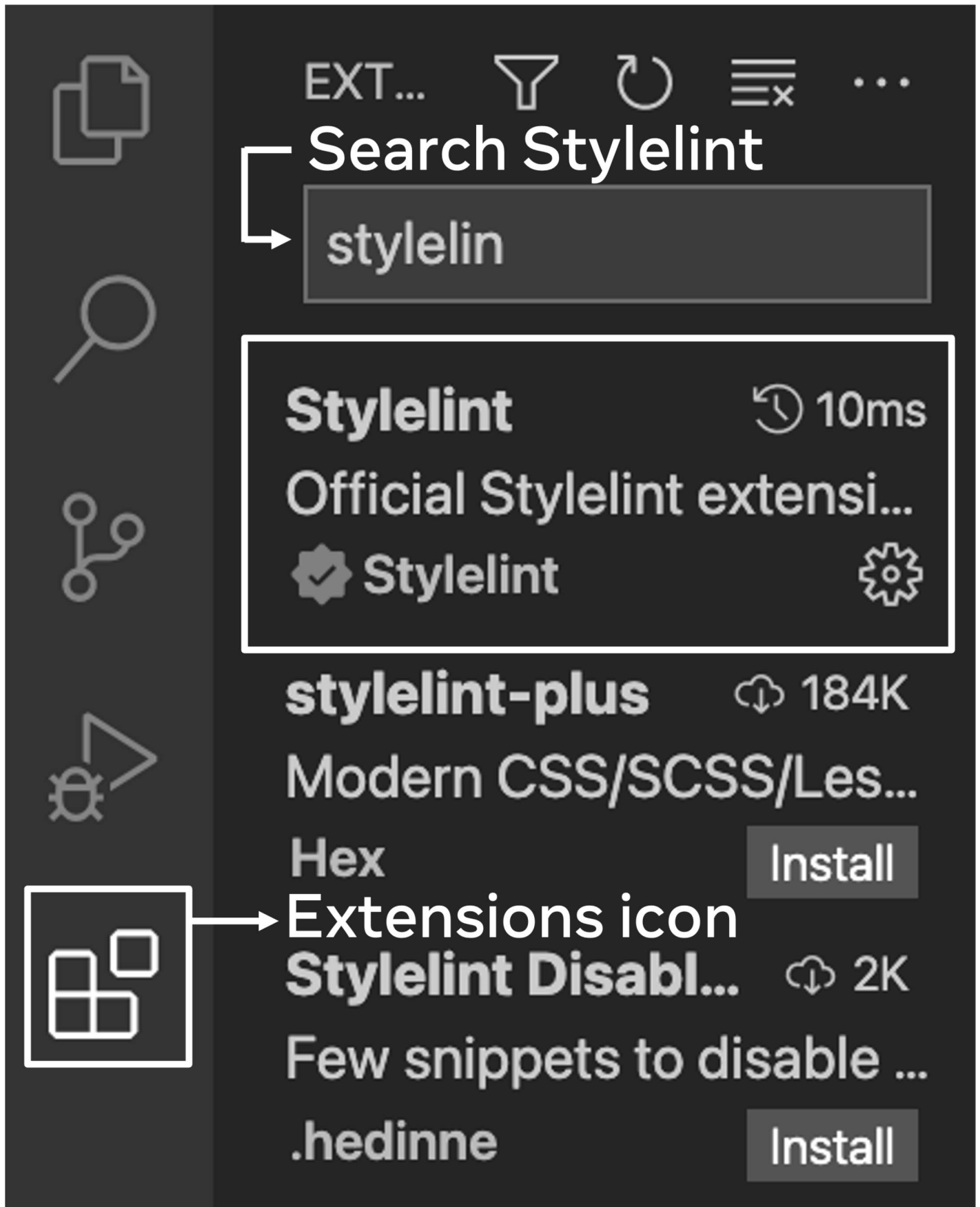Installing a third-party Linter in VS Code
The extension you are going to use that will help you avoid errors is called Stylelint and can be found in the Extensions library of VSCode. Follow the steps below to install it and enable it to use on your CSS code.
Step 1: Install the extension

Inside your VSCode, click on the Extensions icon and search for Stylelint.

Click on the Install button in the Extension box that appears.

Stylelint v1.2.2

Stylelint | 785,649 | ★★★★☆ (16)

Official Stylelint extension for Visual Studio Code

Install ⚙

Details Feature Contributions Changelog

Step 2: Ensure it is enabled
Once installed, the button will be replaced by the 'Disable' and 'Uninstall' button.

If you see 'Enable' instead of 'Disable', click on 'Enable' and make sure the extension is enabled.

Step 3: Create and add a file called '.stylelinttrc.json' to your project directory
The extension will look for this file inside any project folder you are working with.

Add the rules in JSON object format as below, these are the same as the ones you used in 'Handling errors'.

```
1    {
2    "rules": {
3
4            "alpha-value-notation": "number",
5            "selector-type-case": "lower",
6            "color-no-hex": true
7    }
8
9    }
```

Additionally, you can explore their official website for other rules you may want to add.
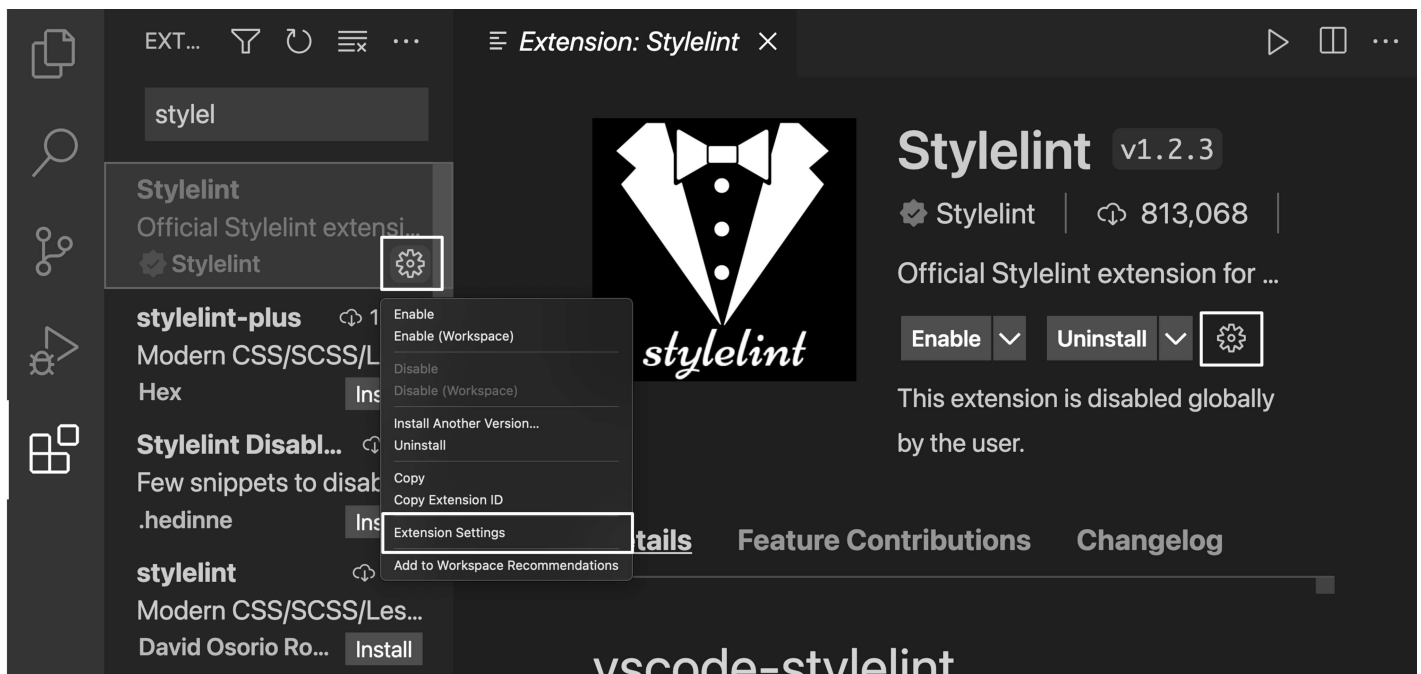
Step 4: Restart the VSCode
This will enable the extension completely.

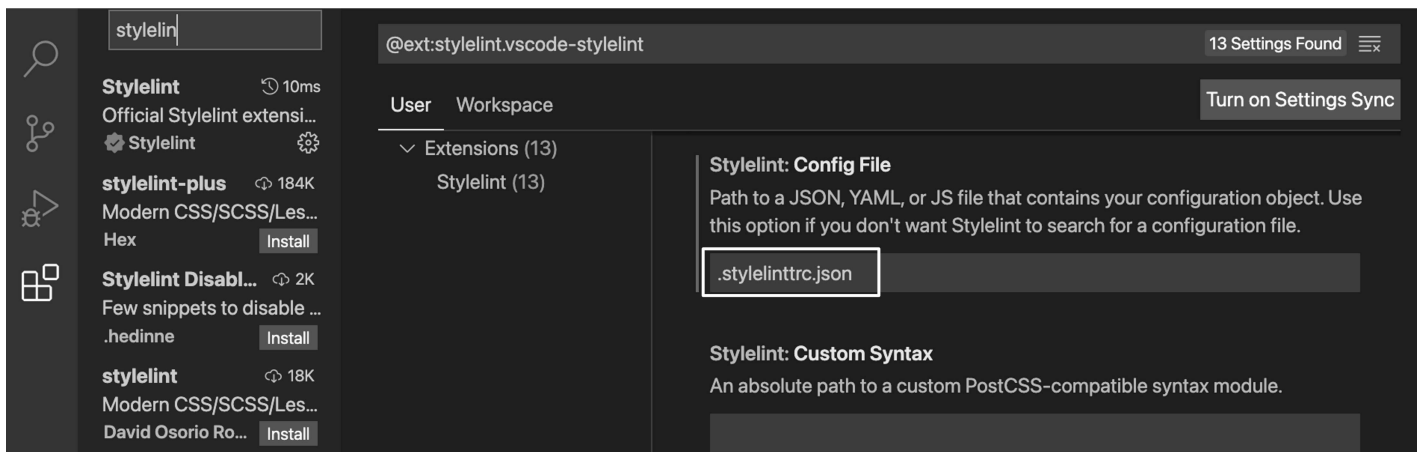Step 5: Add .stylelint.json file inside the Extension settings
Go to the Stylelint extension again by clicking on the Extension icon and then searching for Stylelint.

From directly next to the Stylelint title or next to the 'Uninstall' button, click on the 'Settings' icon and select 'Extension settings' from the drop-down list that appears.
Note that the Extension settings will not appear unless VSCode is properly reloaded.

After clicking on the Extension settings, scroll down until you see the section Stylelint: Config File and add the value ".stylelinttrc.json" under it. This is the same file name that you added inside your project folder.

Step 6: Look at the errors
Now you can open a CSS file and see the changes from Linter at work.

Go to View > Problems in your Menu bar. This will open the 'Problems' tab at the bottom.

The errors found by the Linter will display like in the screenshot above. It shows information about the rule which flagged the error. Additionally, numbers in the bracket here are respectively the line and character number for the place the error was found.

Step 7: Light bulb

PROBLEMS 2 ...

Filter (e.g. text, **/*.ts, !**/node_modules/**)

gridtemparea.css 2

Unexpected hex color "#fff" (color-no-hex) Stylelint(color-no-hex) [24, 12]

Disable color-no-hex for this line
Disable color-no-hex for the entire file
Show documentation for color-no-hex

fff" (color-no-hex) Stylelint(color-no-hex) [46, 12]

If you hover over any of the errors, a light-bulb icon will appear. If you click on it, a drop-down menu will appear. By selecting one of those options, you can explore the error that you have encountered in further detail.
And now you have installed and enabled Stylelint to use on your CSS code!

Note how specific rules can be set for specific projects and the Linter extension can be enabled or disabled inside your VSCode anytime. Specific lines can also be disabled for error-checks as in the screenshot above where you can note the options for disabling the rule for the specific line or disabling a specific rule for the entire file. There are different rules that can be found as per the requirements of your project.
There are a number of good CSS Linters that are extensions or they can be accessed from their respective websites. Some of these are mentioned in the additional readings of this section.