Lesson 07 Demo 03

Node.js GPT Prompt Evaluator

Objective: To showcase Node.js's capability in evaluating ChatGPT-3 responses to diverse prompts

Tools required: Node.js

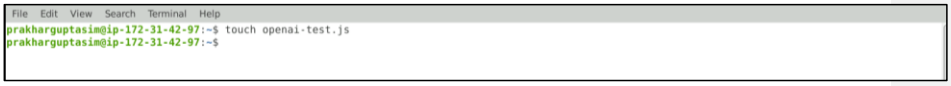
Prerequisites: Before performing this demo, perform second demo of this lesson

Steps to be followed:

1. List some benefits of Agile
2. Generate a React code for a homepage of a website
3. Create an Express.js web service

Step 1: List some benefits of Agile

1.1 Create a **.js** file named as **openai-test.js** using the command **touch openai-test.js**



1.2 Now, open the file using the command **vi openai-test.js** and paste the below code:

```
const OpenAI = require("openai");

 const openai = new OpenAI({
 apiKey: "YOUR API KEY"
 });

 const openFun=async()=>{
 const chatCompletion = await openai.chat.completions.create({
 model: "gpt-3.5-turbo",
 messages: [{"role": "user", "content": "List down the benefits of Agile",}],
 max_tokens:100
 });
 console.log(chatCompletion.choices[0].message.content);
 }
```

Note: In the above code, replace **YOUR API KEY** with the key generated in step 2.3 of the previous demo.

```
File  Edit  View  Search  Terminal  Help
const OpenAI = require("openai");


const openai = new OpenAI({

  apiKey: "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"

});


const openFun=async()=>{

const chatCompletion = await openai.chat.completions.create({

    model: "gpt-3.5-turbo",

    messages: [{"role": "user", "content": "list the benefits of Agile",}],

    max_tokens:1000

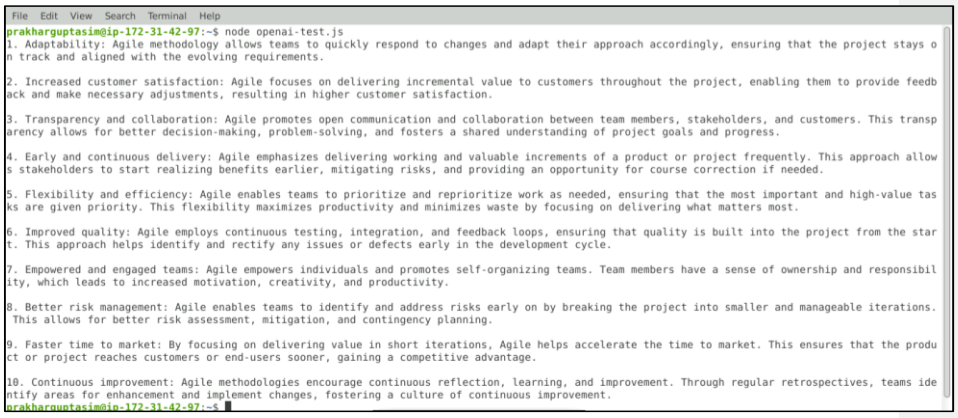
  });

  console.log(chatCompletion.choices[0].message.content);

}

-- INSERT --                                                    19,50         Top
```

1.3 Now, run the **.js** file by using the command **node openai.test.js** to get a response



```
File  Edit  View  Search  Terminal  Help
prakharguptasim@ip-172-31-42-97:~$ node openai-test.js
1. Adaptability: Agile methodology allows teams to quickly respond to changes and adapt their approach accordingly, ensuring that the project stays o
n track and aligned with the evolving requirements.

2. Increased customer satisfaction: Agile focuses on delivering incremental value to customers throughout the project, enabling them to provide feedb
ack and make necessary adjustments, resulting in higher customer satisfaction.

3. Transparency and collaboration: Agile promotes open communication and collaboration between team members, stakeholders, and customers. This transp
arency allows for better decision-making, problem-solving, and fosters a shared understanding of project goals and progress.

4. Early and continuous delivery: Agile emphasizes delivering working and valuable increments of a product or project frequently. This approach allow
s stakeholders to start realizing benefits earlier, mitigating risks, and providing an opportunity for course correction if needed.

5. Flexibility and efficiency: Agile enables teams to prioritize and reprioritize work as needed, ensuring that the most important and high-value tas
ks are given priority. This flexibility maximizes productivity and minimizes waste by focusing on delivering what matters most.

6. Improved quality: Agile employs continuous testing, integration, and feedback loops, ensuring that quality is built into the project from the star
t. This approach helps identify and rectify any issues or defects early in the development cycle.

7. Empowered and engaged teams: Agile empowers individuals and promotes self-organizing teams. Team members have a sense of ownership and responsibil
ity, which leads to increased motivation, creativity, and productivity.

8. Better risk management: Agile enables teams to identify and address risks early on by breaking the project into smaller and manageable iterations.
 This allows for better risk assessment, mitigation, and contingency planning.

9. Faster time to market: By focusing on delivering value in short iterations, Agile helps accelerate the time to market. This ensures that the produ
ct or project reaches customers or end-users sooner, gaining a competitive advantage.

10. Continuous improvement: Agile methodologies encourage continuous reflection, learning, and improvement. Through regular retrospectives, teams ide
ntify areas for enhancement and implement changes, fostering a culture of continuous improvement.
prakharguptasim@ip-172-31-42-97:~$
```

Step 2: Generate a React code for a homepage of a website

2.1 Update the **.js** file created in step 1.1 with the below code:

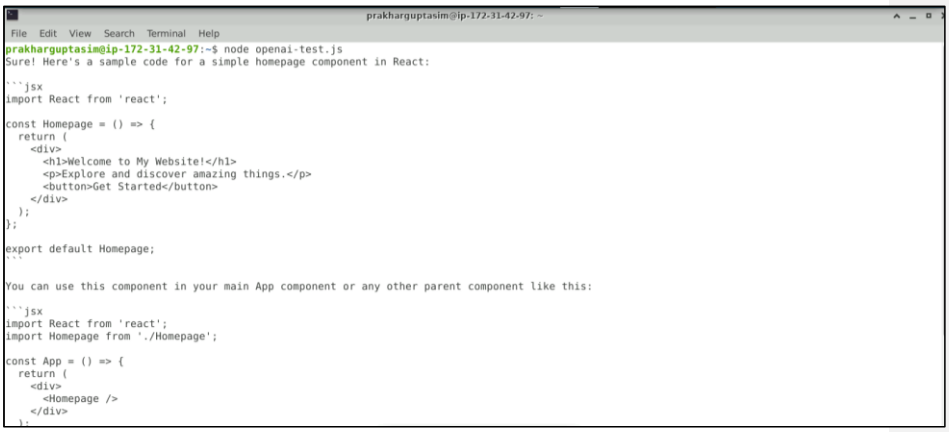
```
const OpenAI = require("openai");

const openai = new OpenAI({
apiKey: "YOUR API KEY"
});

const openFun=async()=>{
const chatCompletion = await openai.chat.completions.create({
model: "gpt-3.5-turbo",
messages: [{"role": "user", "content": "Give a sample react code for website
          homepage",}],
max_tokens:100
});
console.log(chatCompletion.choices[0].message.content);
}
```

Note: In the above code, replace **YOUR API KEY** with the key generated in step 2.3 of previous demo.

2.3 Now, run the **.js** file by using the command **node openai.test.js** to get a response



Step 3: Create an Express.js web service

3.1 Update the **.js** file created in step 1.1 with the below code:

```
const OpenAI = require("openai");

const openai = new OpenAI({
apiKey: "YOUR API KEY"
});

const openFun=async()=>{
const chatCompletion = await openai.chat.completions.create({
model: "gpt-3.5-turbo",
messages: [{"role": "user", "content": "create a webservice using express.js",}],
max_tokens:100
});
console.log(chatCompletion.choices[0].message.content);}
```

Note: In the above code, replace **YOUR API KEY** with the key generated in step 2.3 of previous demo.

```
File  Edit  View  Search  Terminal  Help
const OpenAI = require("openai");


const openai = new OpenAI({

  apiKey: "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"

});


const openFun=async()=>{

const chatCompletion = await openai.chat.completions.create({

    model: "gpt-3.5-turbo",

    messages: [{"role": "user", "content": "create a webservice using express.js",}],

    max_tokens:1000

  });

  console.log(chatCompletion.choices[0].message.content);

}
```

3.2 Open the file by using the command **vi openai-test.js**

```
File  Edit  View  Search  Terminal  Help
prakharguptasim@ip-172-31-42-97:~$ node openai-test.js
Sure! Here is an example of how you can create a basic web service using Express.js.

First, make sure you have Node.js and NPM installed on your machine. Then, create a new directory for your project and navigate into it:

```
mkdir express-webservice
cd express-webservice
```

Now, initialize a new Node.js project and install Express:

```
npm init -y
npm install express
```

Next, create a file called `index.js` and open it in your favorite text editor. Add the following code:

```javascript
const express = require('express');
const app = express();

// Define a route for the web service
app.get('/api/greeting', (req, res) => {
  const name = req.query.name || 'World';
  res.json({ message: `Hello, ${name}!` });
});

// Start the server
const port = 3000;
```

By following the above steps, you have successfully evaluated the ChatGPT-3 responses against the various prompts provided by you.

Commented [RT1]: Objective says GPT responses- Ensure the conclusion matches the objective statement

Commented [VR2R1]: done

Commented [RT3]: Should this be 'responses'?

Commented [VR4R3]: done
```