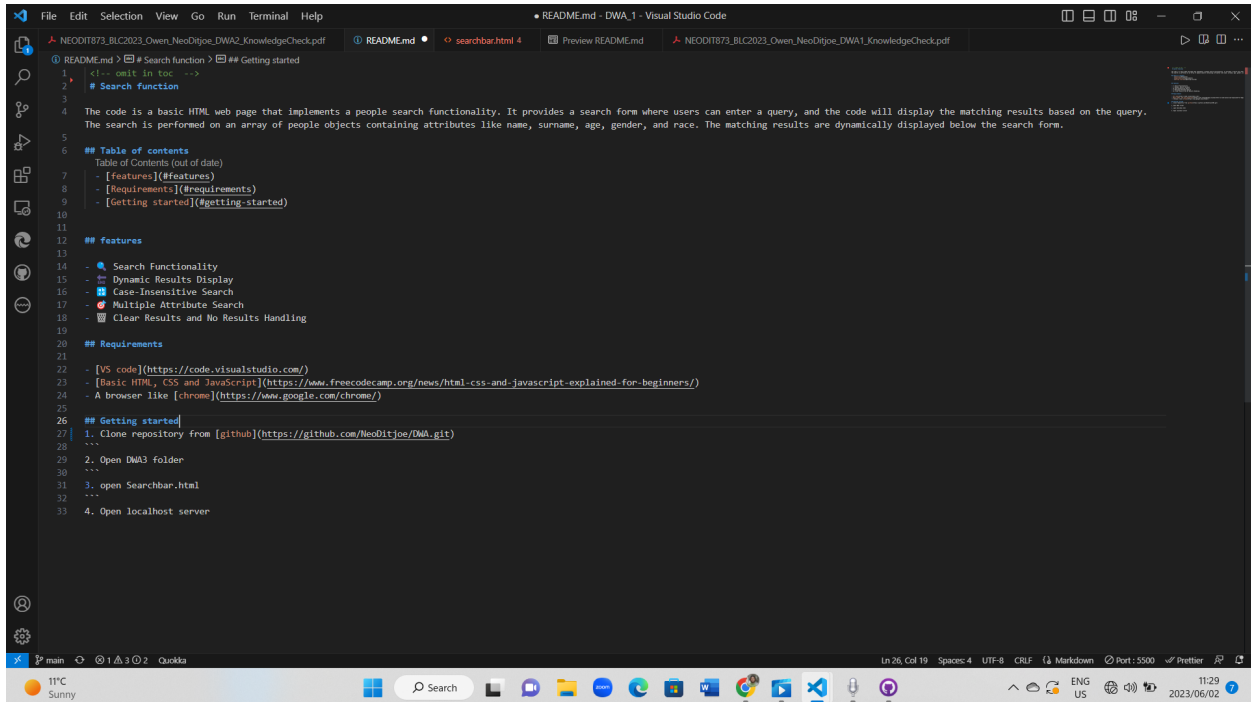


DWA_03.4 Knowledge Check_DWA3.1

1. Please show how you applied a Markdown File to a piece of your code.



The screenshot shows a Visual Studio Code editor window with a README.md file open. The file contains the following content:

```
1 <!-- omit in toc --> ## Getting started
2
3 # Search function
4
5 The code is a basic HTML web page that implements a people search functionality. It provides a search form where users can enter a query, and the code will display the matching results based on the query.
6 The search is performed on an array of people objects containing attributes like name, surname, age, gender, and race. The matching results are dynamically displayed below the search form.
7
8 ## Table of contents
9 Table of Contents (out of date)
10 - [features](#features)
11 - [Requirements](#requirements)
12 - [Getting started](#getting-started)
13
14 ## features
15 - 🔍 Search Functionality
16 - 📄 Dynamic Results Display
17 - 🔍 Case-Insensitive Search
18 - 🔄 Multiple Attribute Search
19 - 🗑️ Clear Results and No Results Handling
20
21 ## Requirements
22 - [VS code](https://code.visualstudio.com/)
23 - [Basic HTML, CSS and JavaScript](https://www.freecodecamp.org/news/html-css-and-javascript-explained-for-beginners/)
24 - A browser like [chrome](https://www.google.com/chrome/)
25
26 ## Getting started
27 1. Clone repository from [github](https://github.com/NeoDitjoe/DWA.git)
28 ...
29 2. Open DWA3 folder
30 ...
31 3. open Searchbar.html
32 ...
33 4. Open localhost server
```

The editor interface includes a sidebar with file explorer, search, and source control views. The bottom status bar shows the current file is 'README.md', the cursor is at line 26, column 19, and the file is encoded in UTF-8 with CR/LF line endings. The system tray at the bottom indicates a temperature of 11°C and a sunny weather condition.

2. Please show how you applied JSDoc Comments to a piece of your code.

```
/**
 * Searches for people based on the entered search value.
 *
 */
function searchPeople() {
    var searchValue = document.getElementById("searchInput").value.toLowerCase();
    var results = [];

    for (var i = 0; i < people.length; i++) {
        var person = people[i];
        var isMatched = Object.values(person).some(function(value) {
            return String(value).toLowerCase().includes(searchValue);
        });

        if (isMatched) {
            results.push(person);
        }
    }

    return results;
}
```

3. Please show how you applied the @ts-check annotation to a piece of your code.

```
//@ts-check
```

```
/**
```

```
 * @typedef {Object} Person
```

```
 * @property {string} name - The name of the person.
```

```
 * @property {string} surname - The surname of the person.
```

```
 * @property {number} age - The age of the person.
```

```
 * @property {string} gender - The gender of the person.
```

```
 * @property {string} race - The race of the person.
```

```
 */
```

```
/**
```

```
 * @type {Person[]}
```

```
 */
```

```
  var people = [
```

```
    {
```

```
      name: "Neo",
```

```
      surname: "Millions",
```

```
      age: 25,
```

```
      gender: "Male",
```

```
      race: "Caucasian"
```

```
    },
```

```
    {
```

```
      name: "john",
```

4. As a BONUS, please show how you applied any other concept covered in the 'Documentation' module.

