Date : 11-04-23

Assignment No 9

PRN No : 2020BTECS00092

Batch : T4

Name : Vishal Shrirang Madle

Lab : ADSL

Title: Install and deploy the following cloud databases on windows platform:

A. MongoDB

B. CassandraDB

Objective/Aim:

1. To install and deploy the given cloud databases on windows platform.
2. To create Python desktop application demonstrating the CRUD operation with above back end cloud databases.

Introduction:

NoSQL databases (aka "not only SQL") are non-tabular databases and store data differently than relational tables. NoSQL databases come in a variety of types based on their data model. The main types are document, key-value, wide-column, and graph. They provide flexible schemas and scale easily with large amounts of data and high user loads.

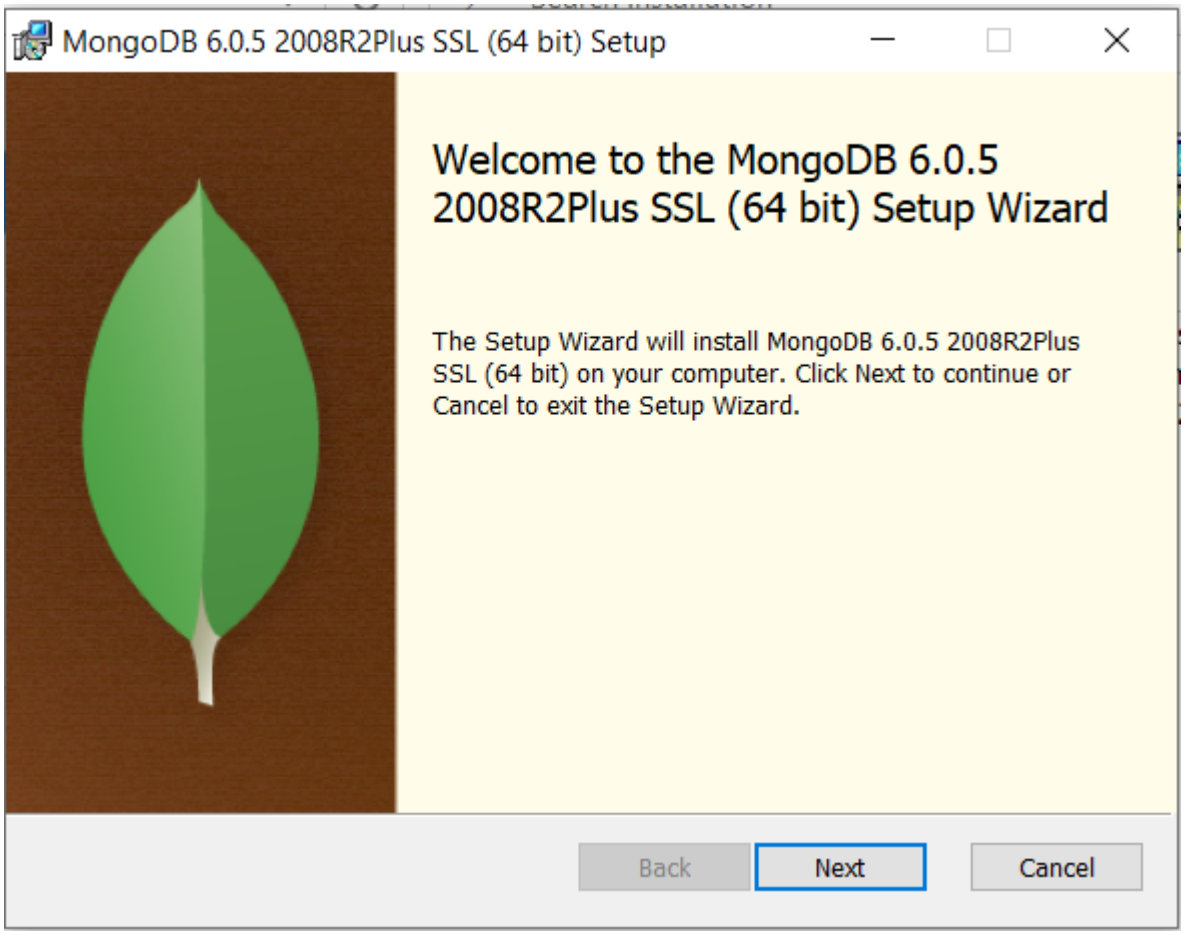
Theory:

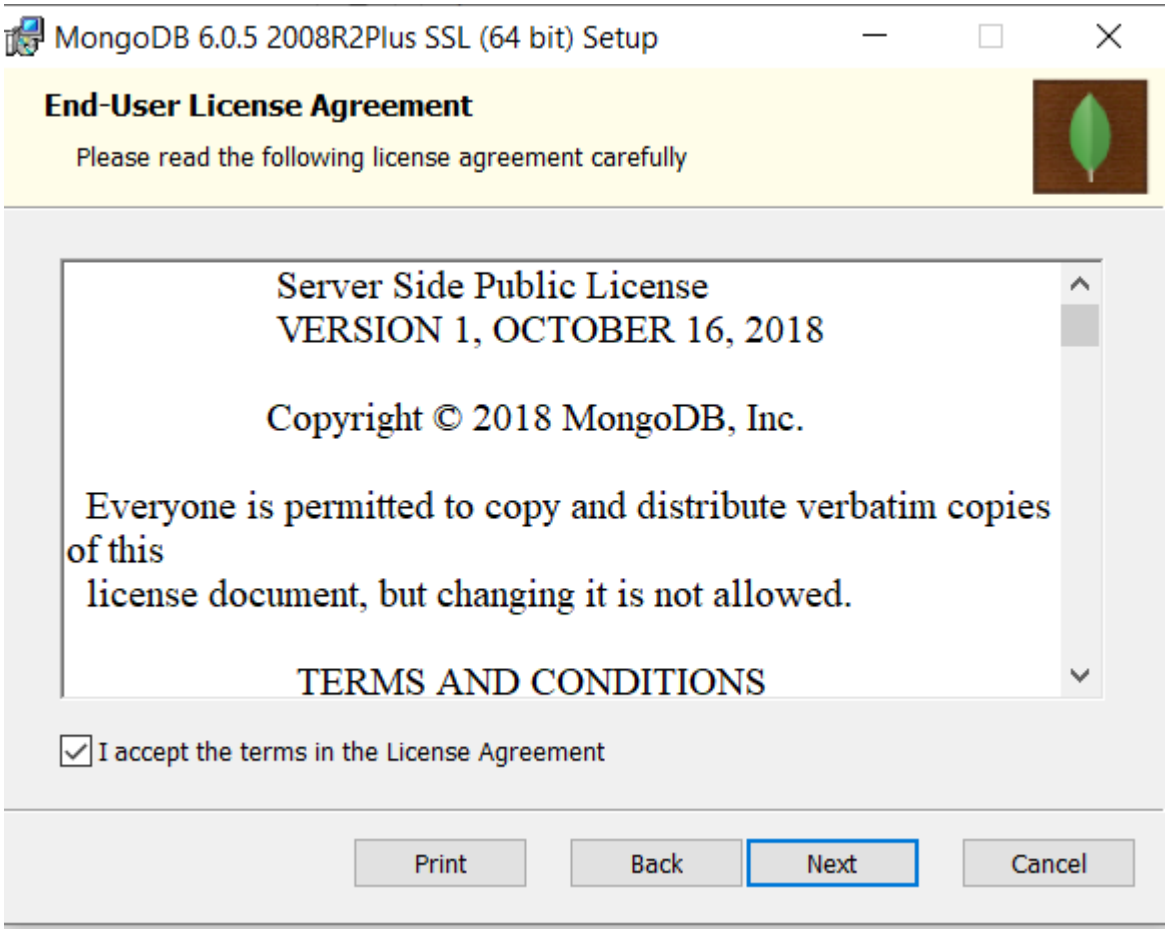
MongoDB is an open-source document database and leading NoSQL database. MongoDB is written in C++. MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

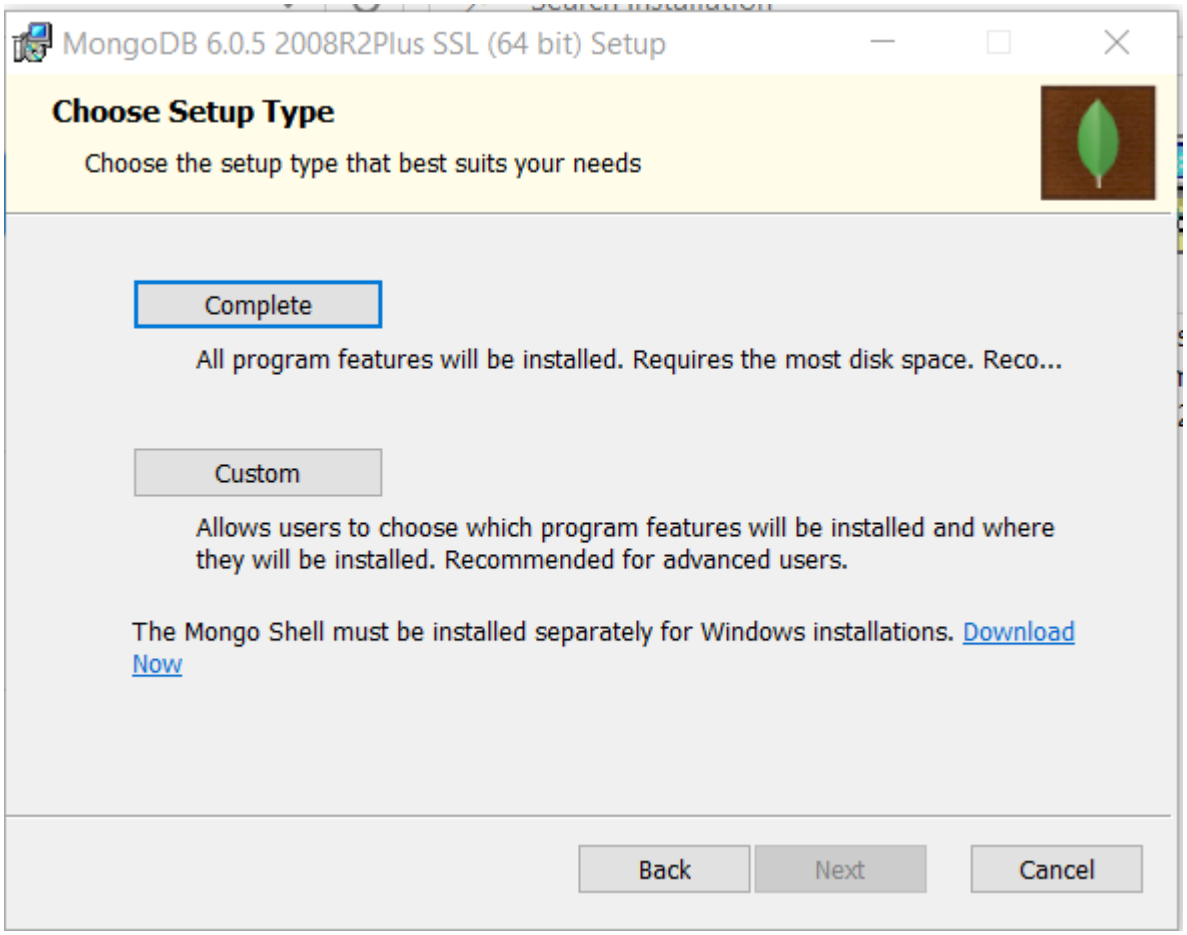
Apache Cassandra is a highly scalable, high-performance distributed database designed to handle large amounts of data across many commodity servers, providing high availability with no single point of failure. It is a type of NoSQL database.

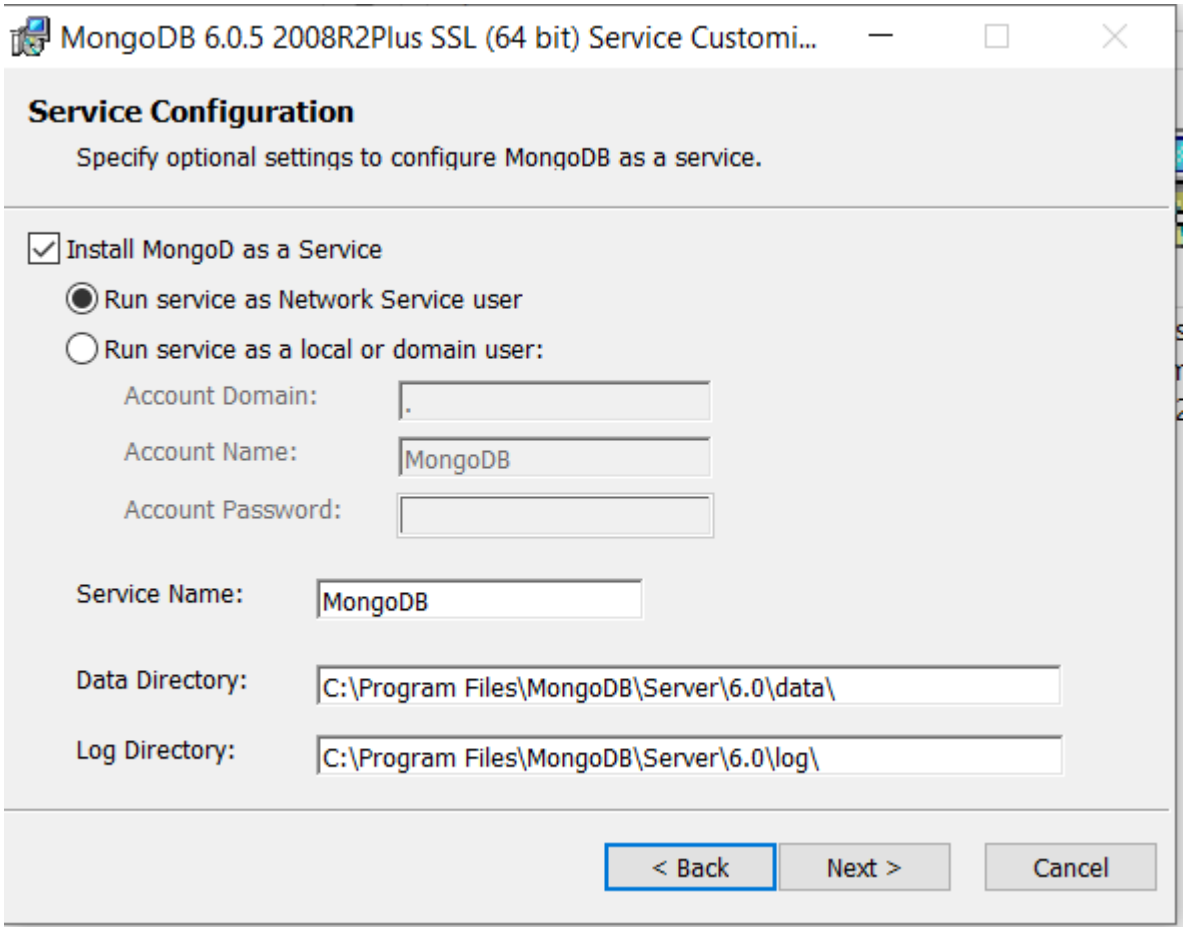
Procedure:

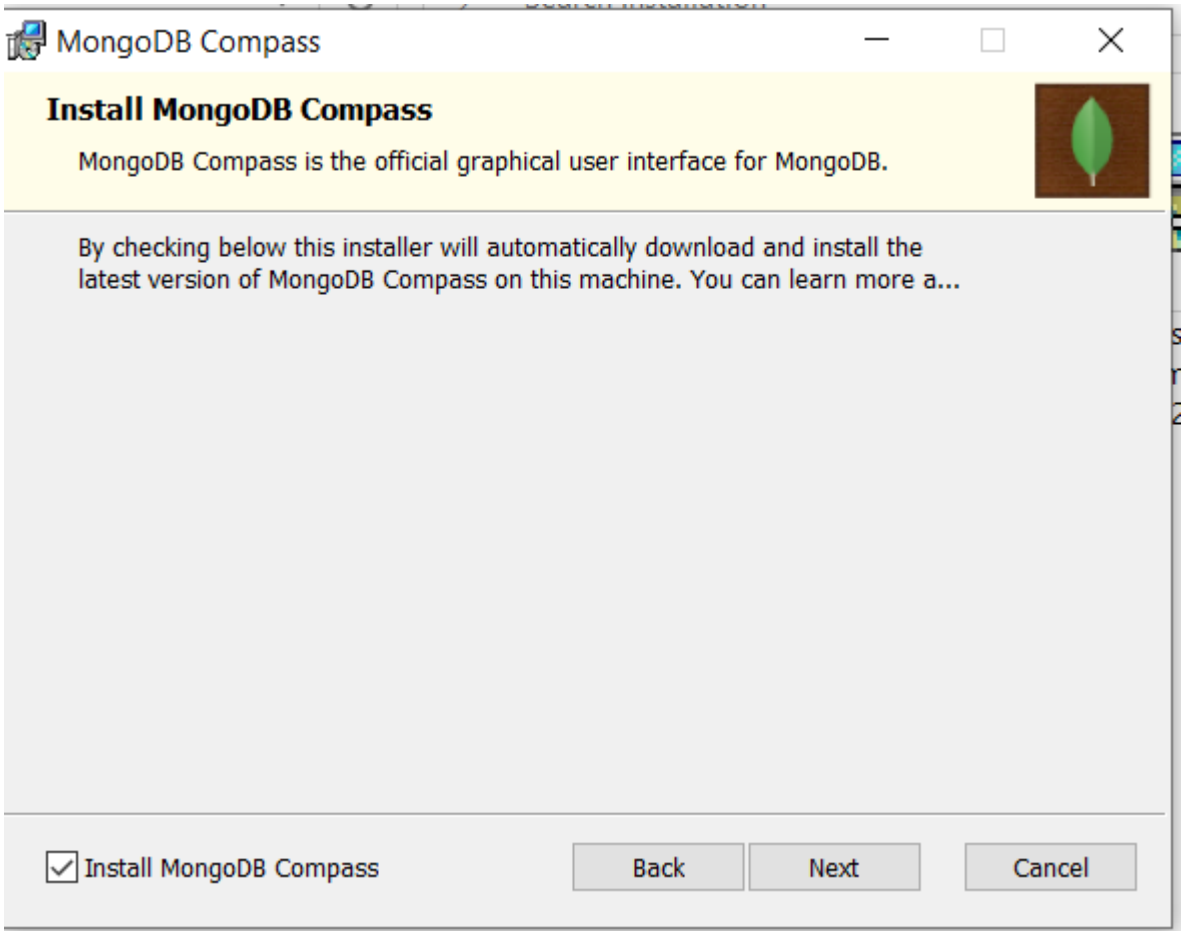
Added screenshots for the procedure followed:

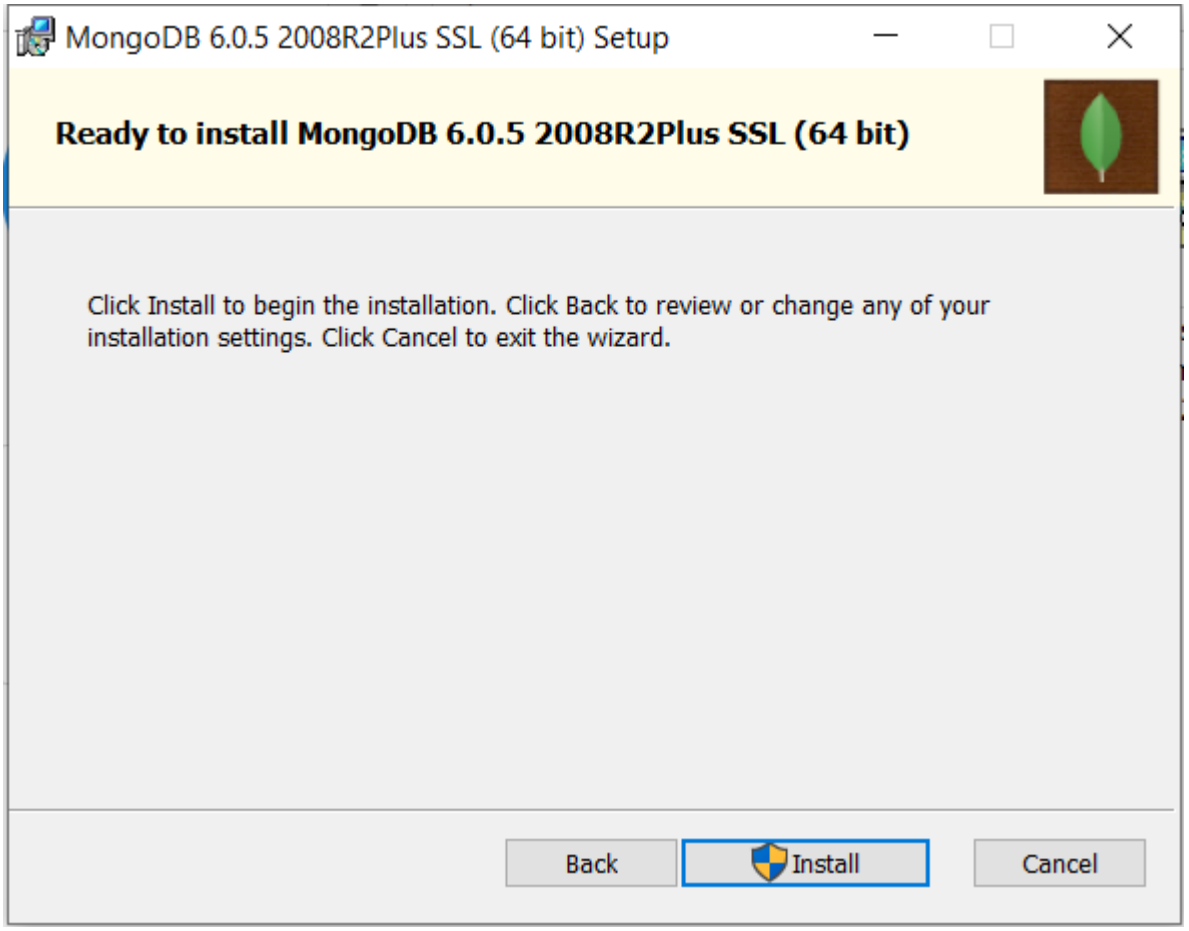

MongoDB 6.0.5 2008R2Plus SSL (64 bit) Setup
Welcome to the MongoDB 6.0.5 2008R2Plus SSL (64 bit) Setup Wizard
The Setup Wizard will install MongoDB 6.0.5 2008R2Plus SSL (64 bit) on your computer. Click Next to continue or Cancel to exit the Setup Wizard.
Back
Next
Cancel

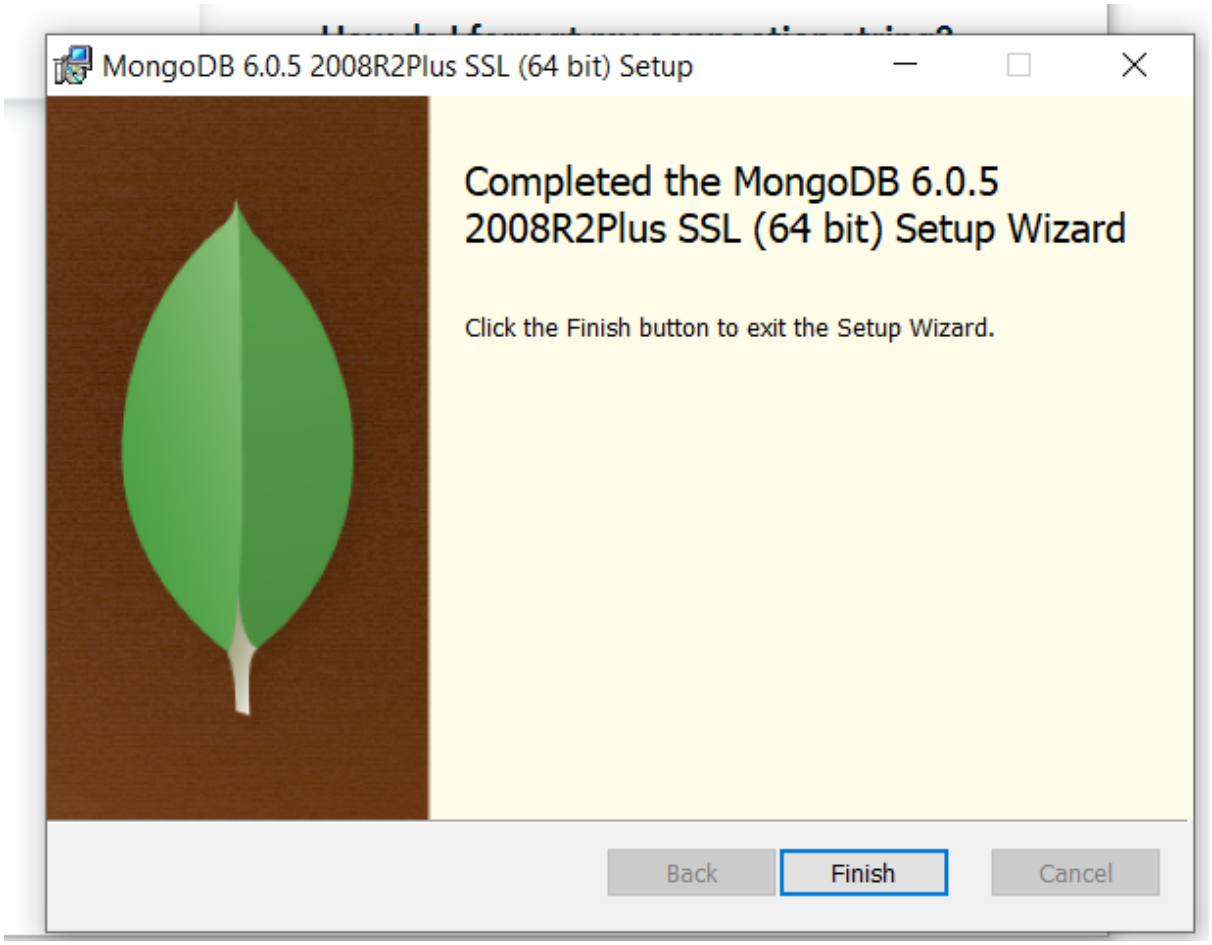

MongoDB 6.0.5 2008R2Plus SSL (64 bit) Setup
End-User License Agreement
Please read the following license agreement carefully
Server Side Public License
VERSION 1, OCTOBER 16, 2018
Copyright © 2018 MongoDB, Inc.
Everyone is permitted to copy and distribute verbatim copies of this
license document, but changing it is not allowed.
TERMS AND CONDITIONS
I accept the terms in the License Agreement
Print
Back
Next
Cancel


MongoDB 6.0.5 2008R2Plus SSL (64 bit) Setup
Choose Setup Type
Choose the setup type that best suits your needs
Complete
All program features will be installed. Requires the most disk space. Reco...
Custom
Allows users to choose which program features will be installed and where they will be installed. Recommended for advanced users.
The Mongo Shell must be installed separately for Windows installations. Download Now
Back
Next
Cancel


MongoDB 6.0.5 2008R2Plus SSL (64 bit) Service Customi...
Service Configuration
Specify optional settings to configure MongoDB as a service.
Install MongoD as a Service
Run service as Network Service user
Run service as a local or domain user:
Account Domain:
.
Account Name:
MongoDB
Account Password:
Service Name:
MongoDB
Data Directory:
C:\Program Files\MongoDB\Server\6.0\data\
Log Directory:
C:\Program Files\MongoDB\Server\6.0\log\
< Back
Next >
Cancel


MongoDB Compass
Install MongoDB Compass
MongoDB Compass is the official graphical user interface for MongoDB.
By checking below this installer will automatically download and install the latest version of MongoDB Compass on this machine. You can learn more a...
Install MongoDB Compass
Back
Next
Cancel


MongoDB 6.0.5 2008R2Plus SSL (64 bit) Setup
Ready to install MongoDB 6.0.5 2008R2Plus SSL (64 bit)
Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.
Back
Install
Cancel



Installation of MongoDB on windows:

Setting up environment variable for mongodb

Executed Python Desktop application to demonstrate CRUD operations for MongoDB:

Installation of CassandraDB on windows:

Installed

1. python 2.7.18 64 bit.

2. JDK 8

3. Stable version of CassandraDB

Installation of python 2.7.18

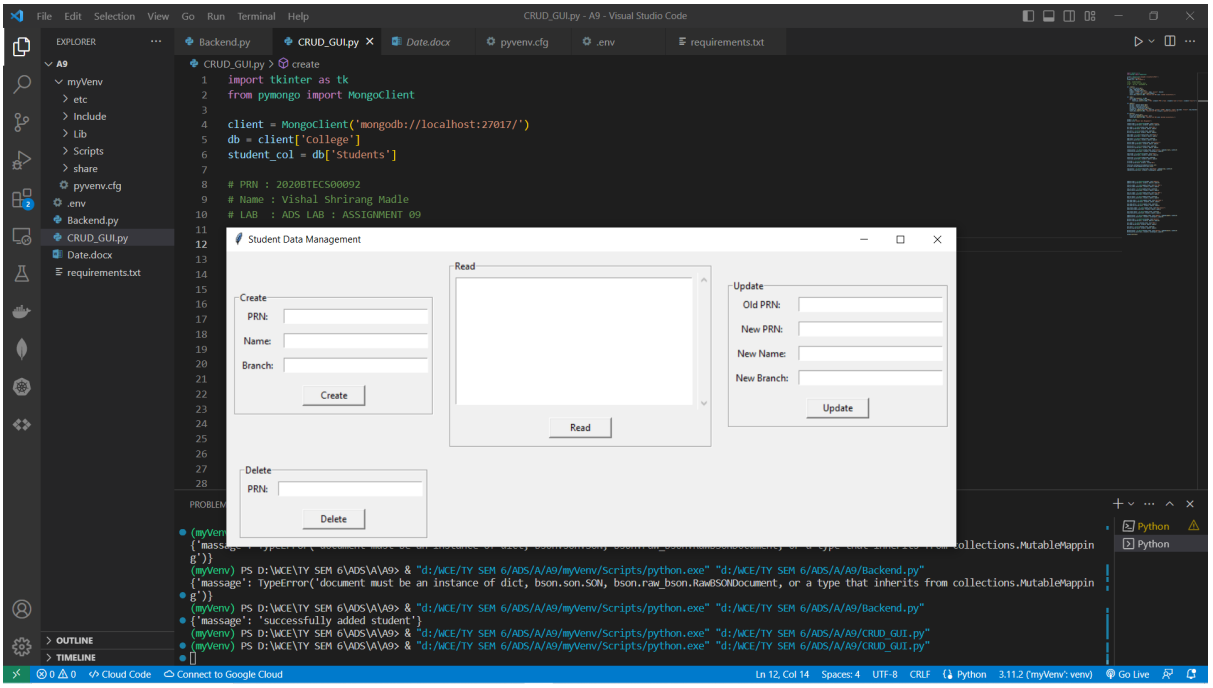
Installation of JDK 8

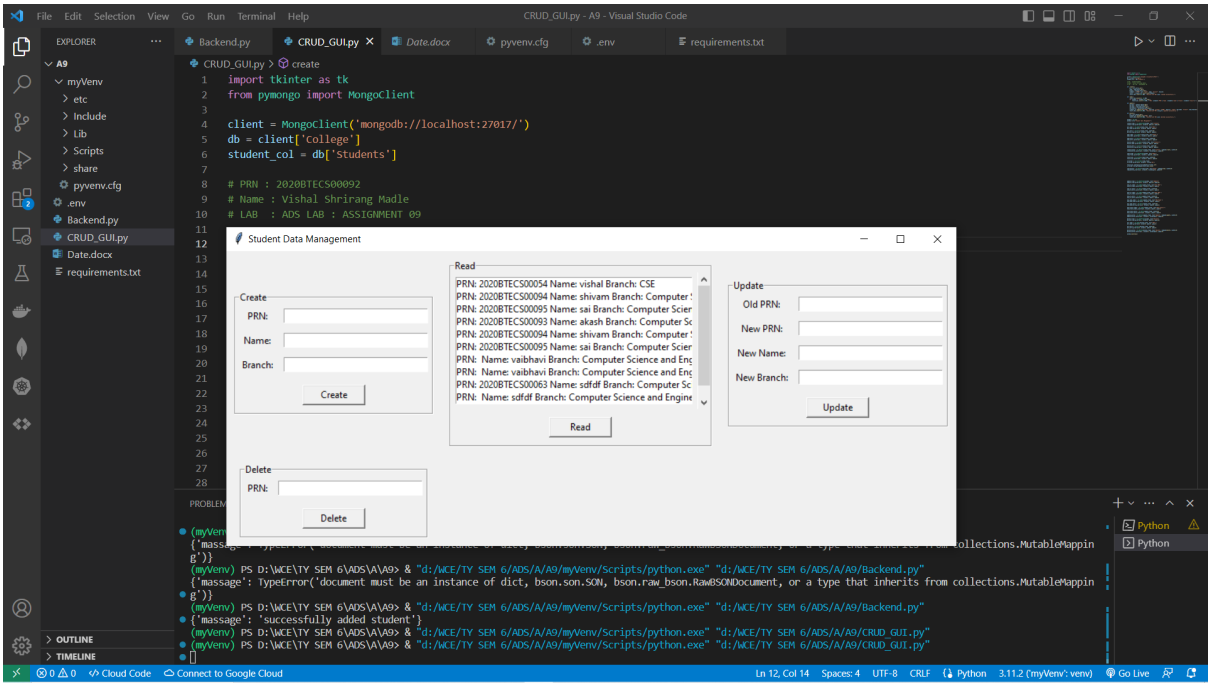
Setting-up environment variable for python, JDK and CassandraDB\

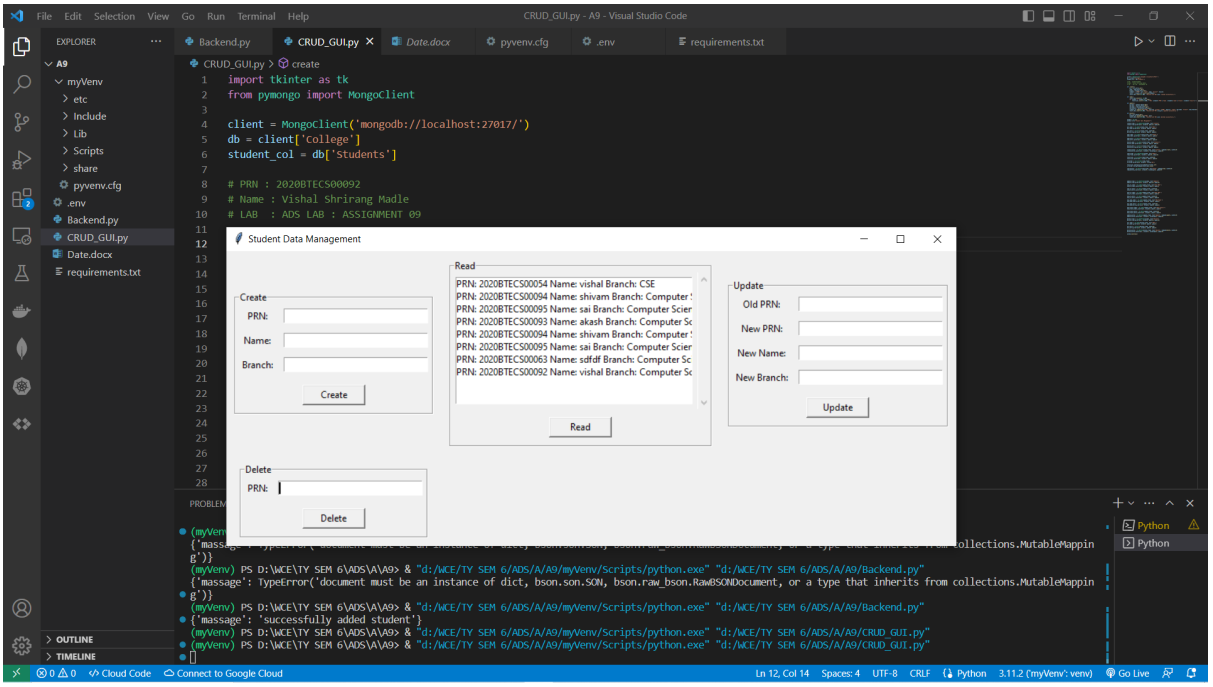
Conclusion:

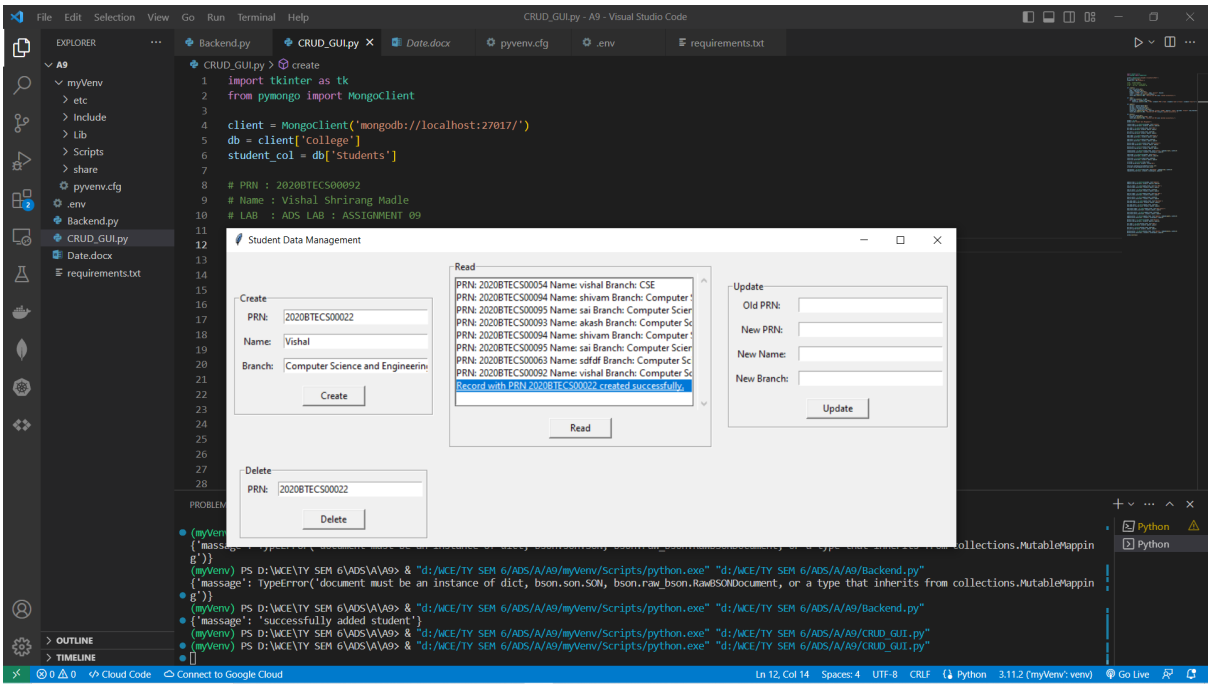
Installed and configured NoSQL MongoDB and CassandraDB on windows and demonstrated respective CRUD operations using python desktop application.

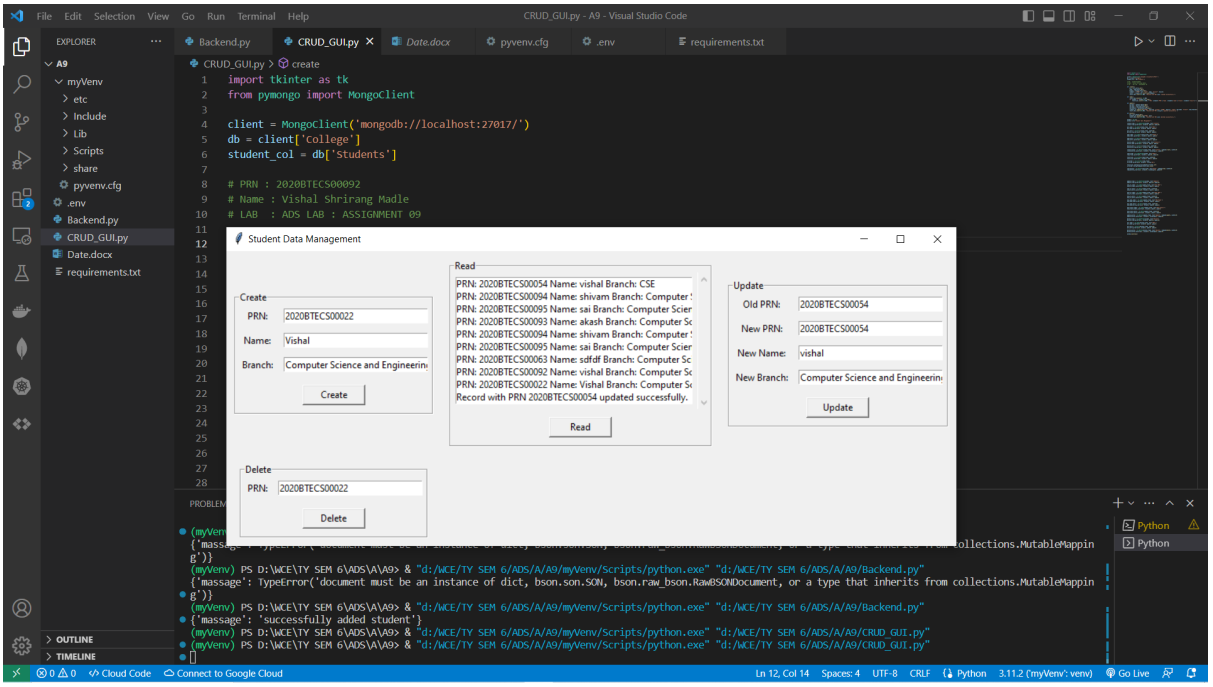
Screenshots :

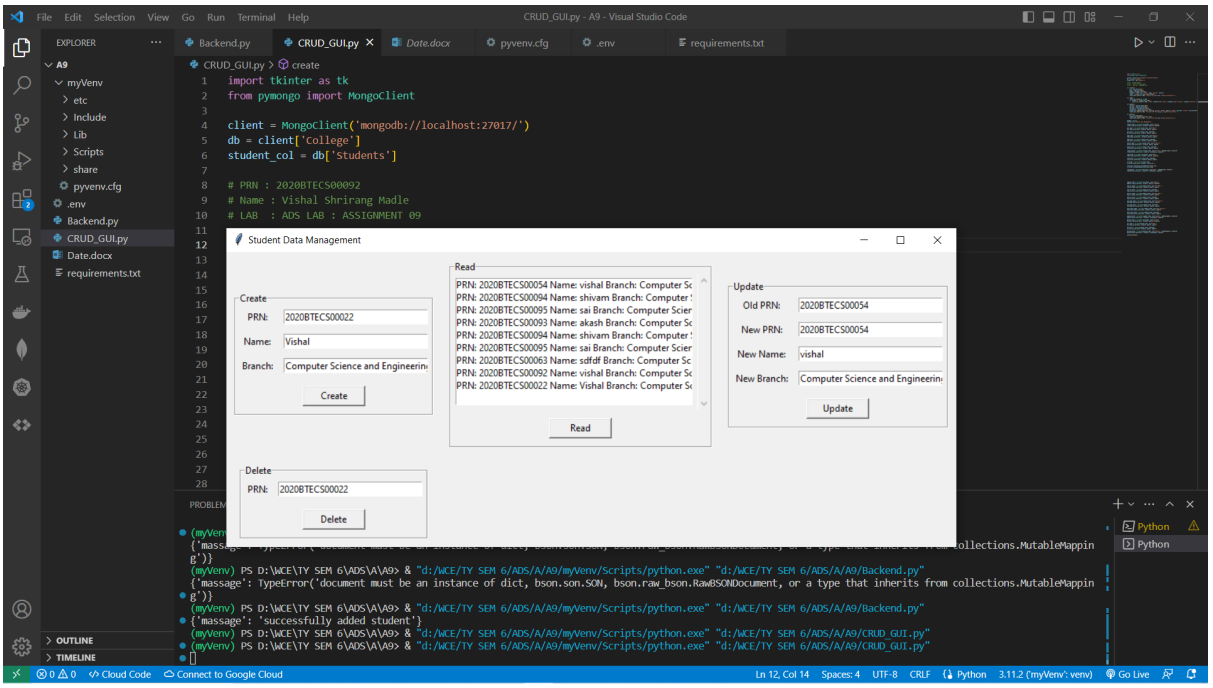





File Edit Selection View Go Run Terminal Help
CRUD_GUI.py - A9 - Visual Studio Code
EXPLORER
A9
myVenv
etc
Include
Lib
Scripts
share
pyvenv.cfg
.env
Backend.py
CRUD_GUI.py
Date.docx
requirements.txt
Backend.py
CRUD_GUI.py
Date.docx
pyvenv.cfg
.env
requirements.txt
CRUD_GUI.py > create
1 import tkinter as tk
2 from pymongo import MongoClient
4 client = MongoClient('mongodb://localhost:27017/')
5 db = client['College']
6 student_col = db['Students']
8 # PRN : 2020BTECS00092
9 # Name : Vishal Shrirang Madle
10 # LAB : ADS LAB : ASSIGNMENT 09
Student Data Management
Create
PRN:
Name:
Branch:
Create
Read
PRN: 2020BTECS00054 Name: vishal Branch: CSE
PRN: 2020BTECS00094 Name: shivam Branch: Computer
PRN: 2020BTECS00095 Name: sai Branch: Computer Scier
PRN: 2020BTECS00093 Name: akash Branch: Computer Sc
PRN: 2020BTECS00094 Name: shivam Branch: Computer
PRN: 2020BTECS00095 Name: sai Branch: Computer Scier
PRN: 2020BTECS00063 Name: sdfdf Branch: Computer Sc
PRN: 2020BTECS00092 Name: vishal Branch: Computer Sc
Read
Update
Old PRN:
New PRN:
New Name:
New Branch:
Update
Delete
PRN:
Delete
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/Backend.py"
{'massage': TypeError('document must be an instance of dict, bson.son.SON, bson.raw_bson.RawBSONDocument, or a type that inherits from collections.MutableMappin
g')}
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/Backend.py"
{'massage': 'successfully added student'}
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/CRUD_GUI.py"
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/CRUD_GUI.py"
OUTLINE
TIMELINE
Python
Python
0 0
Cloud Code
Connect to Google Cloud
Ln 12, Col 14
Spaces: 4
UTF-8
CRLF
Python
3.11.2 ('myVenv': venv)
Go Live


File Edit Selection View Go Run Terminal Help
CRUD_GUI.py - A9 - Visual Studio Code
EXPLORER
A9
myVenv
etc
Include
Lib
Scripts
share
pyvenv.cfg
.env
Backend.py
CRUD_GUI.py
Date.docx
requirements.txt
Backend.py
CRUD_GUI.py
Date.docx
pyvenv.cfg
.env
requirements.txt
CRUD_GUI.py > create
1 import tkinter as tk
2 from pymongo import MongoClient
4 client = MongoClient('mongodb://localhost:27017/')
5 db = client['College']
6 student_col = db['Students']
8 # PRN : 2020BTECS00092
9 # Name : Vishal Shrirang Madle
10 # LAB : ADS LAB : ASSIGNMENT 09
Student Data Management
Create
PRN: 2020BTECS00022
Name: Vishal
Branch: Computer Science and Engineerin
Create
Read
PRN: 2020BTECS00054 Name: vishal Branch: CSE
PRN: 2020BTECS00094 Name: shivam Branch: Computer
PRN: 2020BTECS00095 Name: sai Branch: Computer Scier
PRN: 2020BTECS00093 Name: akash Branch: Computer Sc
PRN: 2020BTECS00094 Name: shivam Branch: Computer
PRN: 2020BTECS00095 Name: sai Branch: Computer Scier
PRN: 2020BTECS00063 Name: sdfdf Branch: Computer Sc
PRN: 2020BTECS00092 Name: vishal Branch: Computer Sc
Record with PRN 2020BTECS00022 created successfully.
Read
Update
Old PRN:
New PRN:
New Name:
New Branch:
Update
Delete
PRN: 2020BTECS00022
Delete
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/Backend.py"
{'massage': TypeError('document must be an instance of dict, bson.son.SON, bson.raw_bson.RawBSONDocument, or a type that inherits from collections.MutableMappin
g')}
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/Backend.py"
{'massage': 'successfully added student'}
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/CRUD_GUI.py"
(myVenv) PS D:\WCE\TY SEM 6\ADS\A\A9> & "d:/WCE/TY SEM 6/ADS/A/A9/myVenv/Scripts/python.exe" "d:/WCE/TY SEM 6/ADS/A/A9/CRUD_GUI.py"
OUTLINE
TIMELINE
Python
Python
0 0
Cloud Code
Connect to Google Cloud
Ln 12, Col 14
Spaces: 4
UTF-8
CRLF
Python
3.11.2 ('myVenv': venv)
Go Live

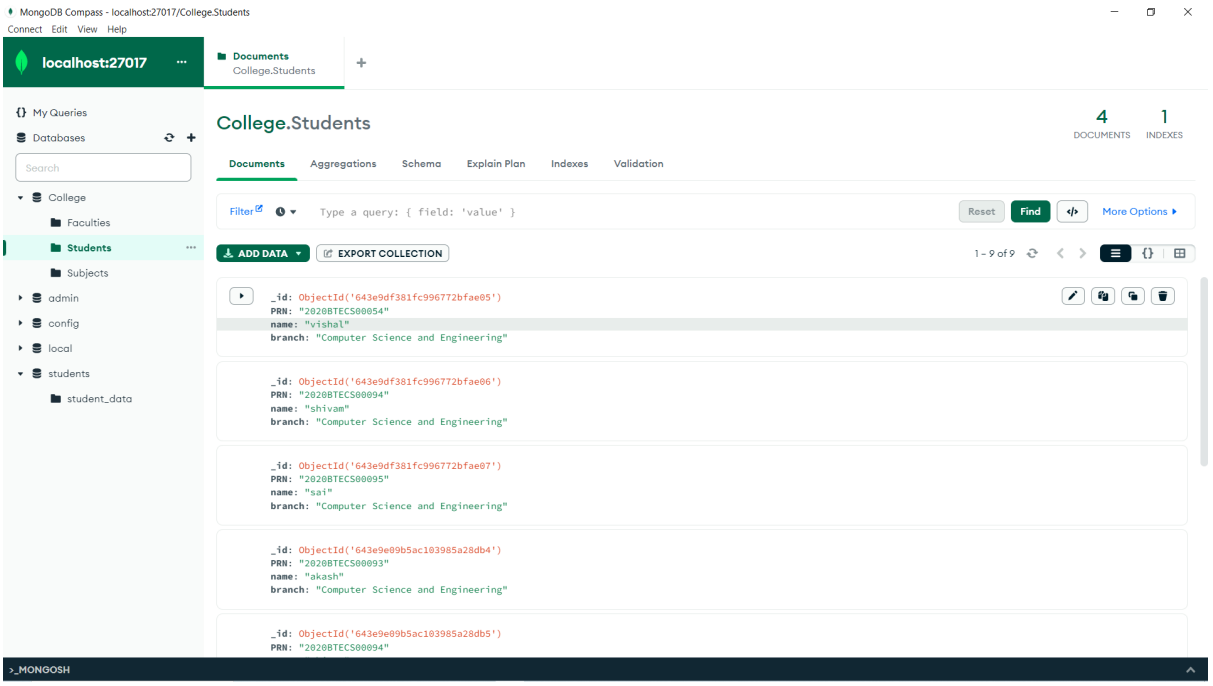





```
import tkinter as tk

from pymongo import MongoClient



client = MongoClient('mongodb://localhost:27017/')
```

```
db = client['College']

student_col = db['Students']



# PRN : 2020BTECS00092

# Name : Vishal Shrirang Madle

# LAB  : ADS LAB : ASSIGNMENT 09



def create():

    prn = prn_entry.get()

    name = name_entry.get()

    branch = branch_entry.get()

    student = {"PRN": prn, "name": name, "branch": branch}

    result = student_col.insert_one(student)

    result_box.insert(tk.END, f"Record with PRN {prn} created successfully.")



def read():

    result_box.delete(0, tk.END)

    for student in student_col.find():

        result_box.insert(tk.END, f"PRN: {student['PRN']} Name:
{student['name']} Branch: {student['branch']}")
```

```
def update():
    old_prn = old_prn_entry.get()
    new_prn = new_prn_entry.get()
    new_name = new_name_entry.get()
    new_branch = new_branch_entry.get()
    student_col.update_one({"PRN": old_prn}, {"$set": {"PRN": new_prn, "name": new_name, "branch": new_branch}})
    result_box.insert(tk.END, f"Record with PRN {old_prn} updated successfully.")


def delete():
    prn = prn_entry.get()
    student_col.delete_one({"PRN": prn})
    result_box.insert(tk.END, f"Record with PRN {prn} deleted successfully.")
```


MongoDB Compass - localhost:27017/College.Students
Connect Edit View Help
localhost:27017
Documents
College.Students
My Queries
Databases
Search
College
Faculties
Students
Subjects
admin
config
local
students
student_data
College.Students
4
DOCUMENTS
1
INDEXES
Documents
Aggregations
Schema
Explain Plan
Indexes
Validation
Filter
Type a query: { field: 'value' }
Reset
Find
More Options
ADD DATA
EXPORT COLLECTION
1 - 9 of 9
_id: ObjectId('643e9df381fc996772bfae05')
PRN: "2020BTECS00054"
name: "vishal"
branch: "Computer Science and Engineering"
_id: ObjectId('643e9df381fc996772bfae06')
PRN: "2020BTECS00094"
name: "shivam"
branch: "Computer Science and Engineering"
_id: ObjectId('643e9df381fc996772bfae07')
PRN: "2020BTECS00095"
name: "sai"
branch: "Computer Science and Engineering"
_id: ObjectId('643e9e09b5ac103985a28db4')
PRN: "2020BTECS00093"
name: "akash"
branch: "Computer Science and Engineering"
_id: ObjectId('643e9e09b5ac103985a28db5')
PRN: "2020BTECS00094"
>_MONGOSH