 'InkjetPrinters'  |      91 | 'Milwaukee'  |    7 |
| 'LaserPrinters'  |      92 | 'Milwaukee'  |    7 |
| 'Human1'         |      95 | 'Arlington'  |    9 |
+------------------+---------+--------------+------+
18 rows in set (0.00 sec)
```

Inserting into MongoDb:

We now have data in MySQL tables. The format of documents we need in MongoDb are

1. The PROJECT document will include the following data: Pnumber, Pname, Dname (of the controlling department), plus a list of the employees that work on the project {employees: Lname, Fname, Hours}.
2. The DEPARTMENT document will include the following data: Dname, the department manager's Lname, and a list of the locations of the department {locations: Dlocation}

To achieve this we retrieve data from MySQL using joins.

For PROJECT document:

SELECT p.pnumber, p.pname, d.dname, e.lname, e.fname, w.hours FROM project p JOIN department d ON p.dnum = d.dnumber JOIN works_on w ON p.pnumber = w.pno JOIN employee e ON w.Essn = e.Ssn ORDER BY p.pnumber ;

For DEPARTMENT document:

SELECT d.dname, e.lname, dl.dlocation FROM department d JOIN dept_locations dl ON d.dnumber = dl.dnumber JOIN employee e ON d.Mgr_Ssn = e.Ssn ORDER BY d.dname;

In mongodb, we create a database named 'project2' in which we store the above 'project' and 'department' documents.

Python Modules that will be used are pymongo.

```
client = MongoClient('localhost', 27017)
//for 'project' document
""" inserting project table into mongodb document named 'project' in collection 'project2"""
db = client.project2.project
cur.execute(r"SELECT p.pnumber, p.pname, d.dname, e.lname, e.fname, w.hours FROM project p JOIN department d ON p.dnum = d.dnumber JOIN works_on w ON p.pnumber = w.pno JOIN employee e ON w.Essn = e.Ssn ORDER BY p.pnumber ")
```

project_table = cur.fetchall()

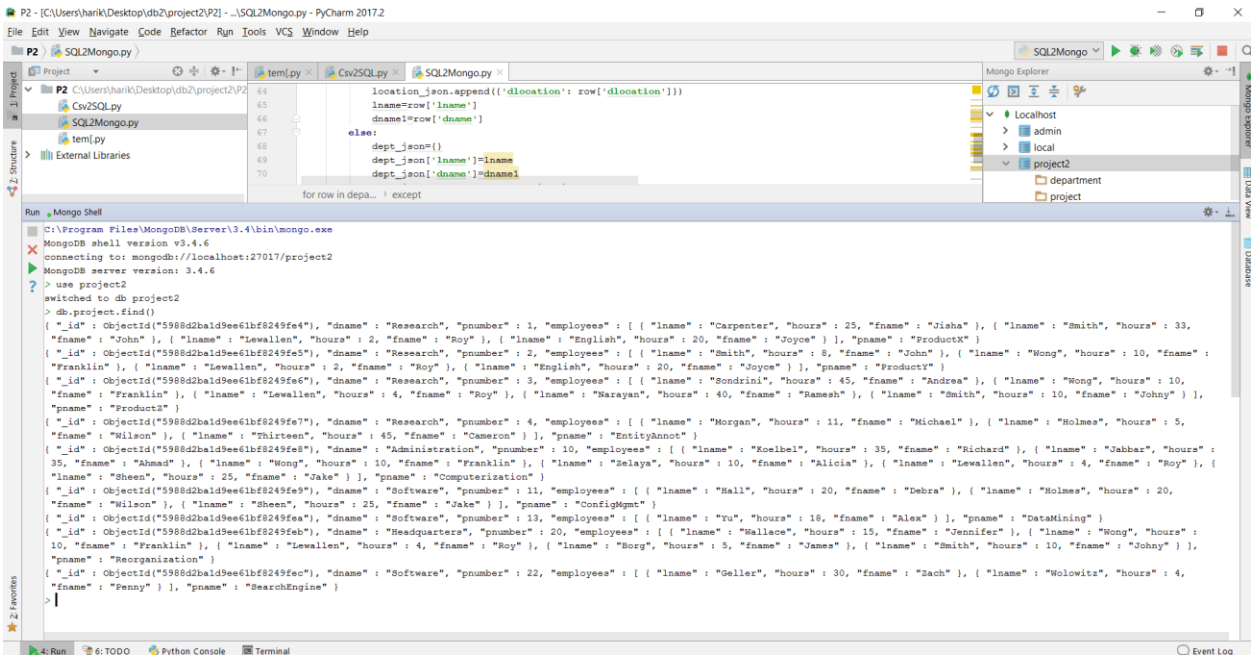
for each project:

we store lname,fname,hours of an employee(from all rows) for a particular project as JSON(employee_json).

```
employee_json.append({'lname': row['lname'], 'fname': row['fname'], 'hours': row['hours']})
```

Project_json has ‘pname’, ‘dname’, ‘pnumber’, ‘employees’

```
project_json={}
project_json['pname']=pname
project_json['dname']=dname
project_json['pnumber']=pnumber
project_json['employees']=employee_json
db.insert(project_json)
```



//for ‘department’ document

""" inserting DEpartment Table into mongodb document named 'department' in collection 'project2' """

cur.execute(**"SELECT d.dname, e.lname, dl.dlocation FROM department d JOIN dept_locations dl ON d.dnumber = dl.dnumber JOIN employee e ON d.Mgr_Ssn = e.Ssn;"**)

department_Table = cur.fetchall()

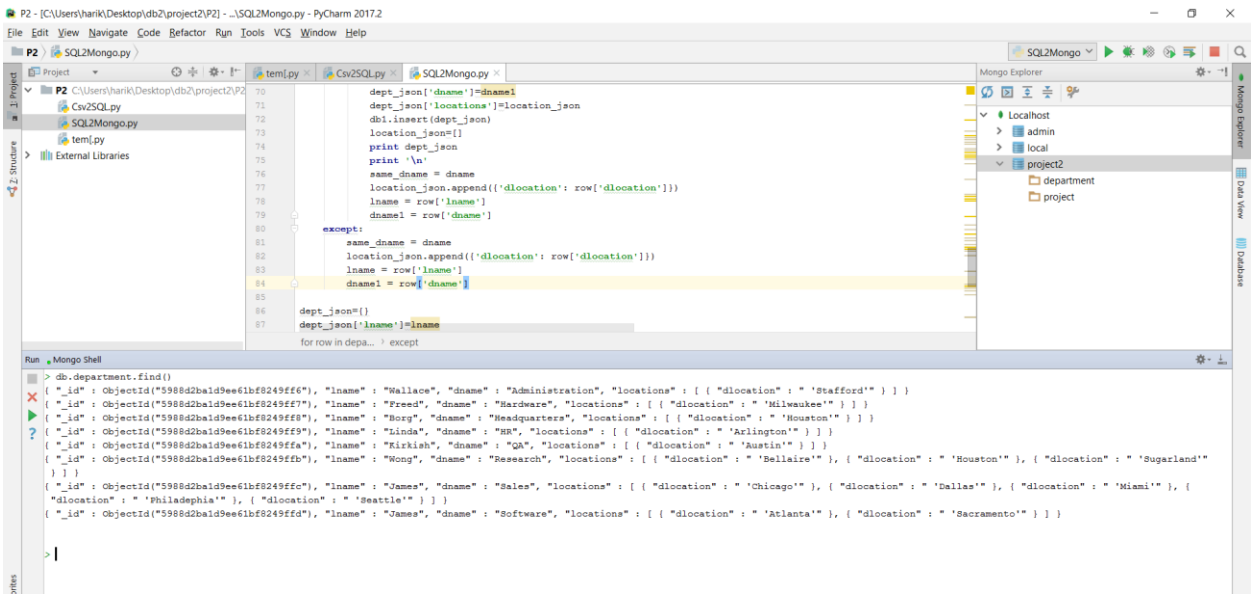
for each department:

we store dept_locations of a department (from all rows) as JSON(location_json).

```
location_json.append({'dlocation': row['dlocation']})
```

department_json has ‘lname’, ‘dname’, ‘locations’

```
dept_json={}
dept_json['lname']=lname
dept_json['dname']=dname1
dept_json['locations']=location_json
db1.insert(dept_json)
```



To show Mongodb installed:

> use project2

switched to db project2

> db.department.find()

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ff6"), "lname" : "Wallace", "dname" : "Administration", "locations" : [{ "dlocation" : " 'Stafford'" }] }

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ff7"), "lname" : "Freed", "dname" : "Hardware", "locations" : [{ "dlocation" : " 'Milwaukee'" }] }

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ff8"), "lname" : "Borg", "dname" : "Headquarters", "locations" : [{ "dlocation" : " 'Houston'" }] }

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ff9"), "lname" : "Linda", "dname" : "HR", "locations" : [{ "dlocation" : " 'Arlington'" }] }

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ffa"), "lname" : "Kirkish", "dname" : "QA", "locations" : [{ "dlocation" : " 'Austin'" }] }

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ffb"), "lname" : "Wong", "dname" : "Research", "locations" : [{ "dlocation" : " 'Bellaire'" }, { "dlocation" : " 'Houston'" }, { "dlocation" : " 'Sugarland'" }] }

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ffc"), "lname" : "James", "dname" : "Sales", "locations" : [{ "dlocation" : " 'Chicago'" }, { "dlocation" : " 'Dallas'" }, { "dlocation" : " 'Miami'" }, { "dlocation" : " 'Philadephia'" }, { "dlocation" : " 'Seattle'" }] }

{ "_id" : ObjectId("5988d2ba1d9ee61bf8249ffd"), "lname" : "James", "dname" : "Software", "locations" : [{ "dlocation" : " 'Atlanta'" }, { "dlocation" : " 'Sacramento'" }] }

Note: Refer Screenshots attached separately.