**Project Report**

Implement a program that helps to learn MongoDB as an example of a document-oriented NOSQL system

**Programming Language**: Python

**Project Description:**

This project includes following steps:

1.Importing relational .txt files into mysql

2.Converting the 5 “.txt” files into two documents using my sql queries(Project,Employee)

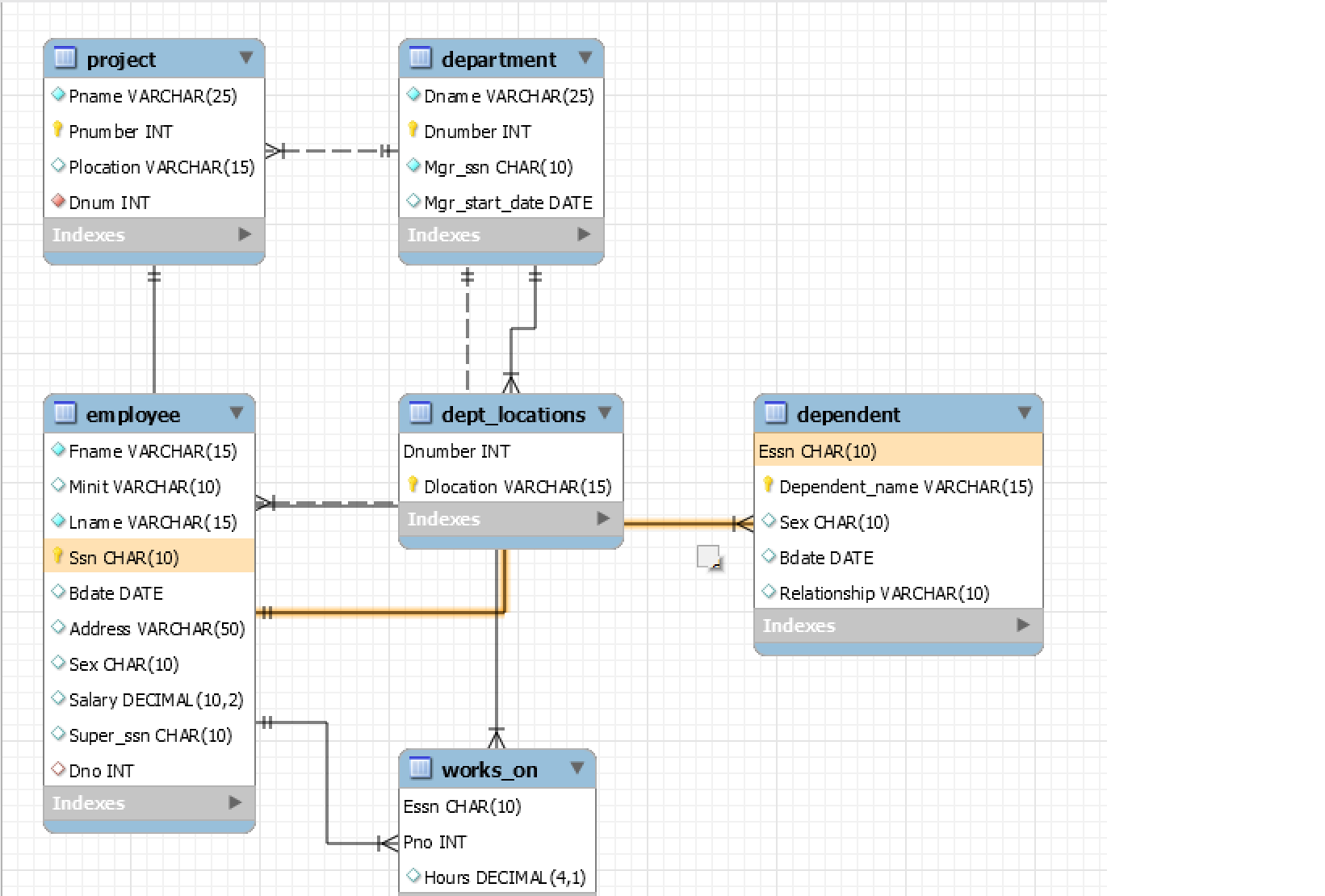
3.Using Python for connection to mysql workbench and retrieving the queries

in spyder IDLE

4. Using python to convert relational files to json documents

5. Importing json documents into MONGODB for storage as “Objects”

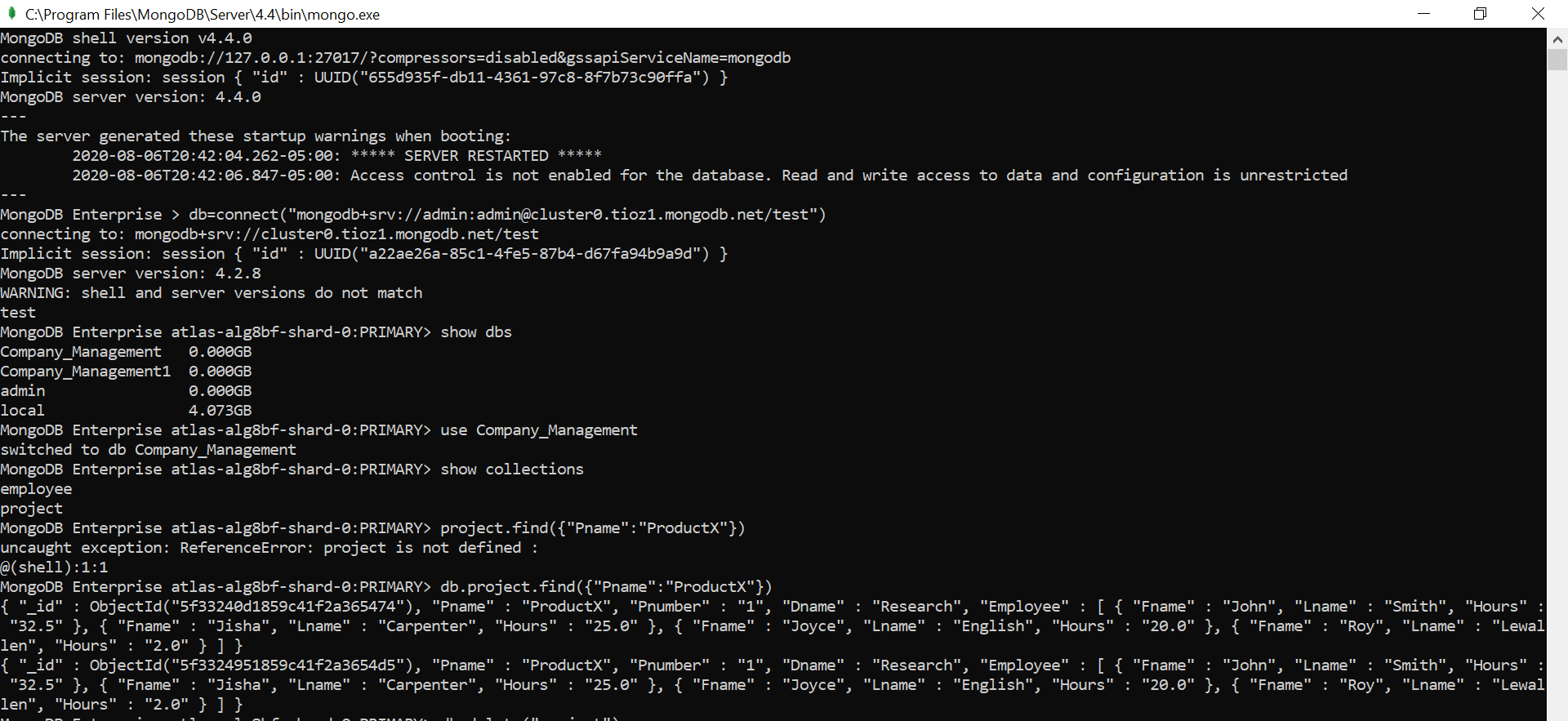
**Schema ER Diagram:**



**Mongo DB :**

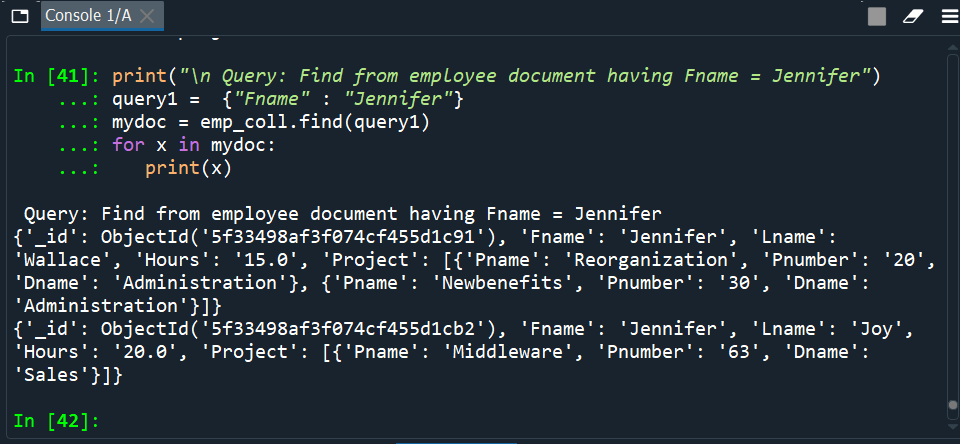
1. Mongo Shell showing mongo db version and other details.

Also showing connection made to db = Company\_Management and collections present in it.



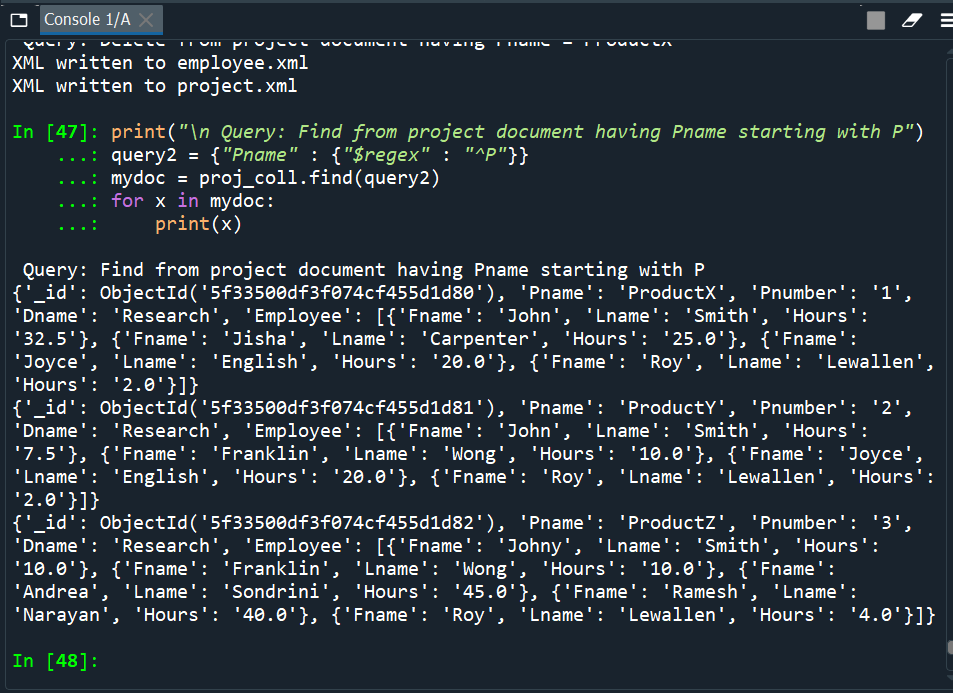
1. Task4 - Query1(find() method in mongo DB)

Find from employee document having Fname = Jennifer



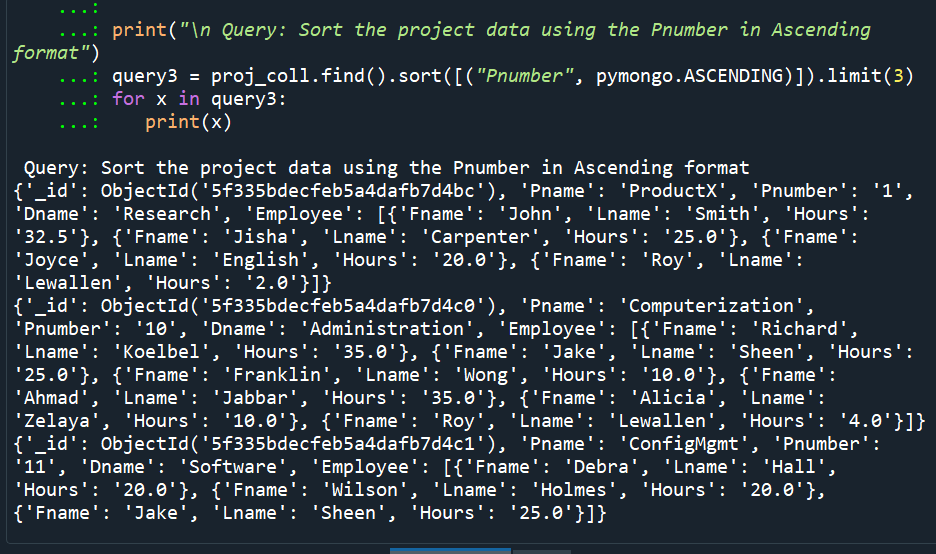
1. Task4 – Query 2 (regex method)

Find from project document having Pname starting with P



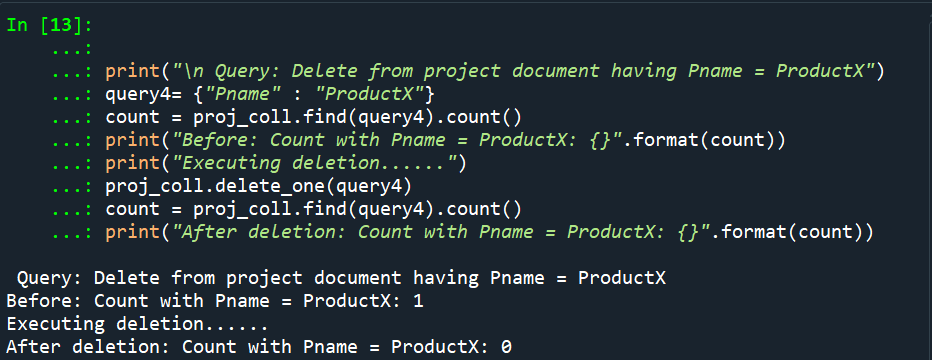
1. Task4 – Query3( Sort and limit)

Sort the project data using the Pnumber in Ascending format



1. Task4 – Query4 (Delete)

Delete from project document having Pname = ProductX



**Extra Credit Description:**

1. XML:

Using python we imported json file data from mongoDB and then we used “dicttoxml” module to convert two documents to XML

1. Creating department document

The procedure is same as the one done for project and employee json

**Project Contribution:**

All the work that includes developing the code ,Testing code for different inputs, building the project report is done equally by both of us

**Pseudo Code:**

**Step1: Importing files into Database**

Create database “company\_management”

Create tables - employee, department, project, works\_on, dept\_locations

Load the given data from text file into MySQL.

**Step2: Joining 5 text files into 2 documents**

Query will be in the form of:

SELECT “ Attribute names” AS “Aliases” FROM “Table name” **JOIN**  “table names” ON “join-condition

**Step3: Using python to connect mysql in spyder**

import mysql.connector

mydb = mysql.connector.connect(

host="127.0.0.1",

user="root",

password="root",

database ="companymng"

)

Also includes execution of queries as mentioned in step2

**Step4 : Using python to convert to json documents**

cursor = mydb.cursor()

Built project query

res=cursor.execute(project\_query)

row\_headers=[x[0] for x in cursor.description]

proj\_result = cursor.fetchall()

proj\_result = [tuple(str(item) for item in t) for t in proj\_result]

proj\_output = {}

for Pnumber, Pname, Pnumber,Dname,Fname, Lname, Hours in proj\_result:

if Pnumber in proj\_output:

emp = {

"Fname" : Fname,

"Lname" : Lname,

"Hours" : Hours

}

proj\_output[Pnumber]["Employee"].append(emp)

else:

proj\_output[Pnumber] = {

"Pname" : Pname,

"Pnumber" : Pnumber,

"Dname" : Dname,

"Employee" : [{

"Fname" : Fname,

"Lname" : Lname,

"Hours" : Hours

}]

}

print(proj\_output)

filename="project.txt"

**Step5: Inserting data into mongoDB**

with open('employee.txt') as file:

file\_data = json.load(file)

if isinstance(file\_data, list):

Collection.insert\_many(file\_data)

else:

Collection.insert\_one(file\_data)