

# CMSC335

---

## Web Application Development with JavaScript



## MongoDB

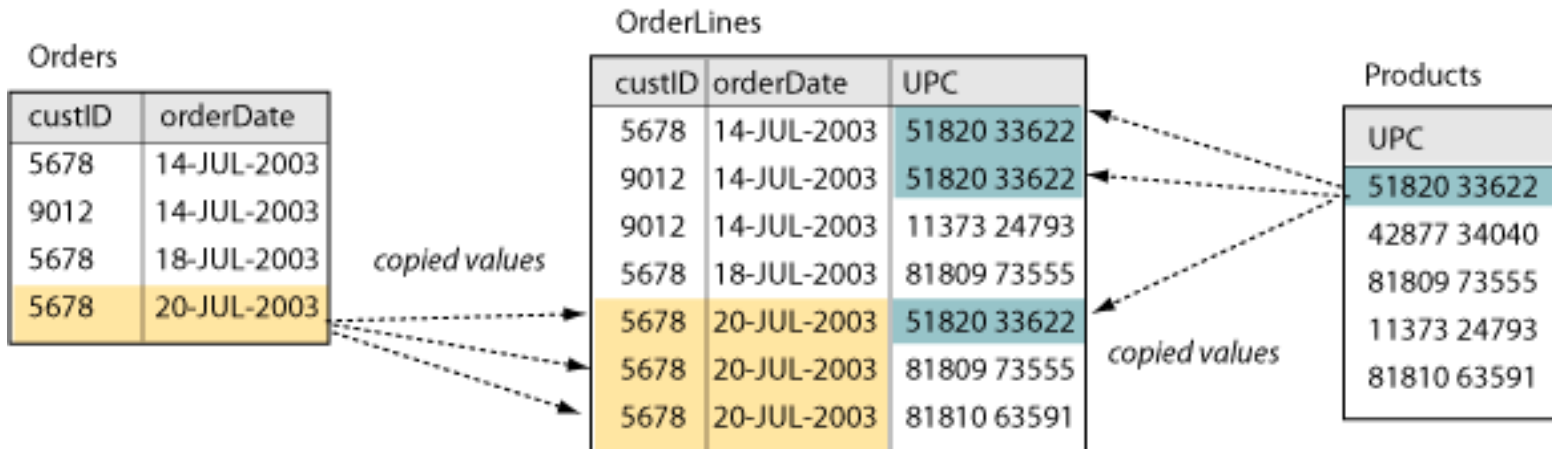
Department of Computer Science  
University of MD, College Park

Slides material developed by Ilchul Yoon, Nelson Padua-Perez

# Databases

---

- An organized collection of data/information
- Relational database (SQL based)(e.g., MySQL)
  - Store data as tables
  - **Structured**, low redundancy of information across tables



# Non-relational database

---

- Called **NoSQL** databases
  - Examples: MongoDB, Cassandra, Redis, ...
- Less structured (or... may say **flexible** data structure)

```
{  
  name: "Pikachu",  
  imageUrl: "https://assets.pokemon.com/assets25.png",  
  stats: {  
    hp: 3,  
    attack: 5,  
    defense: 3,  
    speed: 6  
  }  
}
```

# CRUD operations on Resources

---

- **CRUD** - Create, Read, Update, Destroy (or Delete)
- Typical examples of resources:
  - Data stored in a database
  - Pages
  - Files



- Document database
- A record in mongoDB is a document, which is:
  - a data structure composed of field and value pairs

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

← field: value  
← field: value  
← field: value  
← field: value

- **Stores documents in “Collections”**
  - Similar to tables in relational databases
- We can install mongoDB locally but will use the Cloud version (mongoDB Atlas)
- See: <https://www.cs.umd.edu/~nelson/classes/resources/web/mongodb/>

# dotenv module

---

- The database examples we will see store some username/password information
- We use the **dotenv** module to store this information in a file called .env
- To install the module: **npm i dotenv**
- Using `require("dotenv").config()` will allow access of environment variables via **`process.env.VARIABLE_NAME`**. You can define a folder with a .env file and use it as a parameter to `.config()` as shown in the examples

# Examples

---

- **Examples:** DatabaseExamples
  - First, make sure you have a cloud.mongodb.com account and can access “Atlas” section
  - Update the file credentialsDontPost/.env in the **DatabaseExamples** folder with the username and password corresponding to a database user. In Atlas, you can add a new database user by selecting under the “Security” title, “Database Access.”, There is a “+ ADD NEW DATABASE USER” option on that page
  - To see databases, select “Database” under the “DEPLOYMENT” title and then click on the “Browse Collections” buttons

# Examples

---

- You can create a database by selecting the “+Create Database” button. For the examples below, you don’t need to create a database as a database called “CMSC335DB” and a collection called “moviesCollection” will be created when you run the example **node .\insertMovies.js**. **Run this example, before you run any other example**
- After running the examples, you can **drop** (remove) the CMSC335DB database by selecting the trash bin icon you see when you hover over CMSC335DB
- The following examples illustrate CRUD operations. They are recipes that you can use for your project(s)
- **Examples:** insertMovies.js, listDatabase.js, listAllMovies.js, lookUpMovies.js, updateMovie.js deleteMovies.js, clearCollection.js



# References

---

- <https://www.youtube.com/watch?v=fbYExfeFsI0&t=1815s>
- <https://www.youtube.com/watch?v=5WFyhsnU4Ik>