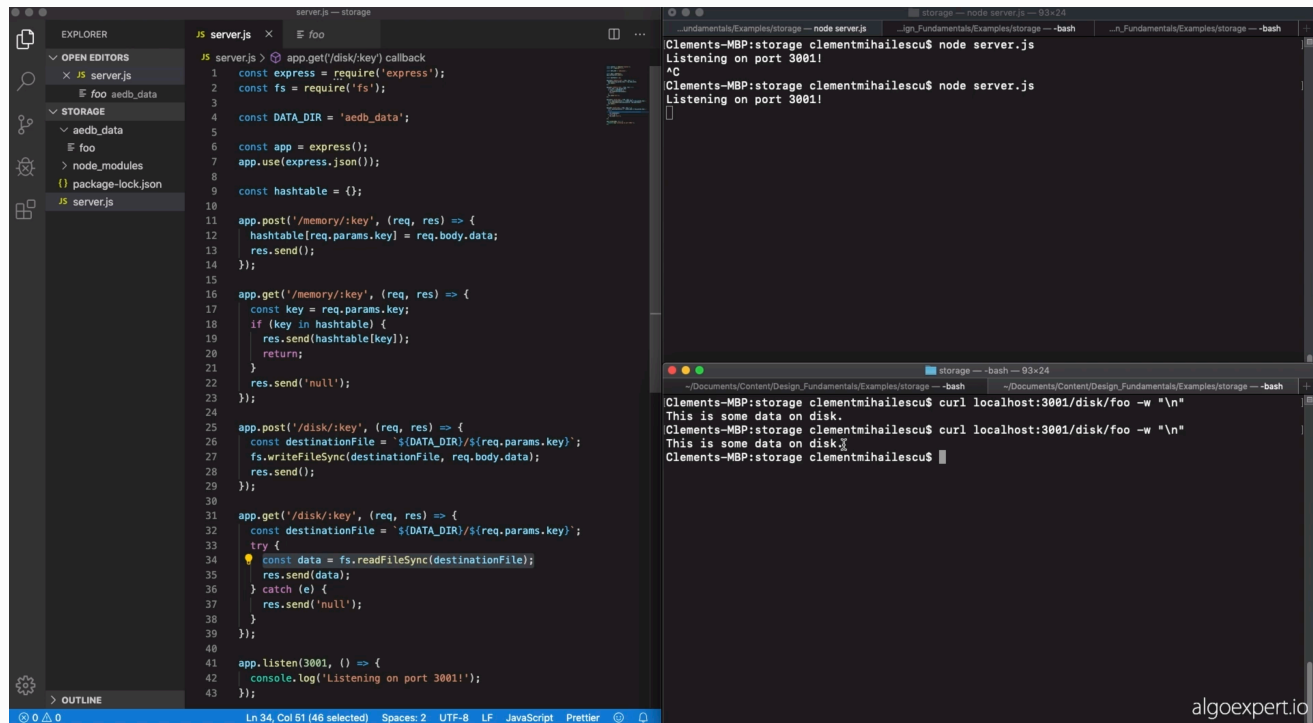


3/20/2021

Storage



The screenshot shows a code editor with a file named `server.js` and a terminal window. The `server.js` file contains the following code:

```
1 const express = require('express');
2 const fs = require('fs');
3
4 const DATA_DIR = 'aedb_data';
5
6 const app = express();
7 app.use(express.json());
8
9 const hashtable = {};
10
11 app.post('/memory/:key', (req, res) => {
12   hashtable[req.params.key] = req.body.data;
13   res.send();
14 });
15
16 app.get('/memory/:key', (req, res) => {
17   const key = req.params.key;
18   if (key in hashtable) {
19     res.send(hashtable[key]);
20   }
21   res.send('null');
22 });
23
24 app.post('/disk/:key', (req, res) => {
25   const destinationFile = `${DATA_DIR}/${req.params.key}`;
26   fs.writeFileSync(destinationFile, req.body.data);
27   res.send();
28 });
29
30 app.get('/disk/:key', (req, res) => {
31   const destinationFile = `${DATA_DIR}/${req.params.key}`;
32   try {
33     const data = fs.readFileSync(destinationFile);
34     res.send(data);
35   } catch (e) {
36     res.send('null');
37   }
38 });
39
40 app.listen(3001, () => {
41   console.log('Listening on port 3001!');
42 });
```

The terminal window shows the following commands and output:

```
Clements-MBP:storage clementmihailescuc$ node server.js
Listening on port 3001!
^C
Clements-MBP:storage clementmihailescuc$ node server.js
Listening on port 3001!

Clements-MBP:storage clementmihailescuc$ curl localhost:3001/disk/foo -w "%n"
This is some data on disk.
Clements-MBP:storage clementmihailescuc$ curl localhost:3001/disk/foo -w "%n"
This is some data on disk.
Clements-MBP:storage clementmihailescuc$
```

4 Key Terms

Databases

Databases are programs that either use disk or memory to do 2 core things: **record** data and **query** data. In general, they are themselves servers that are long lived and interact with the rest of your application through network calls, with protocols on top of TCP or even HTTP.

Some databases only keep records in memory, and the users of such databases are aware of the fact that those records may be lost forever if the machine or process dies.

For the most part though, databases need persistence of those records, and thus cannot use memory. This means that you have to write your data to disk. Anything written to disk will remain through power loss or network partitions, so that's what is used to keep permanent records.

Since machines die often in a large scale system, special disk partitions or volumes are used by the database processes, and those volumes can get recovered even if the machine were to go down permanently.

Disk

Usually refers to either **HDD (hard-disk drive)** or **SSD (solid-state drive)**. Data written to disk will persist through power failures and general machine crashes. Disk is also referred to as **non-volatile storage**.

SSD is far faster than HDD (see latencies of accessing data from SSD and HDD) but also far more expensive from a financial point of view. Because of that, HDD will typically be used for data that's rarely accessed or updated, but that's stored for a long time, and SSD will be used for data that's frequently accessed and updated.

Memory

Short for **Random Access Memory (RAM)**. Data stored in memory will be lost when the process that has written that data dies.

Persistent Storage

Usually refers to disk, but in general it is any form of storage that persists if the process in charge of managing it dies.

Latency and Throughput

2 Key Terms

Latency

The time it takes for a certain operation to complete in a system. Most often this measure is a time duration, like milliseconds or seconds. You should know these orders of magnitude:

- **Reading 1 MB from RAM:** 250 μ s (0.25 ms)
- **Reading 1 MB from SSD:** 1,000 μ s (1 ms)
- **Transfer 1 MB over Network:** 10,000 μ s (10 ms)
- **Reading 1MB from HDD:** 20,000 μ s (20 ms)
- **Inter-Continental Round Trip:** 150,000 μ s (150 ms)

Throughput

The number of operations that a system can handle properly per time unit. For instance the throughput of a server can often be measured in requests per second (RPS or QPS).