

# Jaya Vardhan Swarna

The screenshot shows a Jupyter Notebook running on a local server. The interface includes a sidebar with a MongoDB Explorer, a main code editor, and a bottom status bar. The code in the notebook is as follows:

```
import pymongo
client = pymongo.MongoClient("mongodb://localhost:27017/")

db1 = client['ineuron']

coll1=db1['Addmission_details']

import pandas as pd
dataFrame=pd.read_csv("https://raw.githubusercontent.com/vijmeshk/Admission-Dataset/master/Admission.csv")
```

The output of the last cell shows a preview of the loaded data as a table:

|     | Serial No. | GRE Score | TOEFL Score | University Rating | SOP | LOR | CGPA | Research | Chance of Admit |
|-----|------------|-----------|-------------|-------------------|-----|-----|------|----------|-----------------|
| 0   | 1          | 337       | 118         | 4                 | 4.5 | 4.5 | 9.65 | 1        | 0.92            |
| 1   | 2          | 324       | 107         | 4                 | 4.0 | 4.5 | 8.87 | 1        | 0.76            |
| 2   | 3          | 316       | 104         | 3                 | 3.0 | 3.5 | 8.00 | 1        | 0.72            |
| 3   | 4          | 322       | 110         | 3                 | 3.5 | 2.5 | 8.67 | 1        | 0.80            |
| 4   | 5          | 314       | 103         | 2                 | 2.0 | 3.0 | 8.21 | 0        | 0.65            |
| ... | ...        | ...       | ...         | ...               | ... | ... | ...  | ...      | ...             |
| 395 | 396        | 324       | 110         | 3                 | 3.5 | 3.5 | 9.04 | 1        | 0.82            |
| 396 | 397        | 325       | 107         | 3                 | 3.0 | 3.5 | 9.11 | 1        | 0.84            |

The screenshot shows the same Jupyter Notebook interface, but with a different cell selected. The code in this cell is:

```
dict1=dataframe.to_dict("records")
dict1
```

The output shows a list of dictionaries, each representing a row from the dataset. The first few entries are:

```
[{'Serial No.': 1, 'GRE Score': 337, 'TOEFL Score': 118, 'University Rating': 4, 'SOP': 4.5, 'LOR': 4.5, 'CGPA': 9.65, 'Research': 1, 'Chance of Admit': 0.92}, {'Serial No.': 2, 'GRE Score': 324, 'TOEFL Score': 107, 'University Rating': 4, 'SOP': 4.0, 'LOR': 4.5, 'CGPA': 8.87, 'Research': 1, 'Chance of Admit': 0.76}, {'Serial No.': 3, 'GRE Score': 316, 'TOEFL Score': 104, 'University Rating': 3, 'SOP': 3.0, 'LOR': 3.5, 'CGPA': 8.00, 'Research': 1, 'Chance of Admit': 0.72}, {'Serial No.': 4, 'GRE Score': 322, 'TOEFL Score': 110, 'University Rating': 3, 'SOP': 3.5, 'LOR': 2.5, 'CGPA': 8.67, 'Research': 1, 'Chance of Admit': 0.80}, {'Serial No.': 5, 'GRE Score': 314, 'TOEFL Score': 103, 'University Rating': 2, 'SOP': 2.0, 'LOR': 3.0, 'CGPA': 8.21, 'Research': 0, 'Chance of Admit': 0.65}, ...]
```

Microsoft Teams | 24Online Client | MongoDB/main.py at main | Assignment.ipynb - worksp | Jayavardhan-swarna/neu | job guarantee big data bo |

https://purple-actor-eaqlf.neuron.app/folder=/config/workspace

MONGODB CONNECTIONS

- localhost27017 connected
  - admin
  - config
  - neuron
    - Admission\_details
      - Documents 400
      - Schema
      - Indexes
      - local

PLAYGROUNDS

HELP AND FEEDBACK

- What's New
- Extension Documentation
- MongoDB Documentation
- Suggest a Feature
- Report a Bug
- Create Free Atlas Cluster

Assignment.ipynb

Code | Markdown | Run All | Clear Outputs of All Cells | Restart | Interrupt | Variables | Outline

Python 3.8.10 64-bit

```
coll1.insert_many(dict1)
```

[7] ✓ 0.1s

<pymongo.results.InsertManyResult at 0x7f28eeab1e20>

```
result=coll1.find()
for i in result:
    print(i)
```

[8] ✓ 0.1s

Output exceeds the [size limit](#). Open the full output data [in a text editor](#)

```
{'_id': ObjectId('63e49492a178a2160e83f796'), 'Serial No.': 1, 'GRE Score': 337, 'TOEFL Score': 118, 'University Rating': 4, 'SOP': 4.5, 'LOR ': 4.5, 'CGPA': 9.65, 'Research': 1, 'Chance of Admit ': 0.92}
{'_id': ObjectId('63e49492a178a2160e83f797'), 'Serial No.': 2, 'GRE Score': 324, 'TOEFL Score': 107, 'University Rating': 4, 'SOP': 4.0, 'LOR ': 4.5, 'CGPA': 8.87, 'Research': 1, 'Chance of Admit ': 0.76}
{'_id': ObjectId('63e49492a178a2160e83f798'), 'Serial No.': 3, 'GRE Score': 316, 'TOEFL Score': 104, 'University Rating': 3, 'SOP': 3.0, 'LOR ': 3.5, 'CGPA': 8.0, 'Research': 1, 'Chance of Admit ': 0.72}
{'_id': ObjectId('63e49492a178a2160e83f799'), 'Serial No.': 4, 'GRE Score': 322, 'TOEFL Score': 110, 'University Rating': 3, 'SOP': 3.5, 'LOR ': 2.5, 'CGPA': 8.67, 'Research': 1, 'Chance of Admit ': 0.8}
{'_id': ObjectId('63e49492a178a2160e83f79a'), 'Serial No.': 5, 'GRE Score': 314, 'TOEFL Score': 103, 'University Rating': 2, 'SOP': 2.0, 'LOR ': 3.0, 'CGPA': 8.21, 'Research': 0, 'Chance of Admit ': 0.65}
{'_id': ObjectId('63e49492a178a2160e83f79b'), 'Serial No.': 6, 'GRE Score': 330, 'TOEFL Score': 115, 'University Rating': 5, 'SOP': 4.5, 'LOR ': 3.0, 'CGPA': 9.34, 'Research': 1, 'Chance of Admit ': 0.9}
{'_id': ObjectId('63e49492a178a2160e83f79c'), 'Serial No.': 7, 'GRE Score': 321, 'TOEFL Score': 109, 'University Rating': 3, 'SOP': 3.0, 'LOR ': 4.0, 'CGPA': 8.2, 'Research': 1, 'Chance of Admit ': 0.75}
{'_id': ObjectId('63e49492a178a2160e83f79d'), 'Serial No.': 8, 'GRE Score': 308, 'TOEFL Score': 101, 'University Rating': 2, 'SOP': 3.0, 'LOR ': 4.0, 'CGPA': 7.9, 'Research': 0, 'Chance of Admit ': 0.68}
{'_id': ObjectId('63e49492a178a2160e83f79e'), 'Serial No.': 9, 'GRE Score': 302, 'TOEFL Score': 102, 'University Rating': 1, 'SOP': 2.0, 'LOR ':
```

Jupyter Server: Local Cell 7 of 7 Layout: US

12:13 PM 09-02-2023