Lesson 4

RUMAD Accelerator - Backend

Ice Breaker

Homework Review

PUT

- UPDATE items from whatever storage (whether database or list in python)

STEPS:

1. IDENTIFY what to update
2. Specify the LOCATION of item
   a. URL params
3. update
   a. List:
      i. Remove item and then add it back
   b. Dictionary:
      i. [dictionary_name].update([new dictionary])

DELETE

- REMOVE an item from the database

STEPS:

1. IDENTIFY what to delete
2. Specify the LOCATION of item
   a. URL params
3. DELETE using
   a. .remove()
      i. [list_name].remove(‘[element to remove]’)
   b. del
      i. del obj_name
      ii. Key:value pair

Issues with what we’re doing

- Notice when you stop and restart the server
  - What happens?
- Notice we’re doing some loops
  - What happens when we have millions of restaurants?
- Idea: find a way to store stuff permanently

Persistent Data Storage

- Persistent data storage - permanent way to store data such that you don’t lose data when you stop the program or turn off power
- How do we do this?
- Common example - files

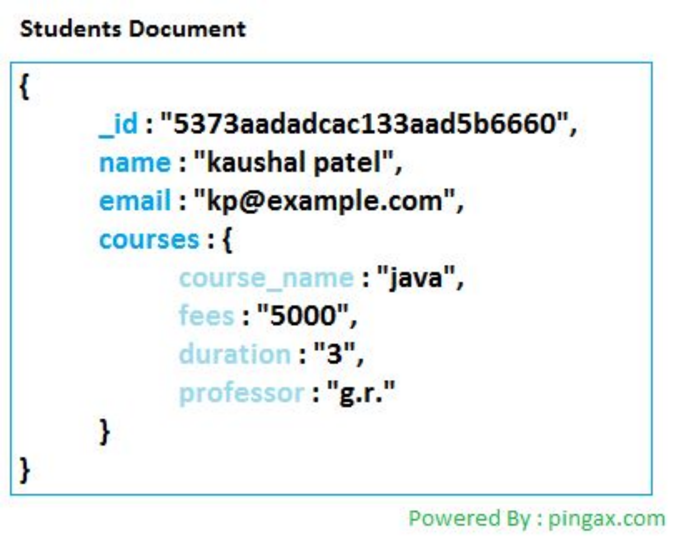
Databases

- Files are useful, but ultimately it's tedious to read and write from files
- Database - a tool whose only job is to permanently store data
- Databases usually provide cool features like
  - Fast lookups - no longer need to use loops
  - Backup - if your computer gets destroyed, it can save a backup somewhere else
  - Much more!

MongoDB

- A database that stores data in a JSON format
- Easy to learn and start, widely used

Document

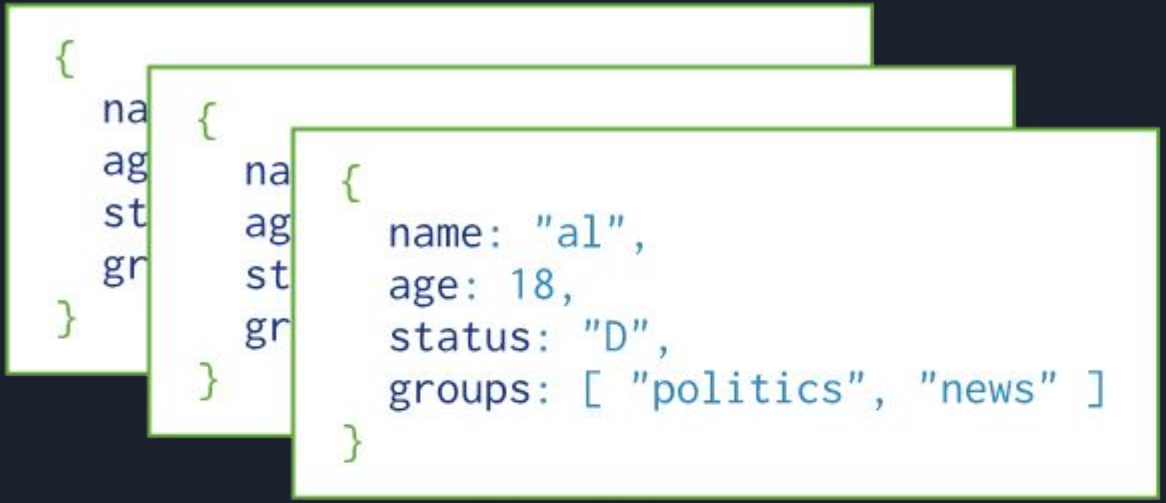


Students Document

```
{
        _id : "5373aadadcac133aad5b6660",
        name : "kaushal patel",
        email : "kp@example.com",
        courses : {
                course_name : "java",
                fees : "5000",
                duration : "3",
                professor : "g.r."
        }
}
```

Powered By : pingax.com

Collection



```
{
   name: "al",
   age: 18,
   status: "D",
   groups: [ "politics", "news" ]
}
```

Collection

Let’s play around with MongoDB!

We need a place to store all the data. Run

Mac: `mkdir -p ~/Documents/data`

Ubuntu: `sudo mkdir -p /data/db`

You only need to run this once. Then,

- For mac:
  ```
  sudo mongod --dbpath ~/Documents/data
  ```

- For ubuntu:
  ```
  sudo mongod
  ```

This command starts mongodb. Now, we can start using it.

Follow along!