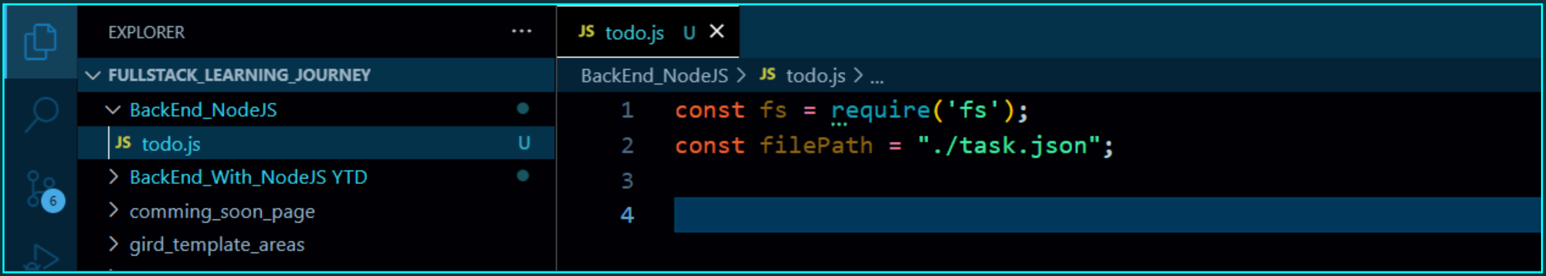
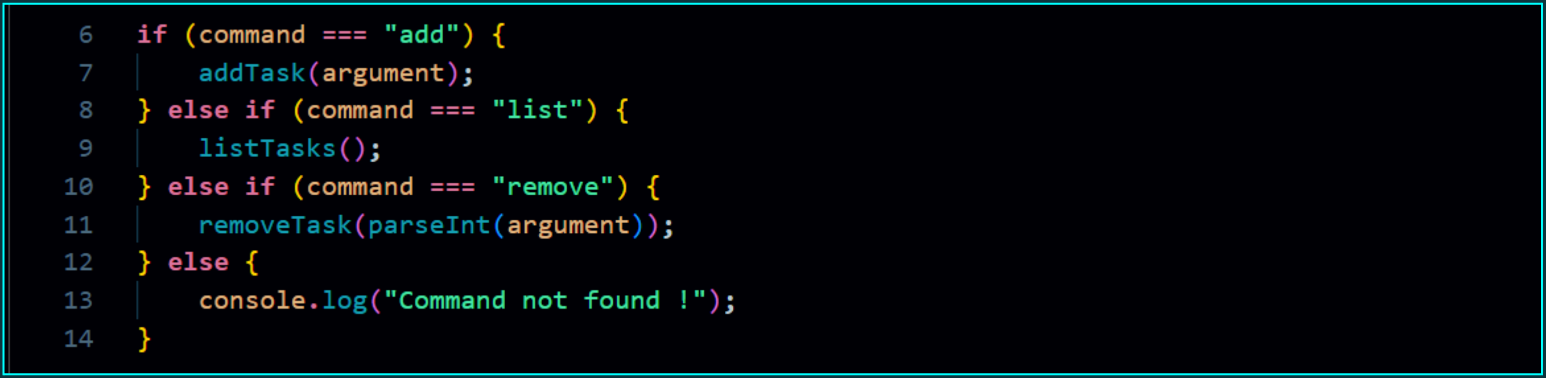
1. **Create a folder and a file todo.js**
2. **Add fs module**



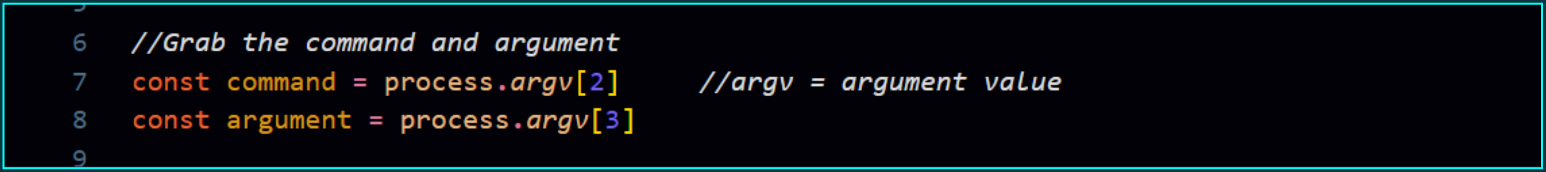
```javascript
const fs = require('fs');
const filePath = "./task.json";

```

3. **Checking certain commands**



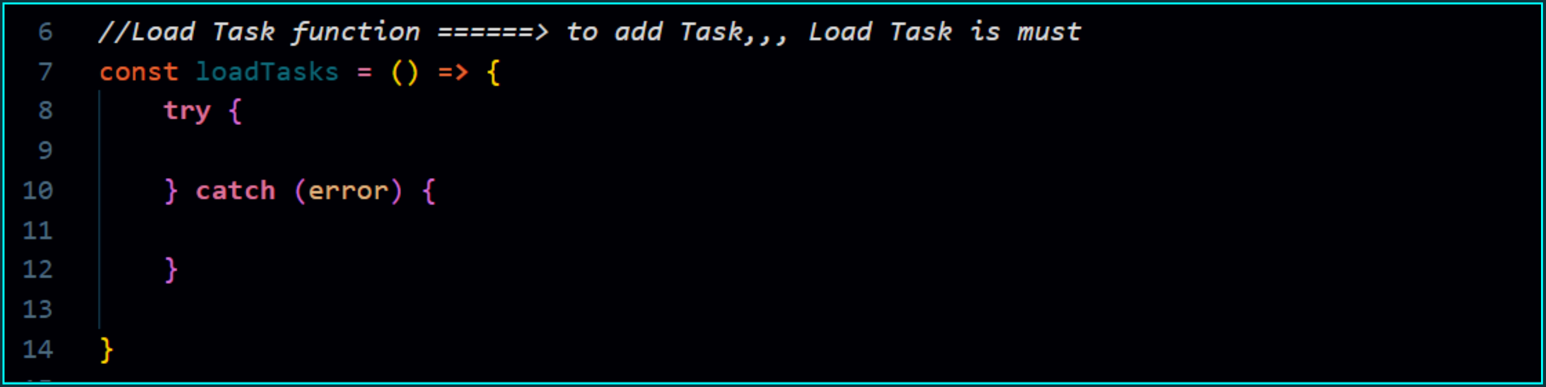
```javascript
if (command === "add") {
    addTask(argument);
} else if (command === "list") {
    listTasks();
} else if (command === "remove") {
    removeTask(parseInt(argument));
} else {
    console.log("Command not found !");
}
```

4. **Grab the command & pass the argument**



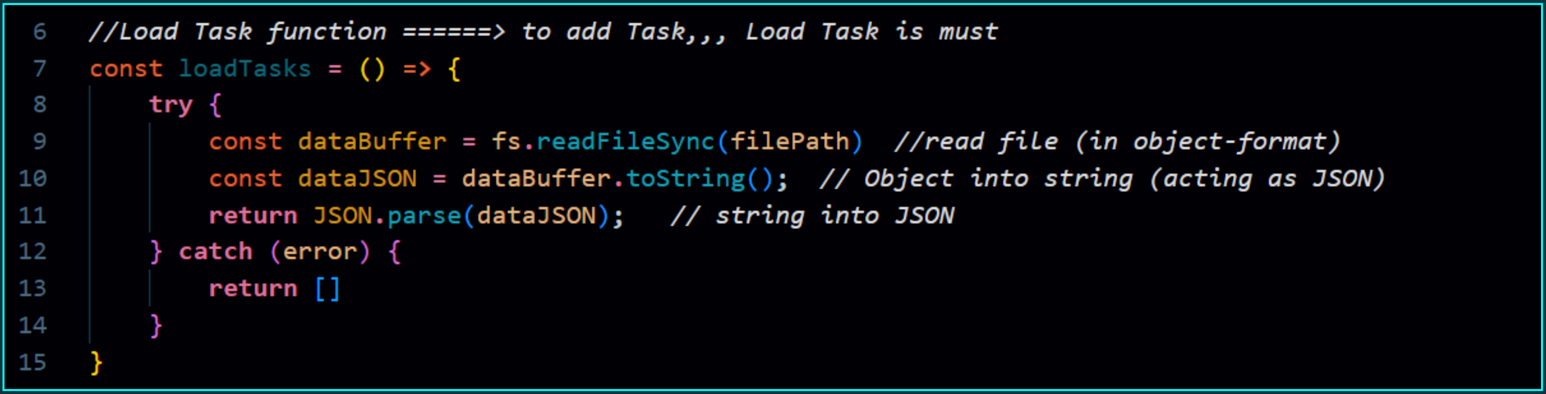
```javascript
//Grab the command and argument
const command = process.argv[2]     //argv = argument value
const argument = process.argv[3]
```

5. **Creating a function ⇒ loadTasks()** *[to add task, load-Task is required]*
[inside it, file-reading is required] ——-- ⇒ [reading may be⇒ succeed or fail]



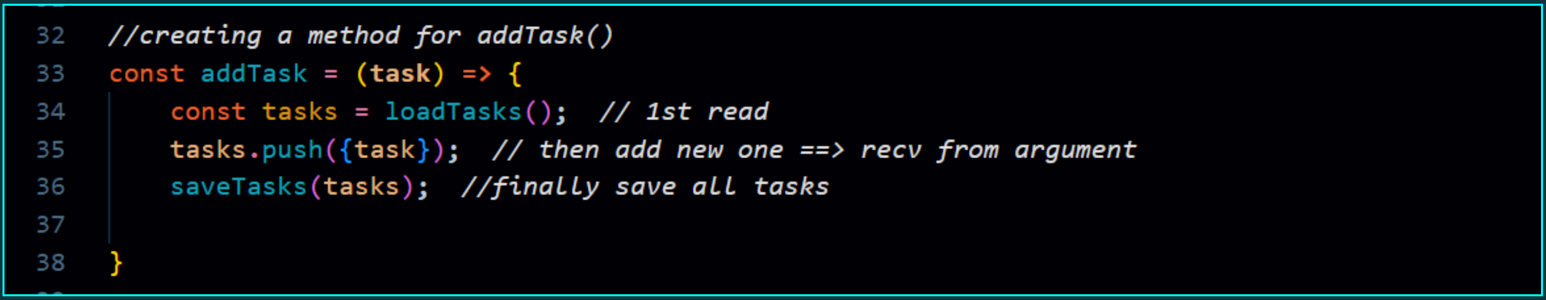
```javascript
//Load Task function ======> to add Task,,, Load Task is must
const loadTasks = () => {
    try {

    } catch (error) {

    }

}
```

6. In try: create loadTask() —-- readfile as Object ⇒ convert to String ⇒ convert to JSON



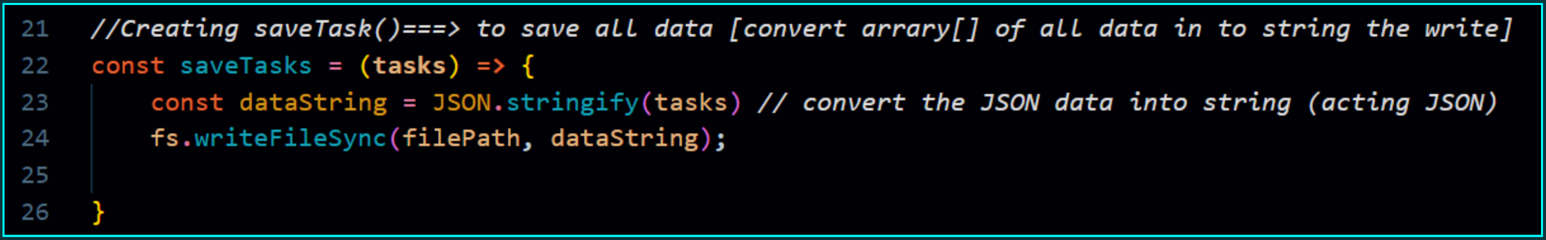
```javascript
//Load Task function ======> to add Task,,, Load Task is must
const loadTasks = () => {
    try {
        const dataBuffer = fs.readFileSync(filePath)  //read file (in object-format)
        const dataJSON = dataBuffer.toString();  // Object into string (acting as JSON)
        return JSON.parse(dataJSON);   // string into JSON
    } catch (error) {
        return []
    }
}
```

7. Creating Method addTask() ====> adding new data to the array
(to add tasks,,, first read task {loadTask() } & then push new tasks) then⇒ save the total tasks



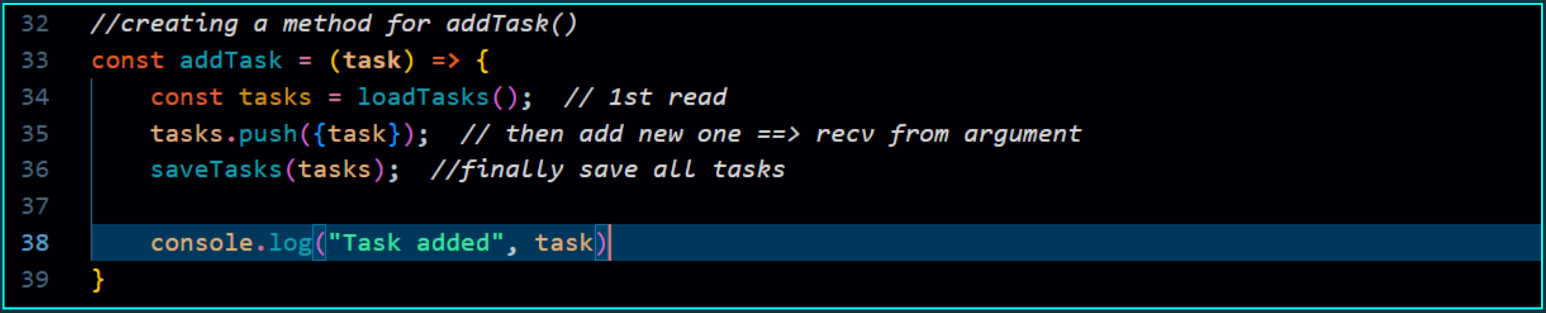
```javascript
//creating a method for addTask()
const addTask = (task) => {
    const tasks = loadTasks();  // 1st read
    tasks.push({task});  // then add new one ==> recv from argument
    saveTasks(tasks);  //finally save all tasks

}
```

8. Creating saveTasks() method ⇒ to save the all (old+new) tasks array [It is write in file]
Array [ ]⇒ convert to String ⇒ write to the file



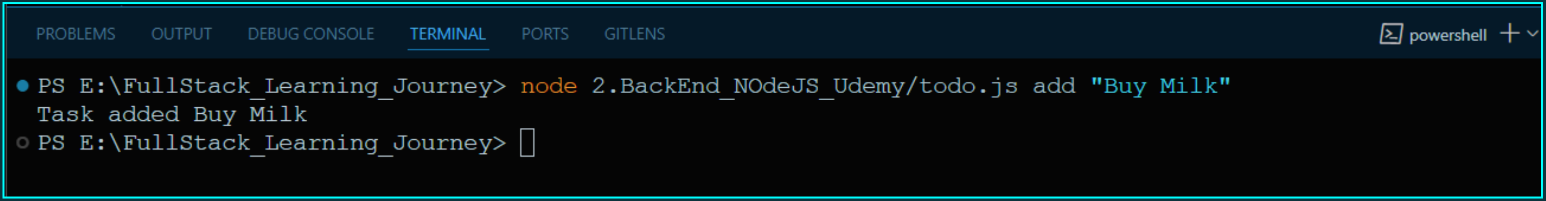
```javascript
//Creating saveTask()===> to save all data [convert arrary[] of all data in to string the write]
const saveTasks = (tasks) => {
    const dataString = JSON.stringify(tasks) // convert the JSON data into string (acting JSON)
    fs.writeFileSync(filePath, dataString);

}
```

9. See at console.log() ⇒ check whether new task is added or not ⇒ in the addTask()



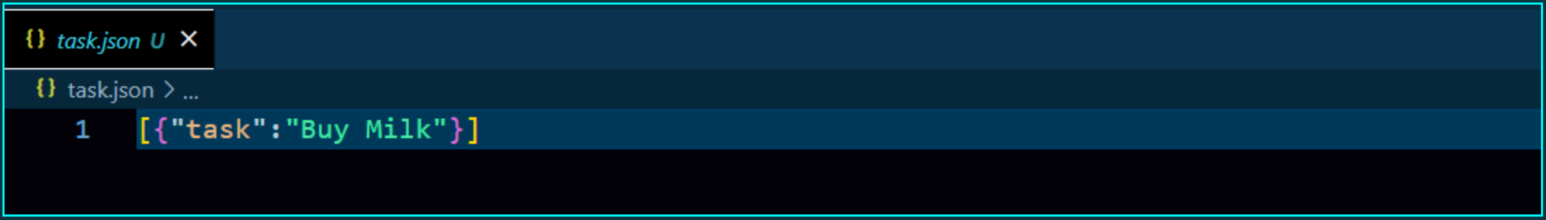
```javascript
//creating a method for addTask()
const addTask = (task) => {
    const tasks = loadTasks();  // 1st read
    tasks.push({task});  // then add new one ==> recv from argument
    saveTasks(tasks);  //finally save all tasks

    console.log("Task added", task)
}
```

10. Lets add "Buy Milk" ⇒ using powershell command ⇒ and see the result



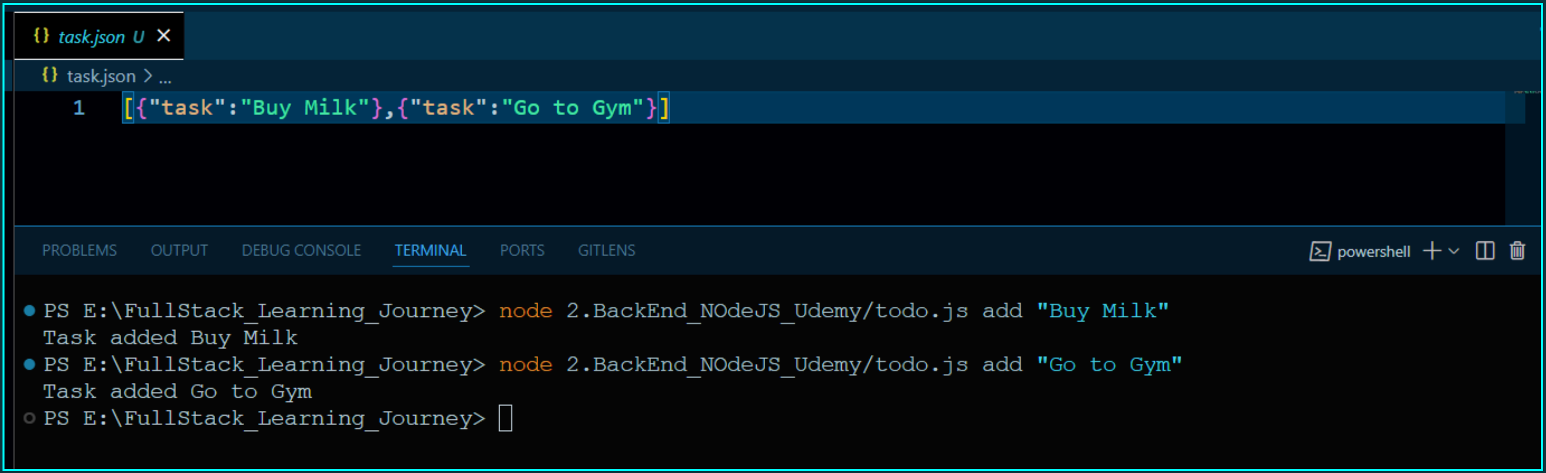
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS powershell

```
PS E:\FullStack_Learning_Journey> node 2.BackEnd_NOdeJS_Udemy/todo.js add "Buy Milk"
Task added Buy Milk
PS E:\FullStack_Learning_Journey>
```



```json
[{"task":"Buy Milk"}]
```

11. Adding Another

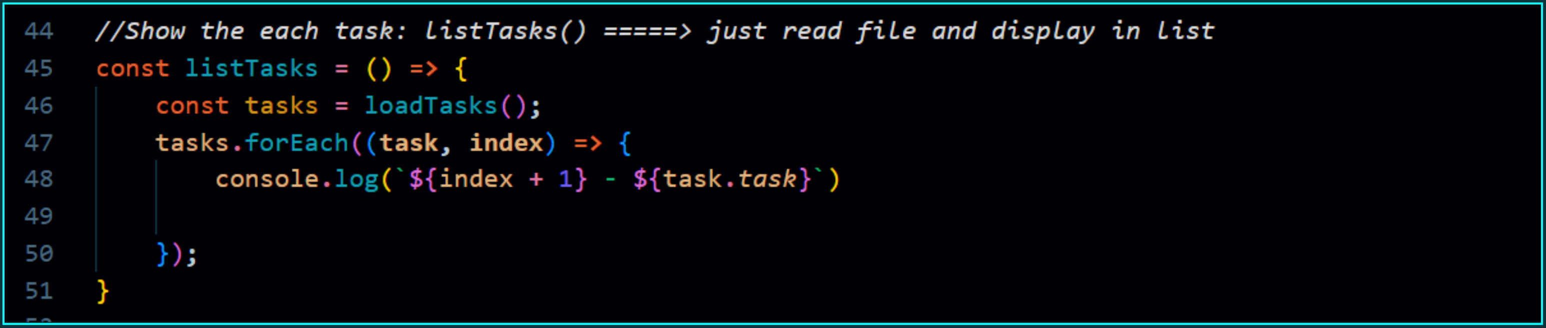


{} task.json U ×
{} task.json > ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS powershell

```json
[{"task":"Buy Milk"},{"task":"Go to Gym"}]
```

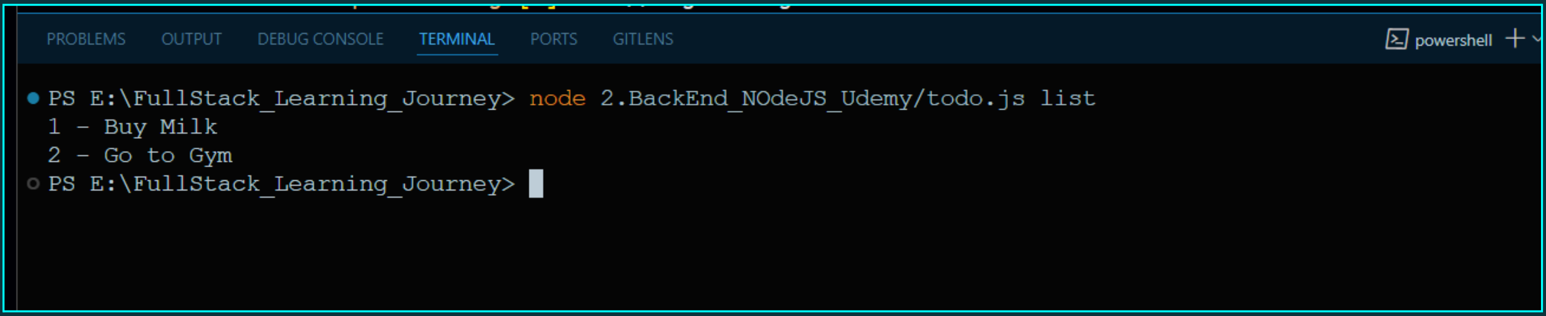
```
PS E:\FullStack_Learning_Journey> node 2.BackEnd_NOdeJS_Udemy/todo.js add "Buy Milk"
Task added Buy Milk
PS E:\FullStack_Learning_Journey> node 2.BackEnd_NOdeJS_Udemy/todo.js add "Go to Gym"
Task added Go to Gym
PS E:\FullStack_Learning_Journey>
```

12. List the tasks: listTasks() ⇒ just read the file & display each data in list



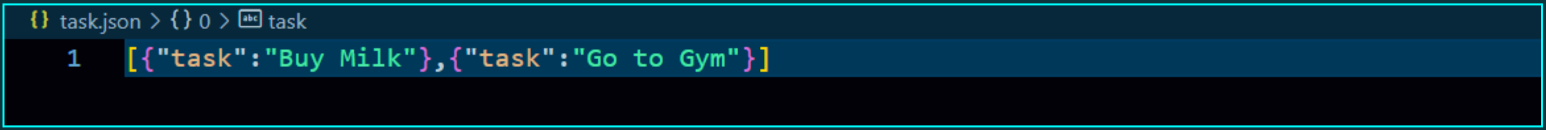
```javascript
//Show the each task: listTasks() =====> just read file and display in list
const listTasks = () => {
    const tasks = loadTasks();
    tasks.forEach((task, index) => {
        console.log(`${index + 1} - ${task.task}`)

    });
}
```

13. Now: run the command - 'list' ⇒ and see result =====> display each item in list



```
PS E:\FullStack_Learning_Journey> node 2.BackEnd_NOdeJS_Udemy/todo.js list
1 - Buy Milk
2 - Go to Gym
PS E:\FullStack_Learning_Journey>
```

From the below JSON, ⇒ all values are listed as above



```json
[{"task":"Buy Milk"},{"task":"Go to Gym"}]
```

Note:

1. **JavaScript file = todo.js**
2. **Backend file = task.json**