ONLINE
TO-DO LIST
Abdelhameed-Karim-Michael

Outline

- Introduction
- Topology
- Implementation
- Final Application

Web Service

- Is any piece of software that makes itself available over the internet and uses a standardized messaging system.
- Modern web browsers and servers implement all features in REST.

REST
REpresentational State Transfer

- Rest is a software architecture style
- The Uniform interface is implemented by HTTP protocol
- REST is characterized by:
  - Stateless Server
  - Uniform Interface
  - Resource is send in the message Body

REST Features

- Resource: Any named concept.
- Resource identification: URI.
- Resource representation: Document, data.
- Messages: HTTP messages.
- Control data in messages: HTTP method (GET, POST, DELETE).
- Resource in message: Data in message body.

Components

MYSQL DATABASE

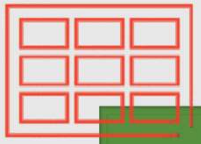
JAVASCRIPT SERVER

PYTHON CLIENT

MySQL Database

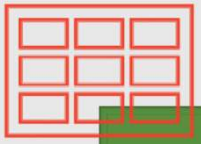
	Table	Action						
☐	tasks	☆	Browse	Structure	Search	Insert	Empty	Drop
☐	users	☆	Browse	Structure	Search	Insert	Empty	Drop
	2 tables	Sum						

MySQL Database



Users Table

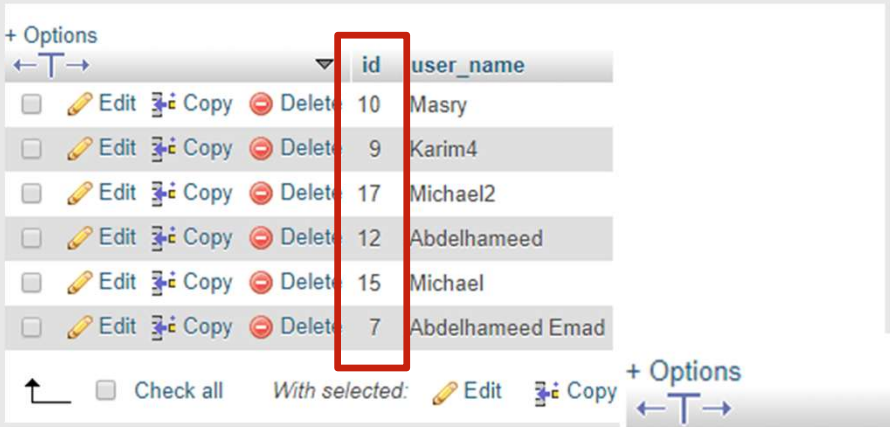
- User ID
- User Unique Name

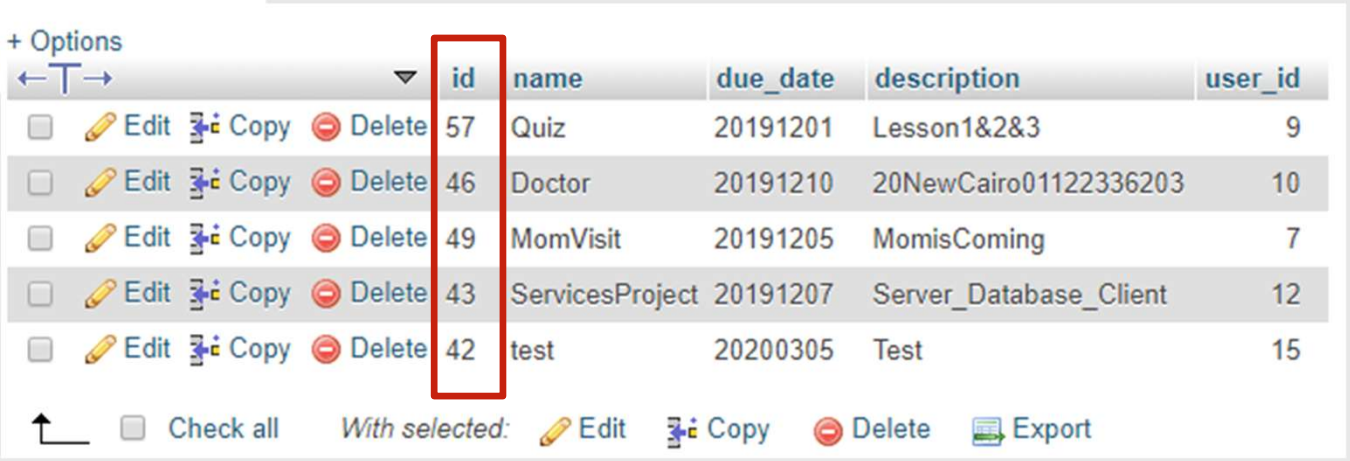


Tasks Table

- Unique ID
- Task Name
- Description
- Due Date
- Task Owner

MySQL Database

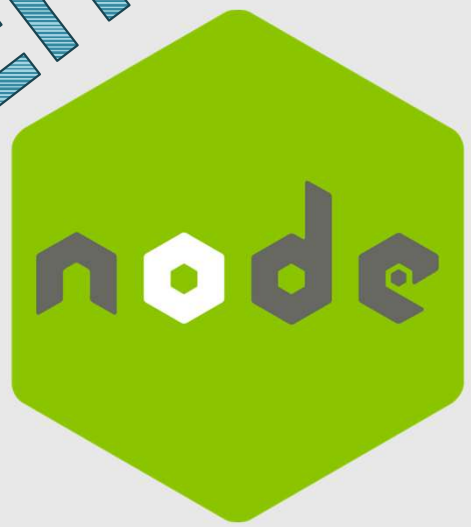




Server

Express JS

Back End



Server

Sign Up and Authentication

Create or Add Tasks

Delete or Remove Tasks

Display all Related Tasks

Database Integrity

Server

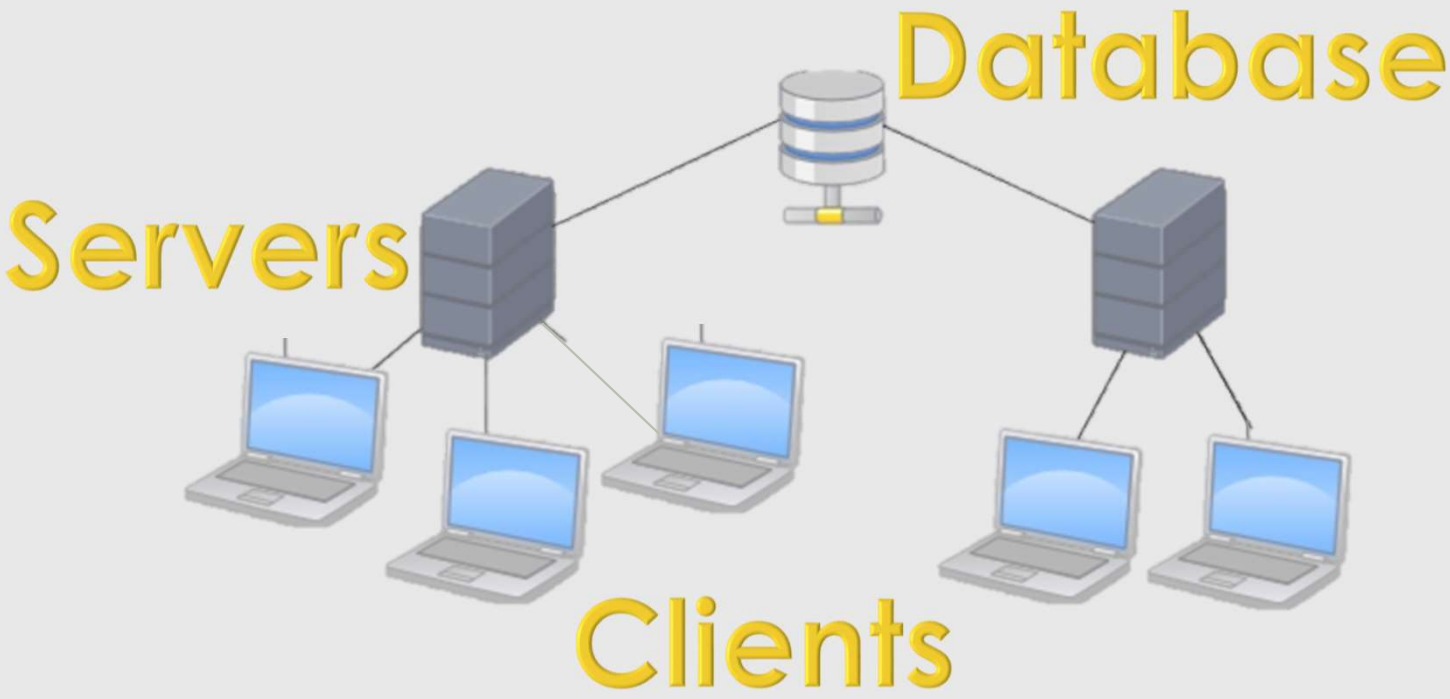
```
// Create task
app.post('/addnewtask', (req, res) => {
    const task_name = req.query.name
    const task_date = req.query.date
    const task_description = req.query.description
    const user_id = req.query.user_id
```

```
// Delete task
app.delete('/deletetask', (req, res) => {
    // Get all the user's tasks
    app.get('/allmytasks', (req, res) => {
        const id = req.query.user_id
        let sql = `SELECT name, due_date, description, id FROM tasks WHERE user_id=${id};`;
        let query = db.query(sql, (err, results) => {
            if(err){
                console.log(err)
                throw err;
            }
            console.log(results);
            res.send(JSON.stringify(results));
        });
    });
});
```

Python Client

Python: Tkinter

Python Client



Python Client

Server & Client

http://127.0.0.1:3000/signup?username=Michael

http://127.0.0.1:3000/allmytasks?user_id=8

http://127.0.0.1:3000/deletetask?id=2

http://127.0.0.1:3000/addnewtask?name=servicesHW&date=20191210&description=homework33&user_id=8

REST ???