Name : Avinash
Roll number : 21011102019

Node js assignment

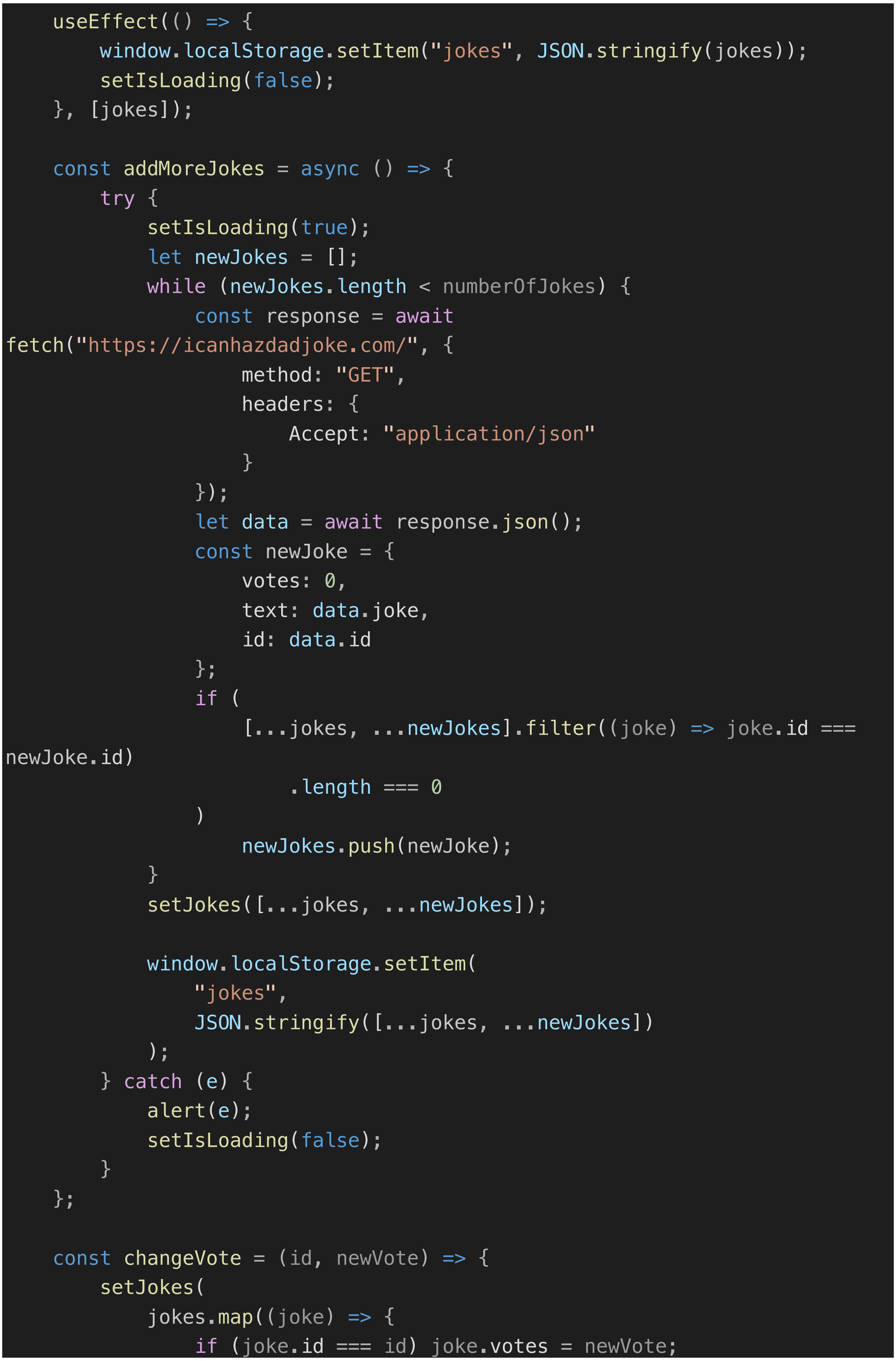
Source code:

Index.js

```
import React from "react";
import ReactDOM from "react-dom";
import "./styles.css";
import Jokes from "./jokes";

function App() {
  return (
    <div className="App">
      <Jokes />

    </div>
  );
}

const rootElement = document.getElementById("root");
ReactDOM.render(<App />, rootElement);
```

Jokes.js

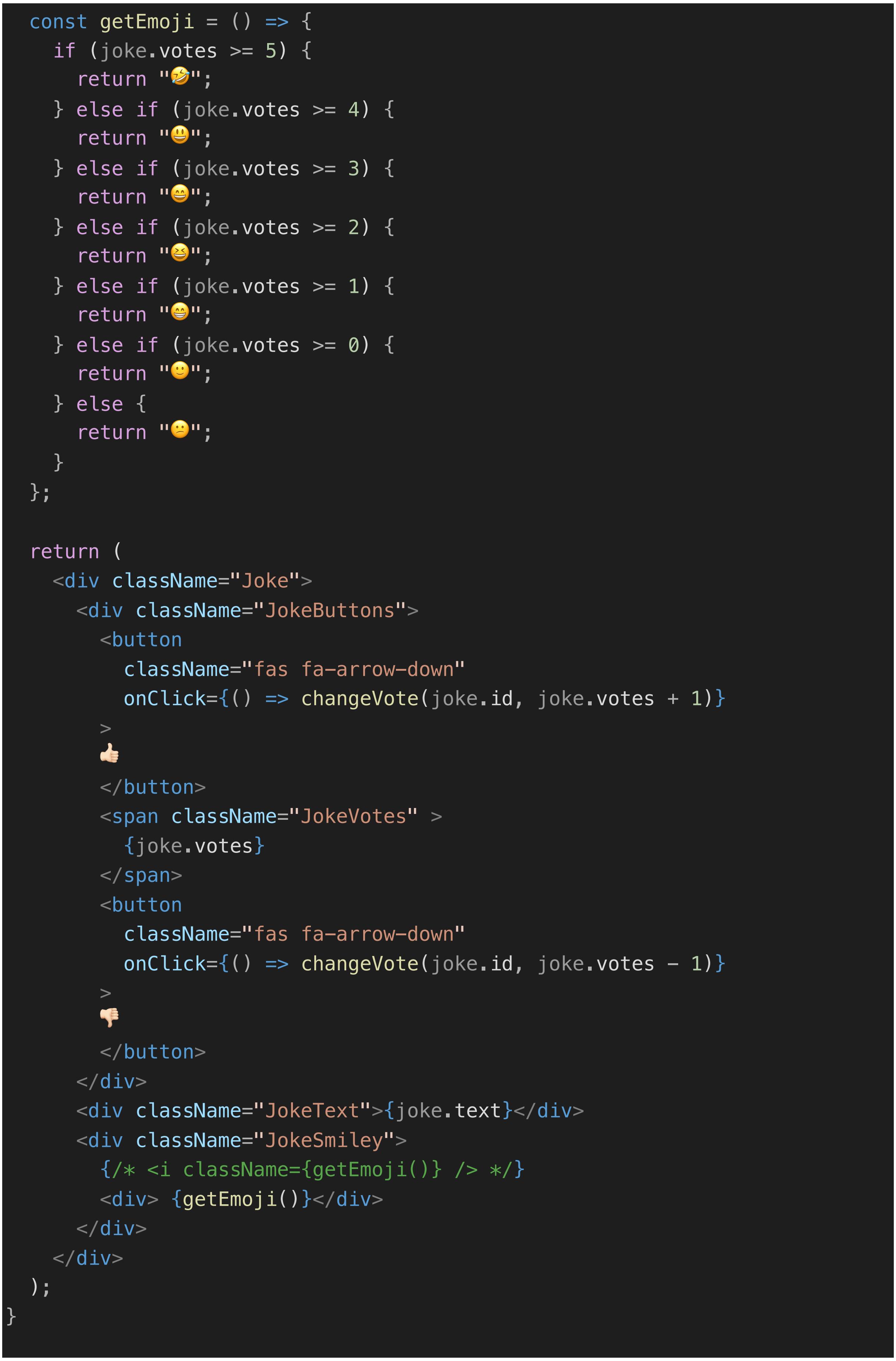
```
import React, { useEffect, useState } from "react";
import "./Jokes.css";
import Joke from "./joke";

Jokes.defaultProps = {
    numberOfJokes: 1
};

function Jokes({ numberOfJokes }) {
    const [jokes, setJokes] = useState([]);
    const [isLoading, setIsLoading] = useState(false);

    useEffect(() => {
        let currentJokes =
JSON.parse(window.localStorage.getItem("jokes")) || [];
        if (currentJokes.length === 0) addMoreJokes();
        else setJokes(currentJokes);
    }, []);

```



```
    useEffect(() => {
        window.localStorage.setItem("jokes", JSON.stringify(jokes));
        setIsLoading(false);
    }, [jokes]);

    const addMoreJokes = async () => {
        try {
            setIsLoading(true);
            let newJokes = [];
            while (newJokes.length < numberOfJokes) {
                const response = await
fetch("https://icanhazdadjoke.com/", {
                    method: "GET",
                    headers: {
                        Accept: "application/json"
                    }
                });
                let data = await response.json();
                const newJoke = {
                    votes: 0,
                    text: data.joke,
                    id: data.id
                };
                if (
                    [...jokes, ...newJokes].filter((joke) => joke.id ===
newJoke.id)
                        .length === 0
                )
                    newJokes.push(newJoke);
            }
            setJokes([...jokes, ...newJokes]);

            window.localStorage.setItem(
                "jokes",
                JSON.stringify([...jokes, ...newJokes])
            );
        } catch (e) {
            alert(e);
            setIsLoading(false);
        }
    };

    const changeVote = (id, newVote) => {
        setJokes(
            jokes.map((joke) => {
                if (joke.id === id) joke.votes = newVote;
        return joke;
      })
    );
  };

  const sortedJokes = jokes.sort((a, b) => b.votes - a.votes);

  if (isLoading) {
    return (
      <div className="JokeListSpinner">
        {/* <i className="far fa-8x fa-laugh fa-spin" /> */}
        <div className="far fa-8x fa-laugh fa-spin">😵‍💫</div>
        <h1 className="JokeListTitle">Loading...</h1>
      </div>
    );
  }

  return (
    <div className="Jokes">
      <div className="JokesSidebar">
        <h1 className="JokeListTitle">
          <span>Dad</span> Jokes
        </h1>
        <button className="GetMoreButton" onClick={addMoreJokes}>
          Fetch Jokes
        </button>
      </div>
      <div className="JokeList">
        {sortedJokes.map((joke) => (
          <Joke key={joke.id} joke={joke}
changeVote={changeVote} />
        ))}
      </div>
    </div>
  );
}

export default Jokes;
```

joke.js :

```
import React from "react";
import "./joke.css";

function Joke({ joke, changeVote }) {
```



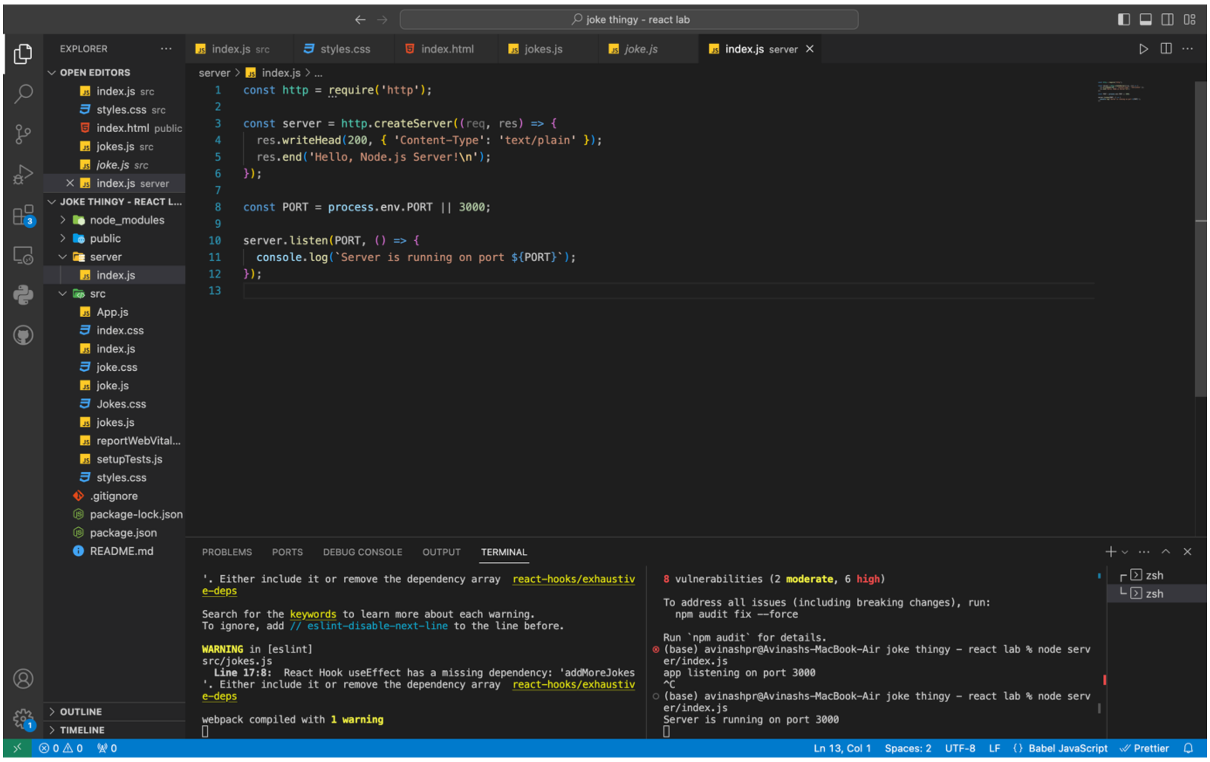
```
  const getEmoji = () => {
    if (joke.votes >= 5) {
      return "🤣";
    } else if (joke.votes >= 4) {
      return "😃";
    } else if (joke.votes >= 3) {
      return "😄";
    } else if (joke.votes >= 2) {
      return "😆";
    } else if (joke.votes >= 1) {
      return "😁";
    } else if (joke.votes >= 0) {
      return "🙂";
    } else {
      return "😕";
    }
  };

  return (
    <div className="Joke">
      <div className="JokeButtons">
        <button
          className="fas fa-arrow-down"
          onClick={() => changeVote(joke.id, joke.votes + 1)}
        >
          👍🏻
        </button>
        <span className="JokeVotes" >
          {joke.votes}
        </span>
        <button
          className="fas fa-arrow-down"
          onClick={() => changeVote(joke.id, joke.votes - 1)}
        >
          👎🏻
        </button>
      </div>
      <div className="JokeText">{joke.text}</div>
      <div className="JokeSmiley">
        {/* <i className={getEmoji()} /> */}
        <div> {getEmoji()}</div>
      </div>
    </div>
  );
}
```

```
export default Joke;
```

server/index.js:

```
const http = require('http');

const server = http.createServer((req, res) => {
  res.writeHead(200, { 'Content-Type': 'text/plain' });
  res.end('Hello, Node.js Server!\n');
});

const PORT = process.env.PORT || 3000;

server.listen(PORT, () => {
  console.log(`Server is running on port ${PORT}`);
});
```

Output in Vs code terminal:

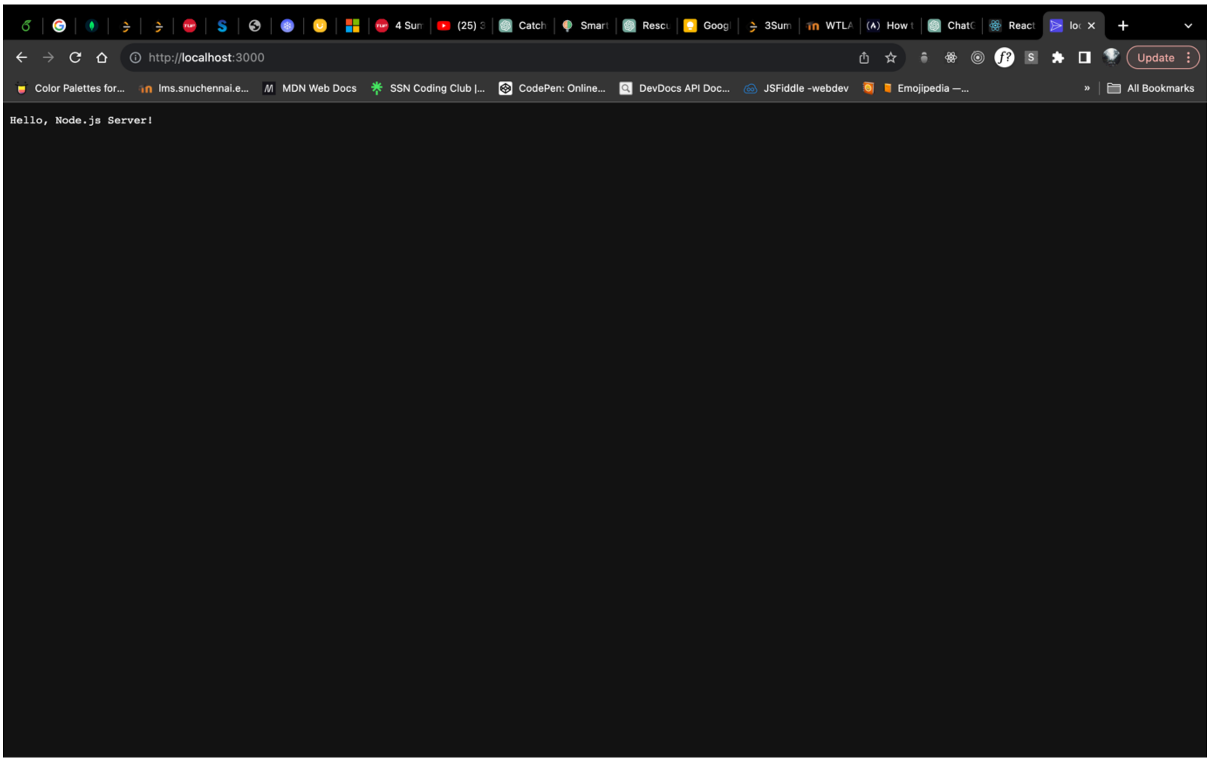


```
const http = require('http');

const server = http.createServer((req, res) => {
  res.writeHead(200, { 'Content-Type': 'text/plain' });
  res.end('Hello, Node.js Server!\n');
});

const PORT = process.env.PORT || 3000;

server.listen(PORT, () => {
  console.log(`Server is running on port ${PORT}`);
});
```

```
'. Either include it or remove the dependency array  react-hooks/exhaustiv
e-deps

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.

WARNING in [eslint]
src/jokes.js
  Line 17:8:  React Hook useEffect has a missing dependency: 'addMoreJokes
'. Either include it or remove the dependency array  react-hooks/exhaustiv
e-deps

webpack compiled with 1 warning
```

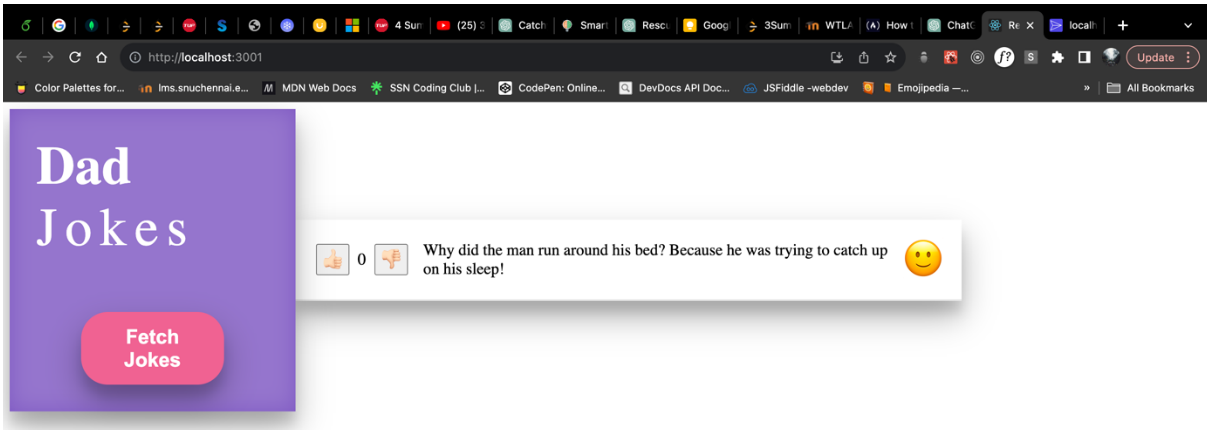
```
8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
(base) avinashpr@Avinashs-MacBook-Air joke thingy - react lab % node serv
er/index.js
app listening on port 3000
^C
(base) avinashpr@Avinashs-MacBook-Air joke thingy - react lab % node serv
er/index.js
Server is running on port 3000
```

Outut in localhost:3000



```
Hello, Node.js Server!
```

Output in localhost:3001



Why did the man run around his bed? Because he was trying to catch up on his sleep!