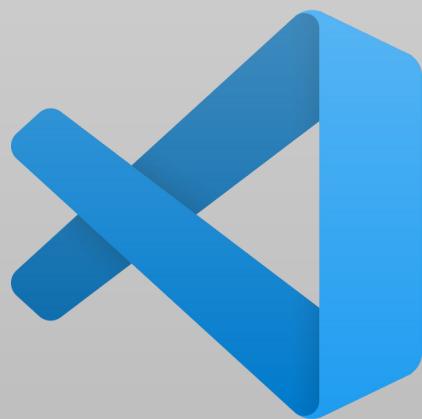




Dominando
o

Visual Studio Code

PET.COMP





Conhecendo o Visual Studio Code

PET.COMP

Introdução

- Desenvolvido pela Microsoft;
- Escrito em me JavaScript TypeScript,CSS;
- Baseado no Electron (framework para desenvolver aplicativos Node.js para o desktop) ;
- Software livre de código aberto;
- Lançamento em 29 de abril de 2015 (8 anos)



Visual Studio Code



Introdução

Vantagens:

- Altamente customizável;
- Interface intuitiva;
- Suporte a uma ampla variedade de linguagens de programação(C, C++, Python, Java...);
- Recursos Poderosos e Produtividade Aumentada;
 - IntelliSense;
 - Depuração Integrada;
 - Terminal Integrado
 - Controle de Versão Integrado

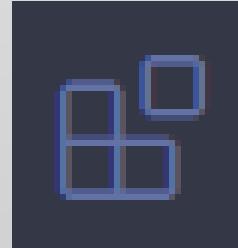


Visual Studio Code



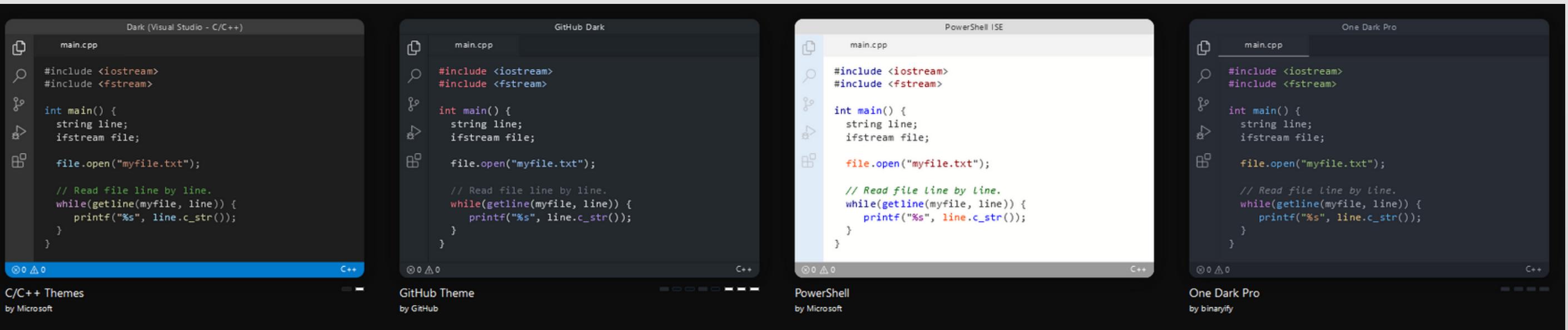
PET.COMP

Extensões para o VS Code



PET.COMP

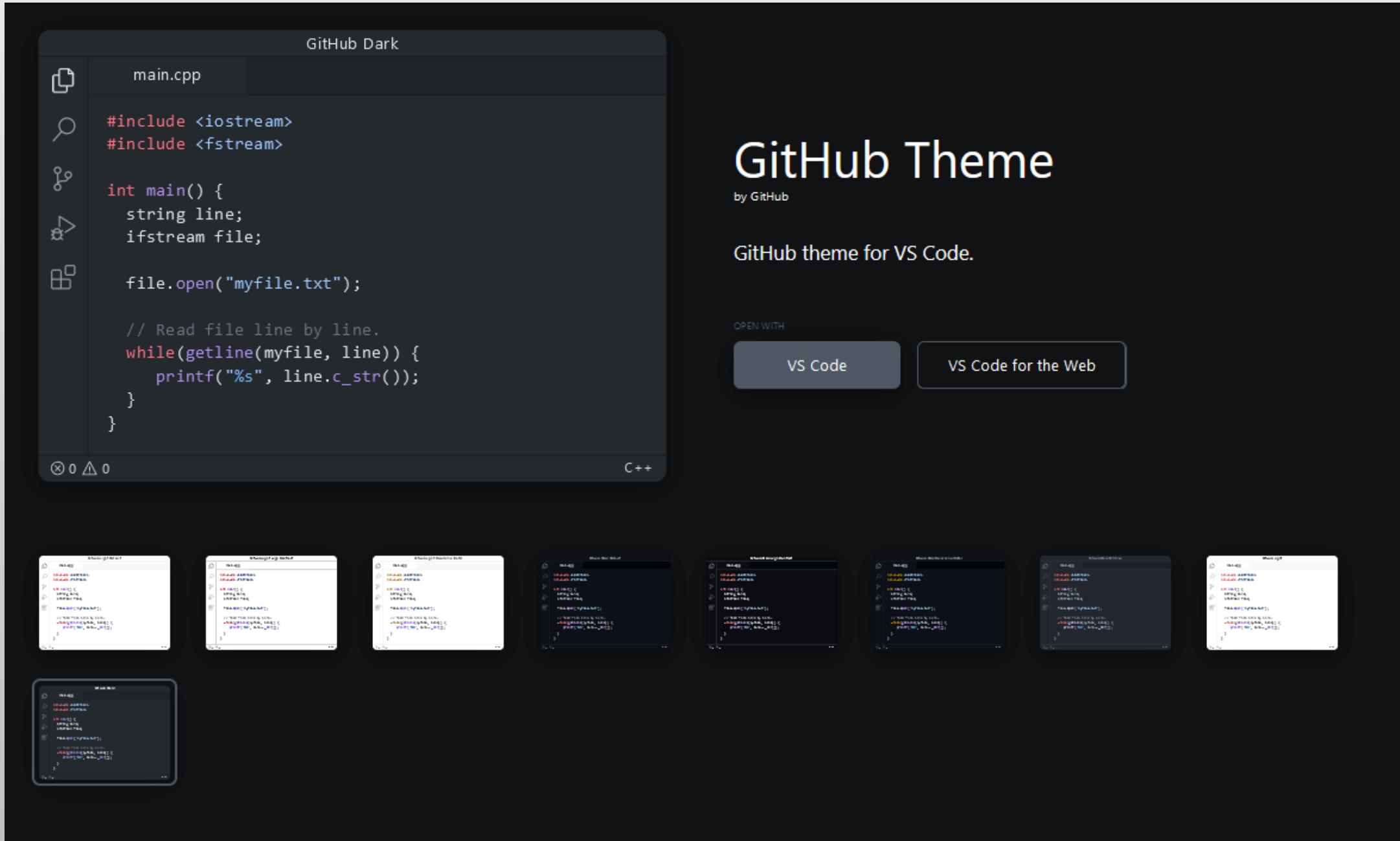
Temas para o VS Code



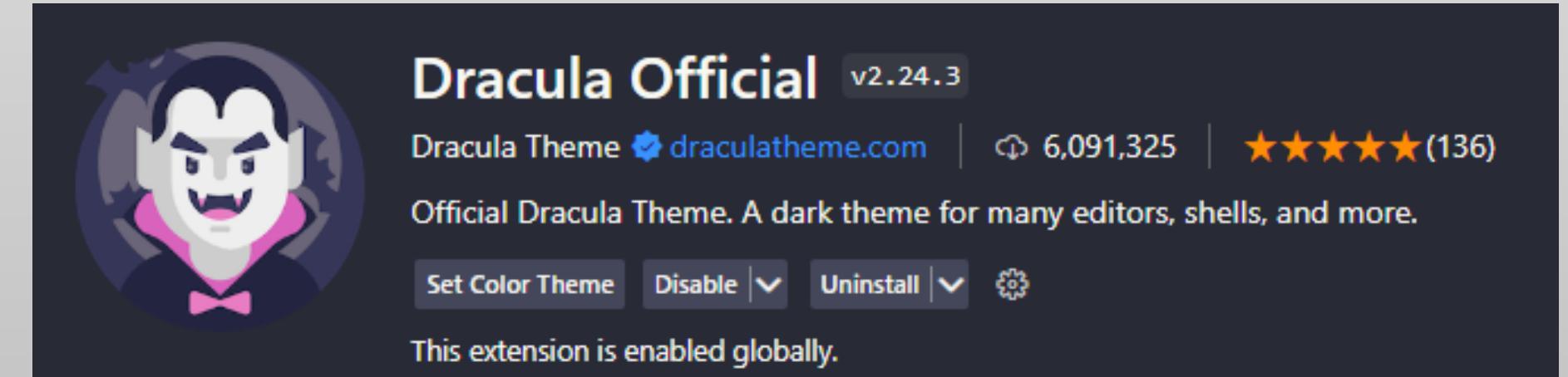
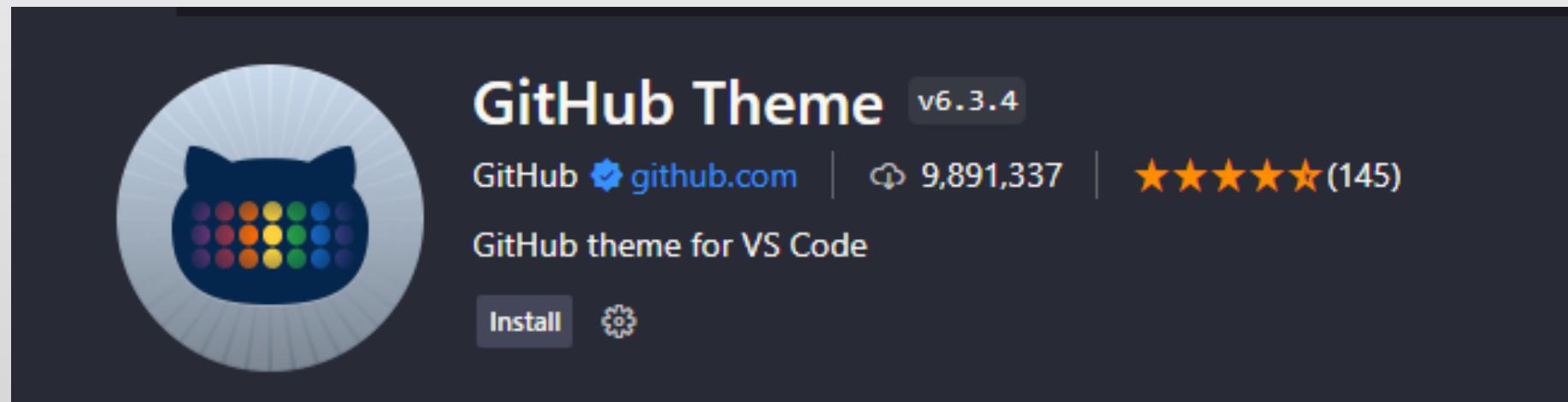
Site: vscodethemes.com



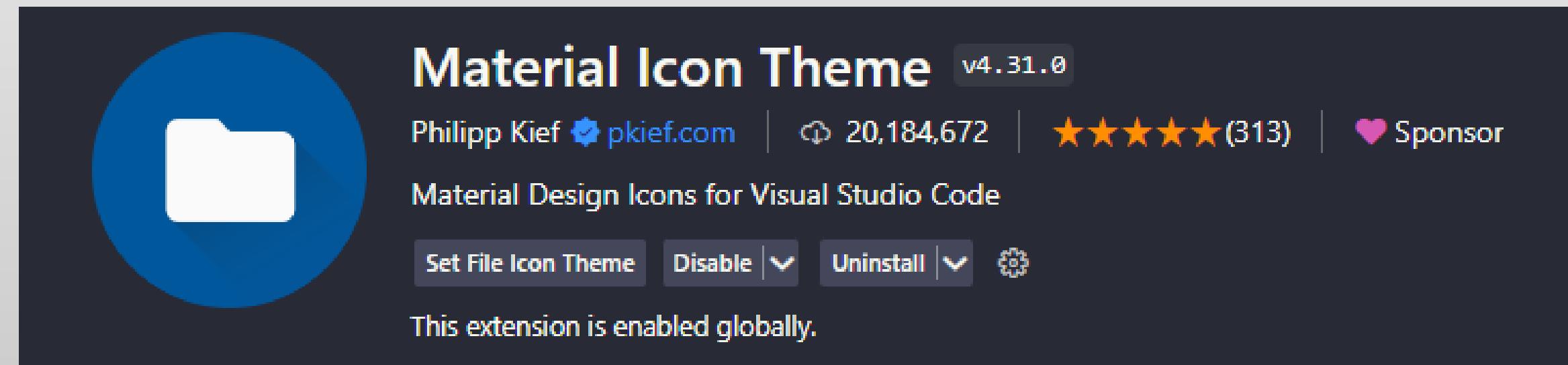
Temas para o VS Code



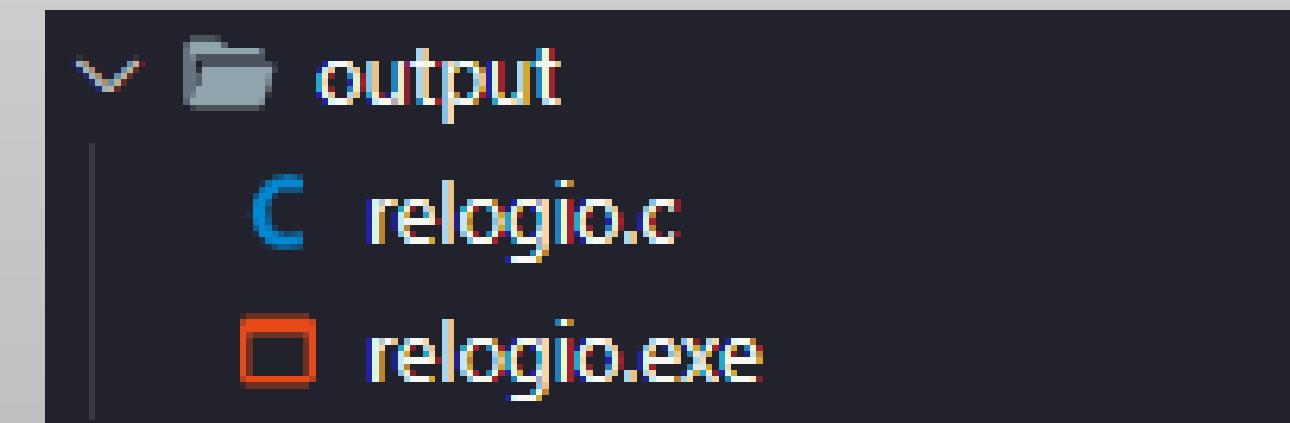
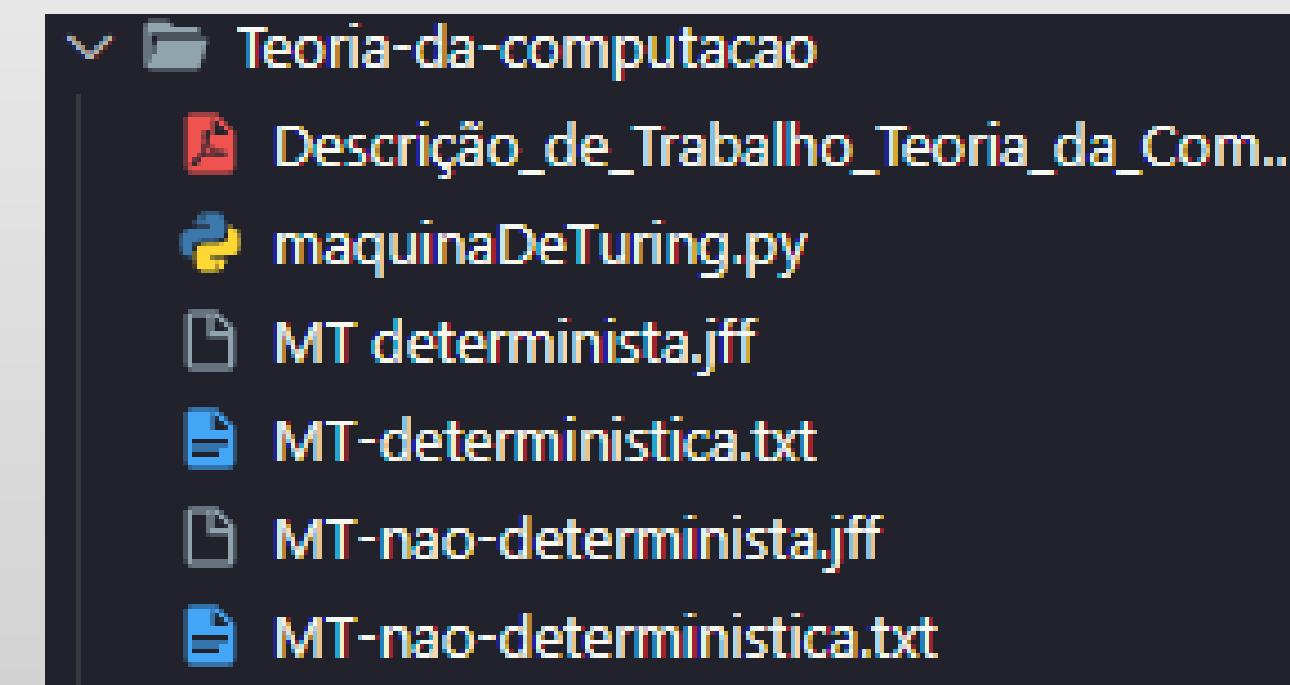
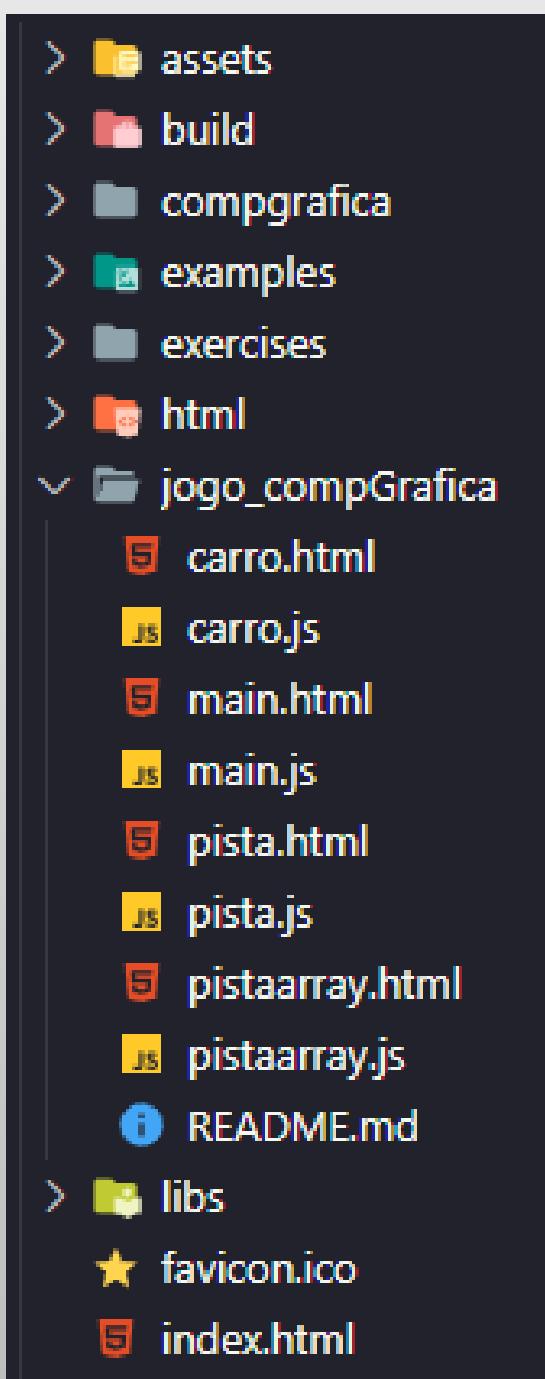
Temas para o VS Code



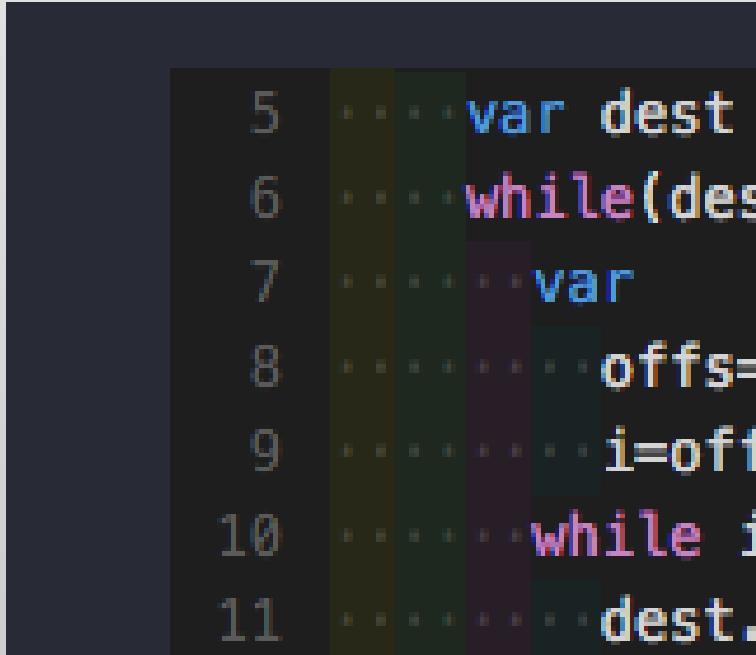
ícones de arquivos



Ícones de arquivos



Ajuda na indentação



5 ... var dest
6 ... while(des
7 var
8 offs=
9 i=off
10 while i
11 dest.

indent-rainbow v8.3.1

odenwat | ⚡ 6,169,121 | ★★★★★(119)

Makes indentation easier to read

[Disable](#) | [Uninstall](#)

This extension is enabled globally.



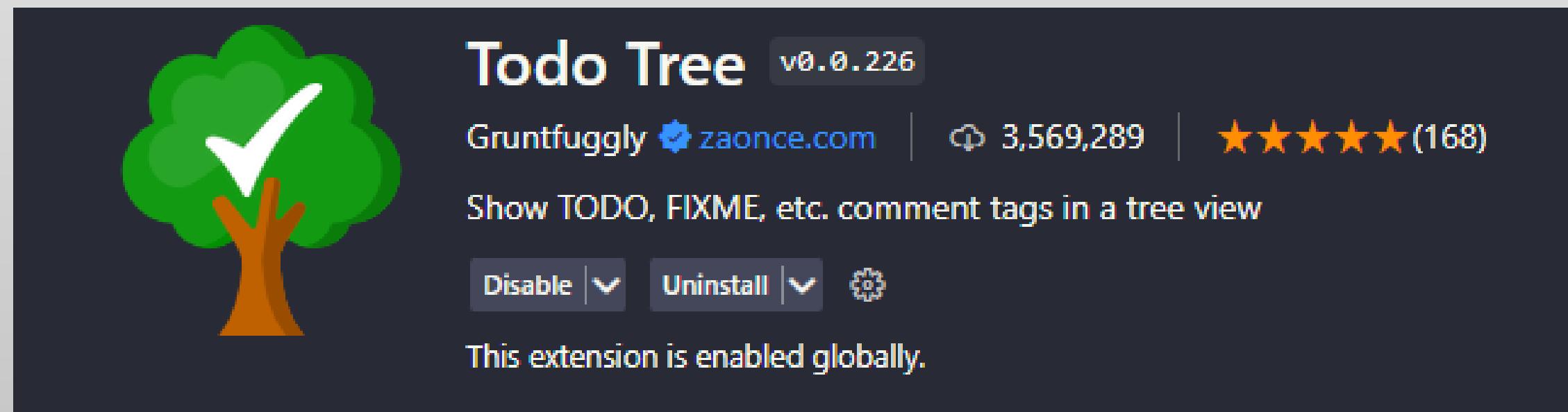
Ajuda na indentação

```
void clock(Page* memory, int num_frames, int page_to_insert) {
    int current_page = 0; // Índice da página atualmente sendo avaliada
    int page_replaced = 0; // Flag para indicar se uma página foi substituída

    while (1) {
        // Verifica se a página atual tem o bit de referência igual a 0
        if (memory[current_page].reference_bit == 0) {
            // Substitui a página atual pela nova página
            memory[current_page].page_number = page_to_insert;
            memory[current_page].reference_bit = 1;
            page_replaced = 1;
            break;
        }
        else {
            // Atualiza o bit de referência para 0 e passa para a próxima página
            memory[current_page].reference_bit = 0;
            current_page = (current_page + 1) % num_frames; // Incrementa o índice circularmente
        }
    }
}
```

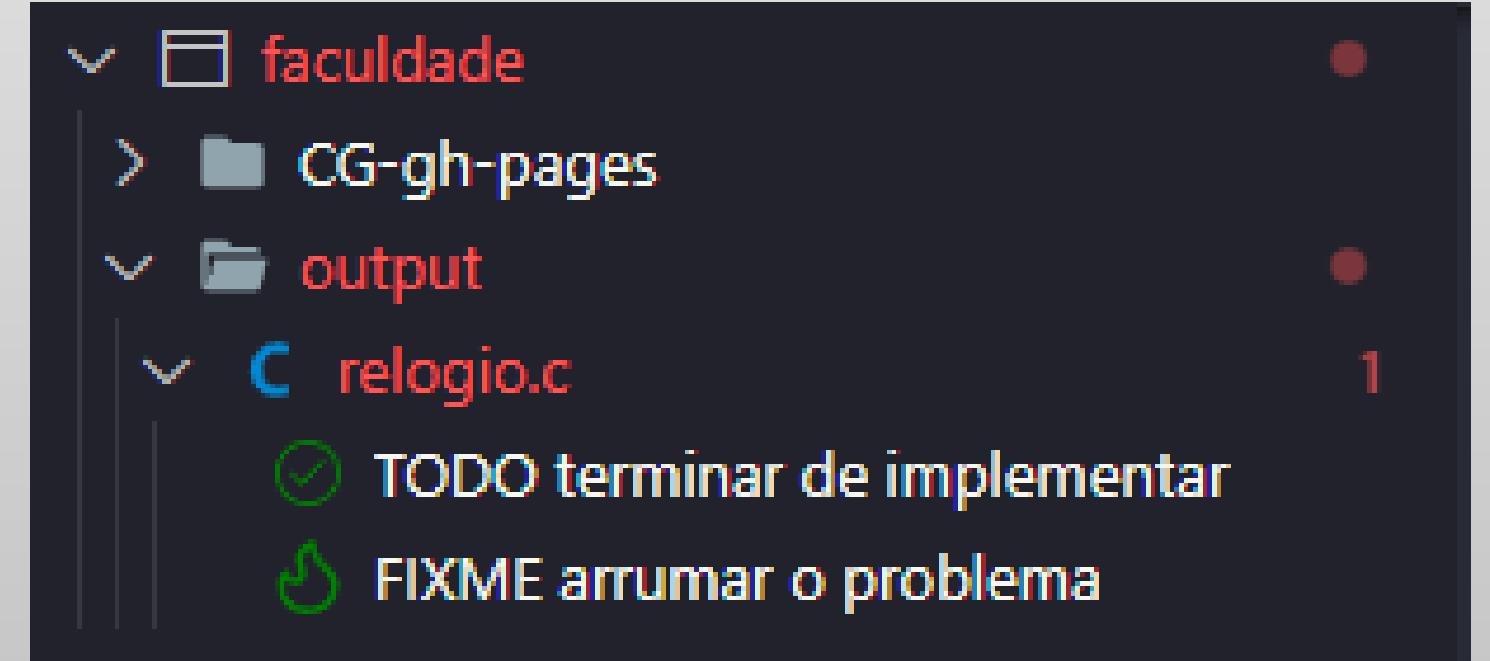


Marcação de afazeres em um código

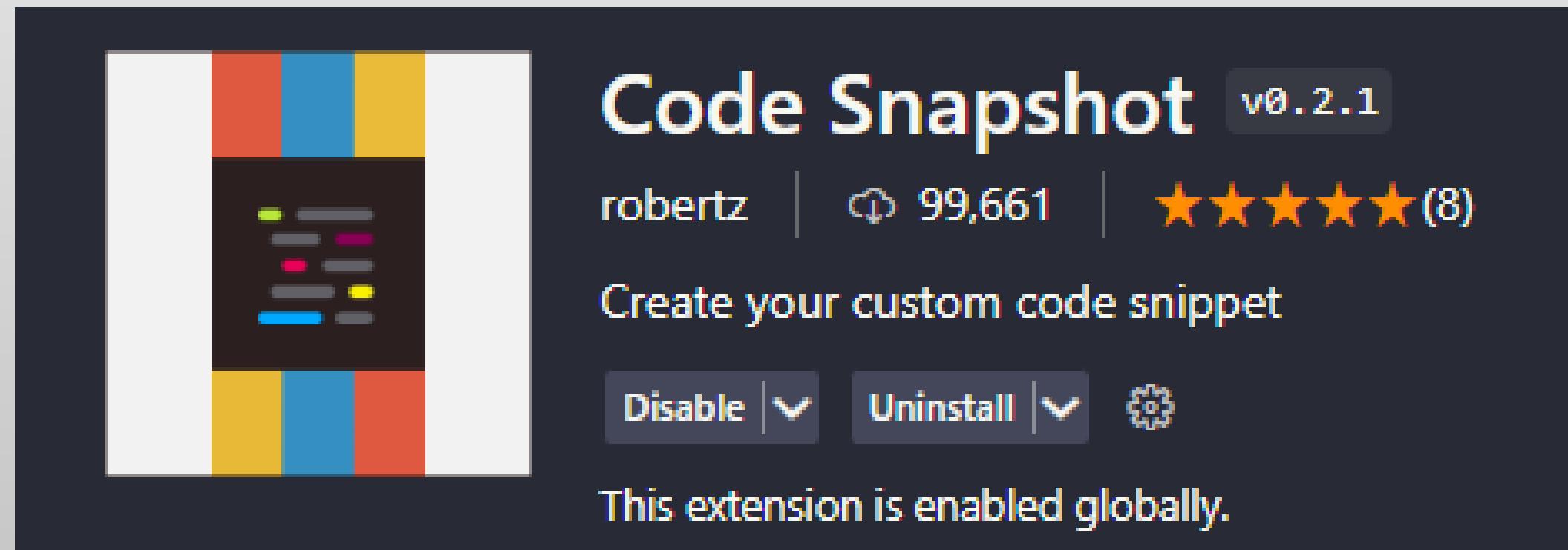


Marcação de afazeres em um código

```
//TODO terminar de implementar
int main() {
    Page memory[NUM_PAGES]; // Memória principal
    int num_frames = NUM_PAGES; // Número de quadros de página
    //FIXME arrumar o problema
    // Inicializa a memória principal com páginas vazias
    for (int i = 0; i < num_frames; i++) {
        memory[i].page_number = -1
        memory[i].reference_bit = 0;
    }
}
```



Capturar código



Capturar código

The image shows a code editor interface with two windows. On the left, a file named 'relogio.c' is open, displaying C code for a memory management algorithm. On the right, a 'Code Snapshot' window displays the same code in a simplified, compressed format.

```
C relogio.c x
faculdade > output > C relogio.c > clock(Page*, int, int)
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 #define NUM_PAGES 4 // Número de páginas na memória principal
5
6 typedef struct {
7     int page_number;
8     int reference_bit;
9 } Page;
10
11 void clock(Page* memory, int num_frames, int page_to_insert) {
12     int current_page = 0; // Índice da página atualmente sendo avaliada
13     int page_replaced = 0; // Flag para indicar se uma página foi substituída
14
15     while (1) {
16         // Verifica se a página atual tem o bit de referência igual a 0
17         if (memory[current_page].reference_bit == 0) {
18             // Substitui a página atual pela nova página
19             memory[current_page].page_number = page_to_insert;
20             memory[current_page].reference_bit = 1;
21             page_replaced = 1;
22             break;
23         }
24         else {
25             // Atualiza o bit de referência para 0 e passa para a próxima página
26             memory[current_page].reference_bit = 0;
27             current_page = (current_page + 1) % num_frames; // Incrementa o índice circularmente
28         }
29     }
30
31     // Se nenhuma página foi substituída ainda, todas as páginas têm o bit de referência igual a 0
32     // Portanto, substituimos a página atual e avançamos o ponteiro para a próxima página
33     if (!page_replaced) {
34         memory[current_page].page_number = page_to_insert;
35         memory[current_page].reference_bit = 1;
36         current_page = (current_page + 1) % num_frames;
37     }
38
39     // Exibe o conteúdo atual da memória principal após a inserção/substituição
40     printf("Memória principal: ");
41     for (int i = 0; i < num_frames; i++) {
42         printf("%d ", memory[i].page_number);
43     }
44 }
```

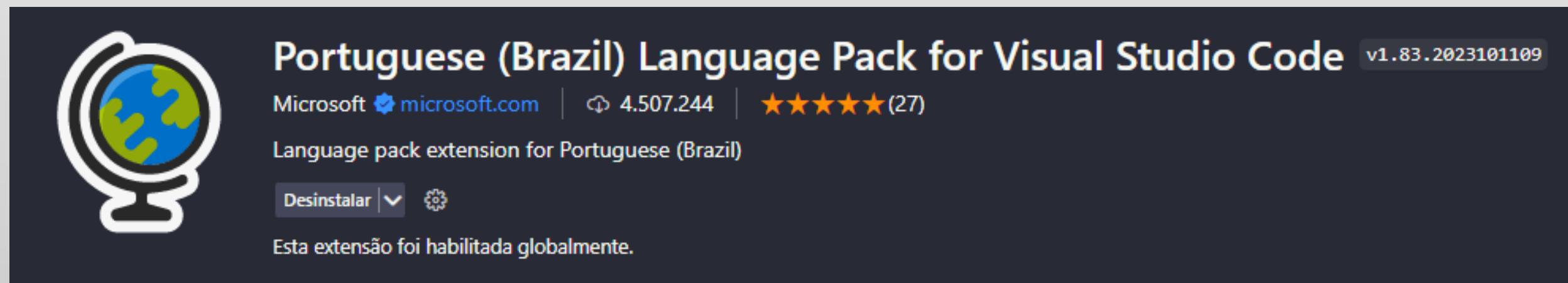
Code Snapshot X

Background Show line numbers 1356x1286

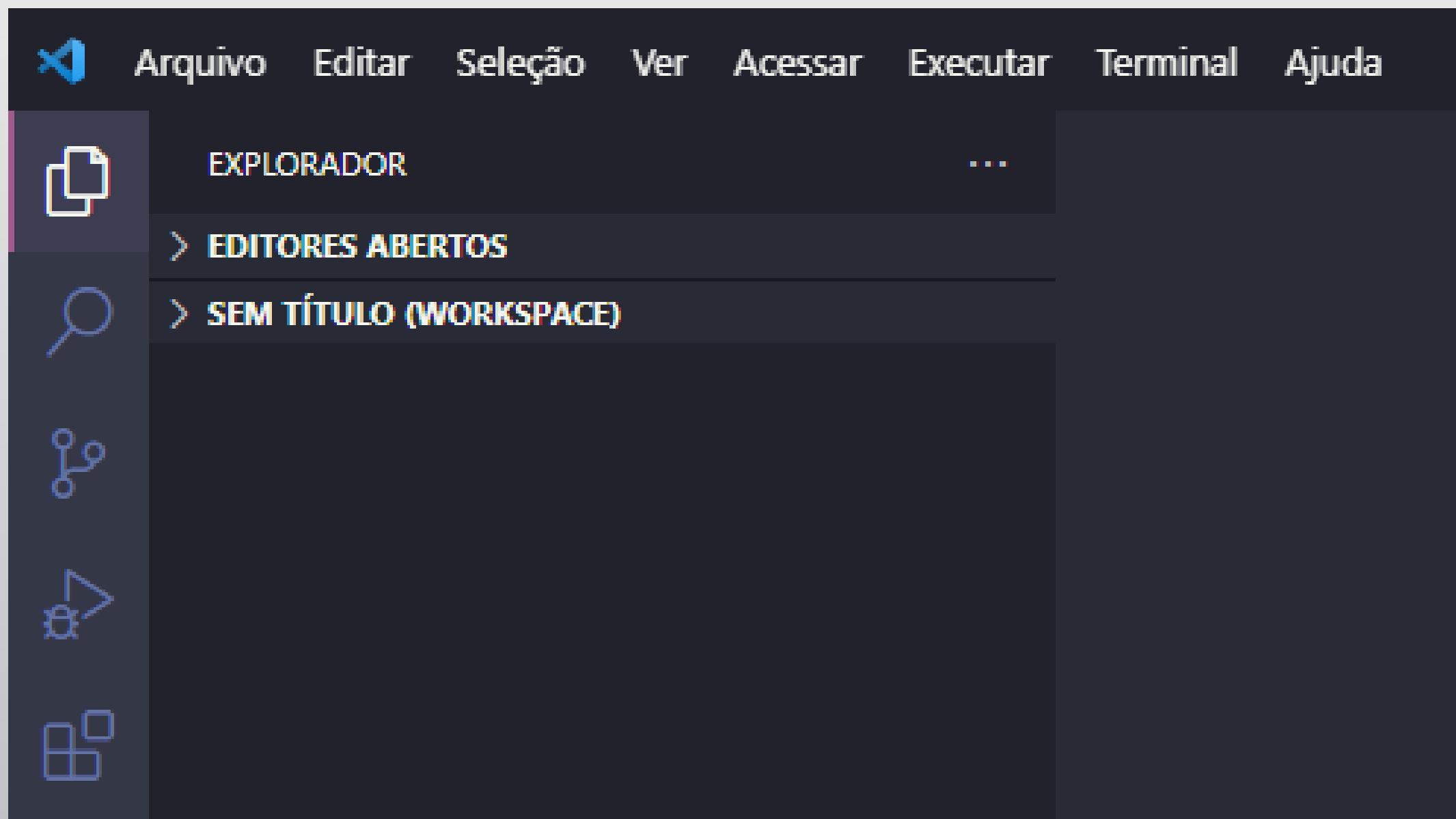
```
void clock(Page* memory, int num_frames, int page_to_insert) {
    int current_page = 0;
    // Índice da página atualmente sendo avaliada
    int page_replaced = 0;
    // Flag para indicar se uma página foi substituída
    while (1) {
        // Verifica se a página atual tem o bit de referência igual a 0
        if (memory[current_page].reference_bit == 0) {
            // Substitui a página atual pela nova página
            memory[current_page].page_number = page_to_insert;
            memory[current_page].reference_bit = 1;
            page_replaced = 1;
            break;
        }
        else {
            // Atualiza o bit de referência para 0 e passa para a próxima página
            memory[current_page].reference_bit = 0;
            current_page = (current_page + 1) % num_frames;
            // Incrementa o índice circularmente
        }
    }
}
```



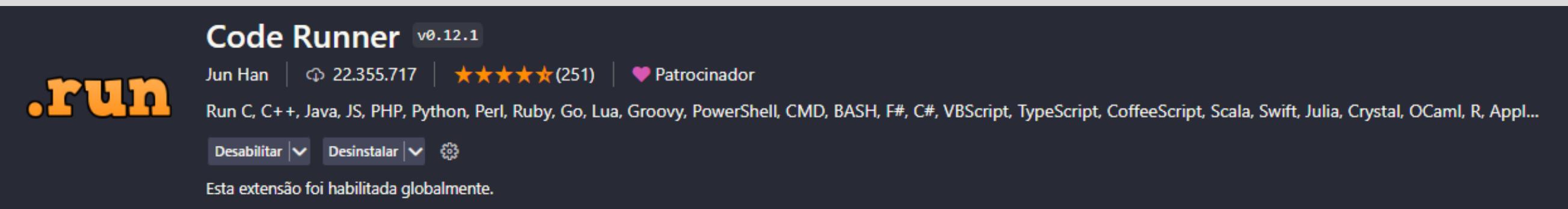
Tradução do VS Code



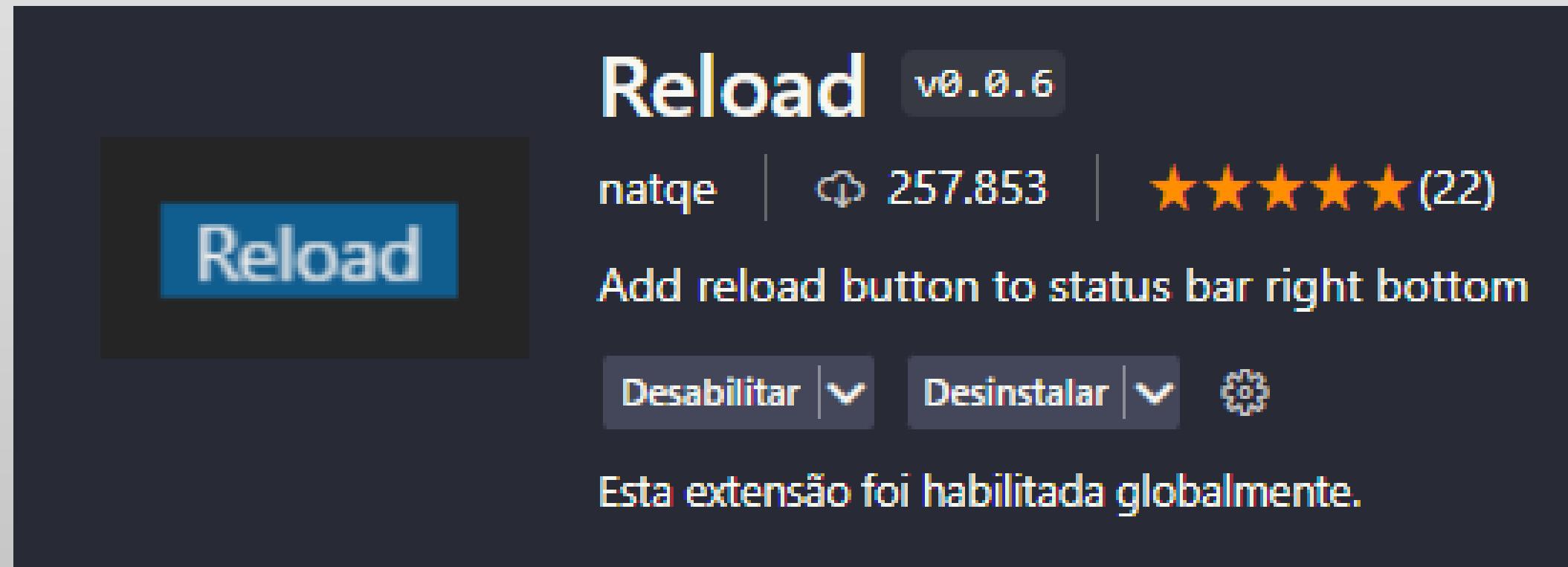
Tradução do VS Code



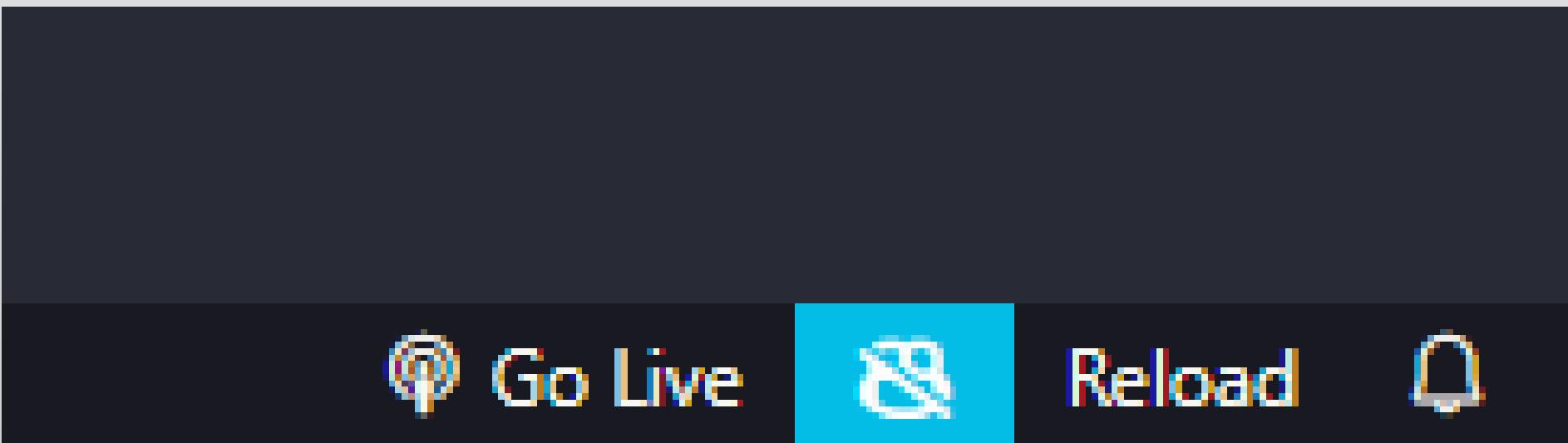
Compilar linguagens



Recarregar VS Code



Recargar VS Code

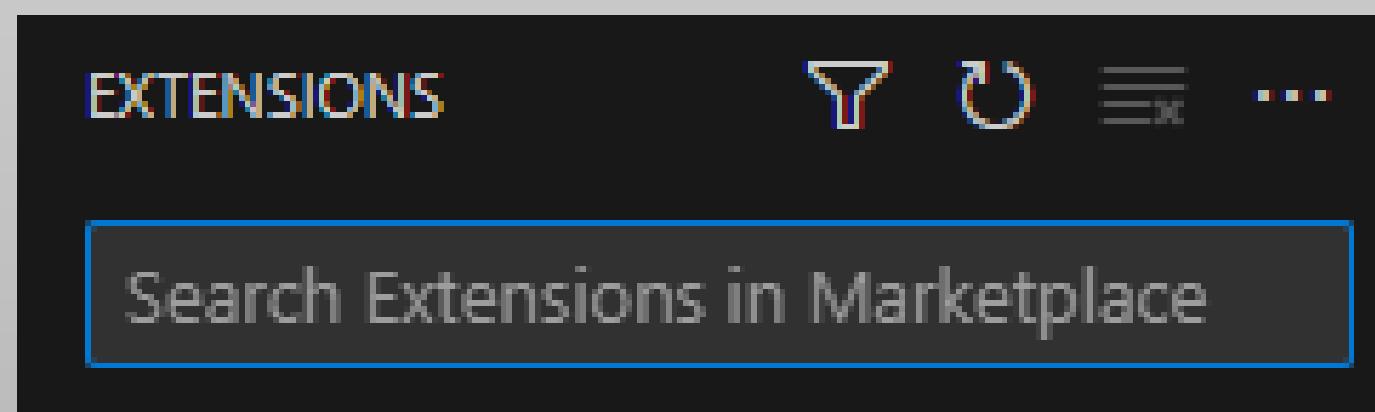
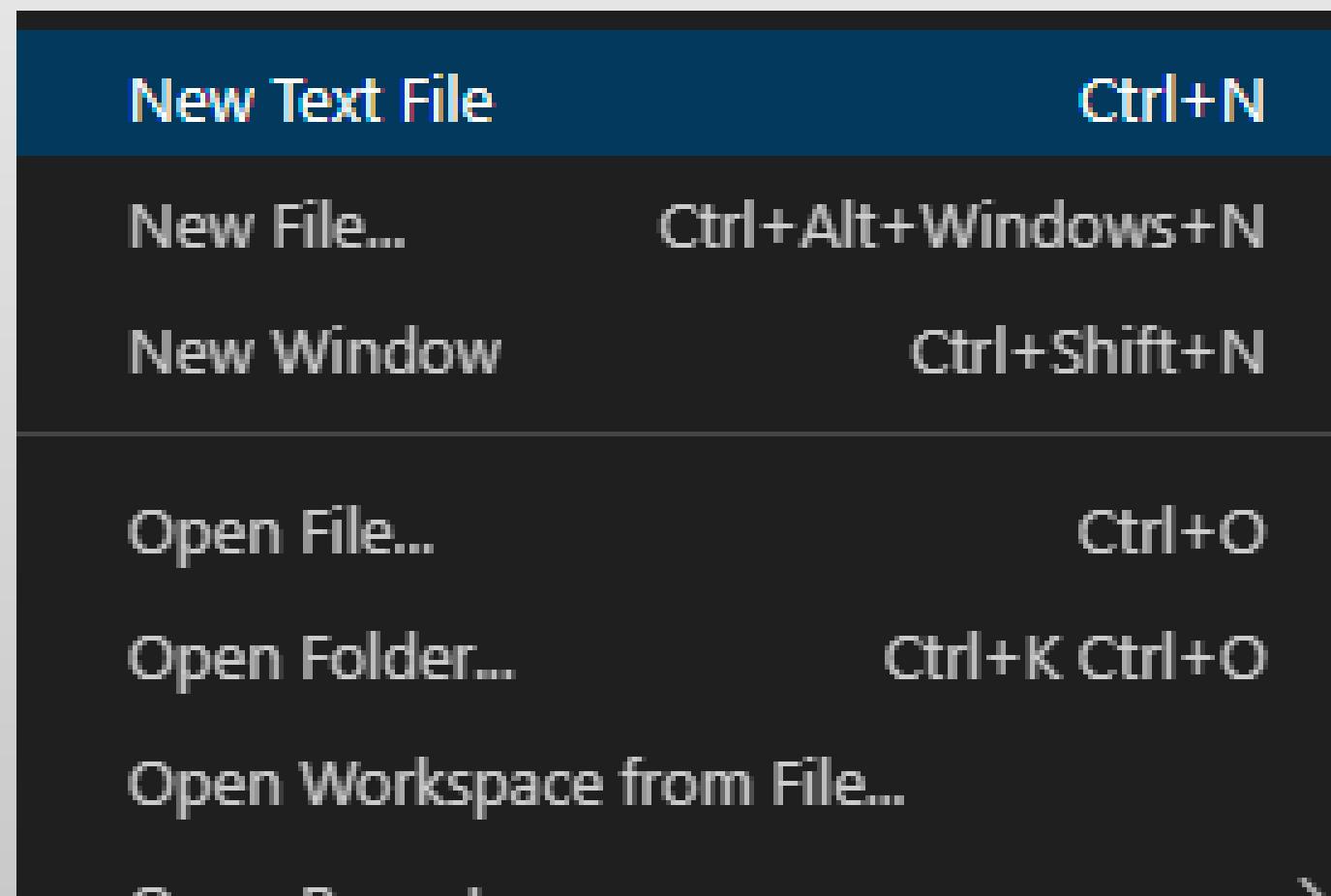




Atalhos para aumentar a produtividade

PET.COMP

Atalhos

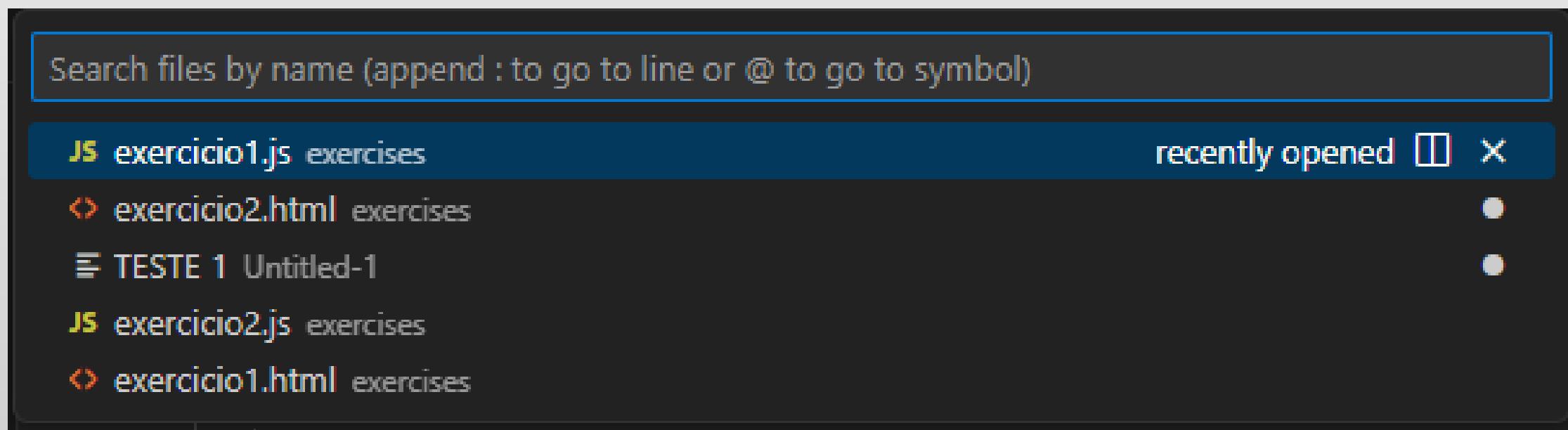


Lembrar da palavra Novo

EXTENSÕES = CTRL + SHIFT+ X

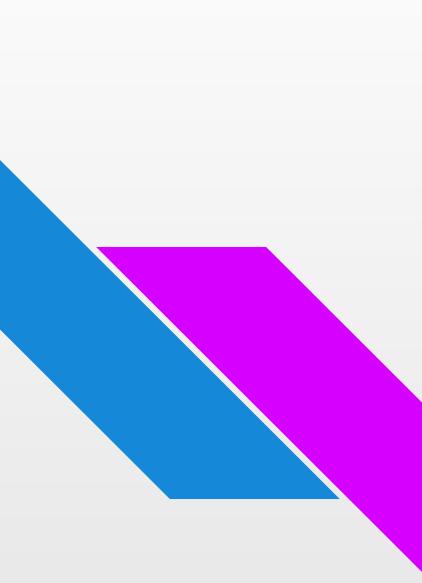


Atalhos

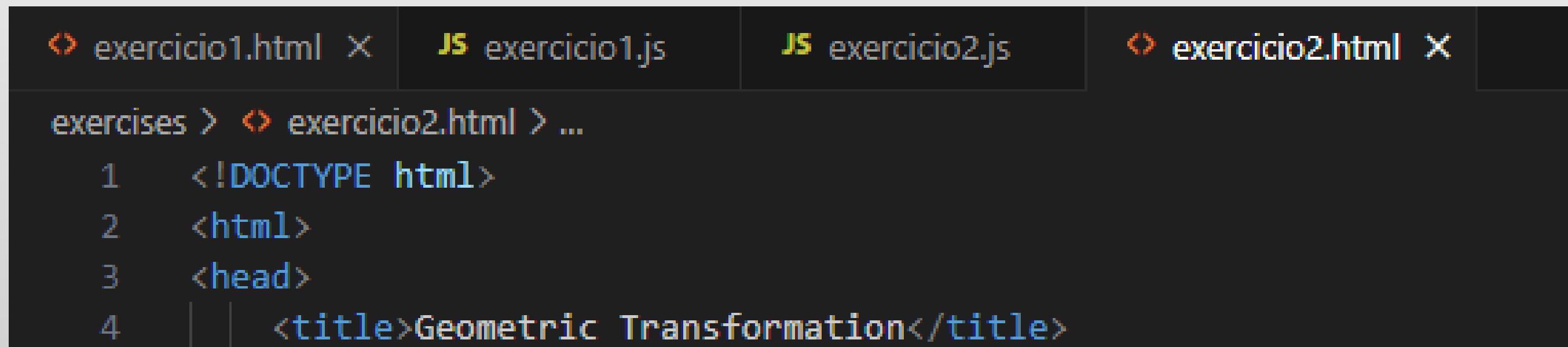


Abrir arquivos
CTRL + P





Atalhos



```
exercises > ◊ ejercicio2.html > ...
1   <!DOCTYPE html>
2   <html>
3   <head>
4     <title>Geometric Transformation</title>
```

ALT + NUMERO
ou
CTRL + TAB



Atalhos

```
#include <stdio.h>

int main() {
    int num1 = 10;
    int num2 = 20;
    int num3 = 30;
    int soma = num1 + num2 + num3;

    printf("Os números são: %d, %d e %d\n", num1, num2, num3);
    printf("Os números somados são: %d\n", soma);

    return 0;
}
```

Adicionar seleção a próxima ocorrência
CTRL + D



Atalhos

```
#include <stdio.h>

int main() {
    int num1 = 10;
    int num2 = 20;
    int num3 = 30;
    int soma = num1 + num2 + num3;

    printf("Os números são: %d, %d e %d\n", num1, num2, num3);
    printf("Os números somados são: %d\n", soma);

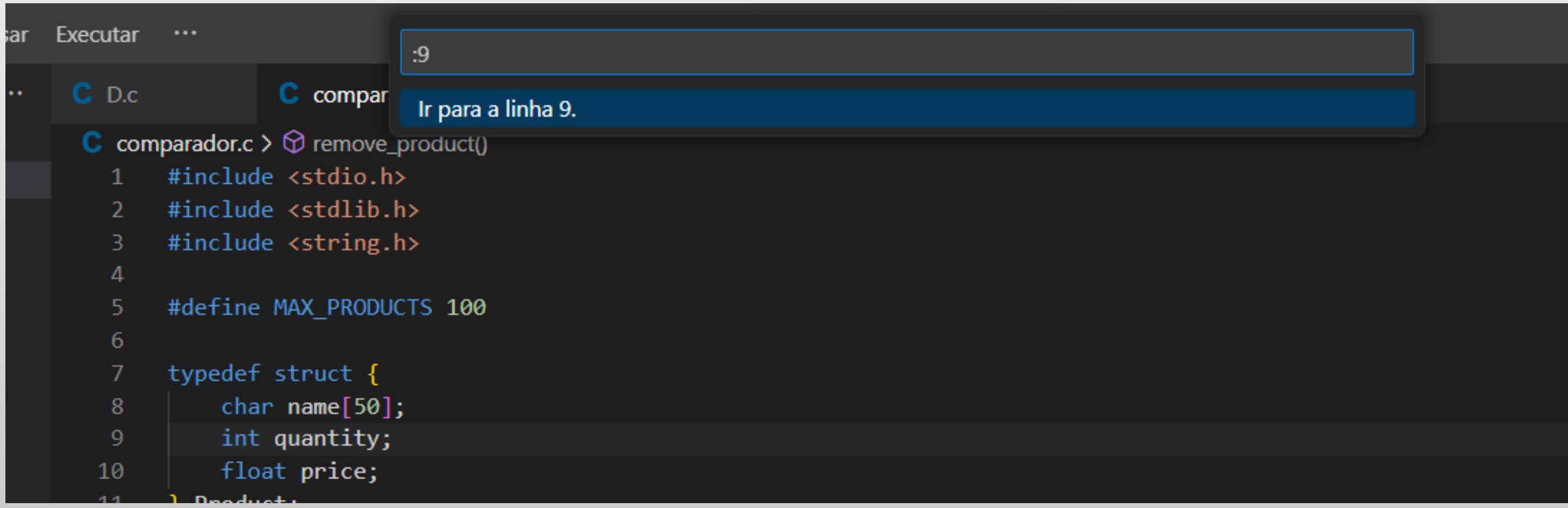
    return 0;
}
```

Adicionar seleção a todas ocorrência

CTRL + SHIFT + L



Atalhos



```
car Executar ... :9
.. C D.c      C compar Ir para a linha 9.
C comparador.c > remove_product()
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 #define MAX_PRODUCTS 100
6
7 typedef struct {
8     char name[50];
9     int quantity;
10    float price;
11 } Product;
```

Navegar para linha específica do código
CTRL + G



Atalhos

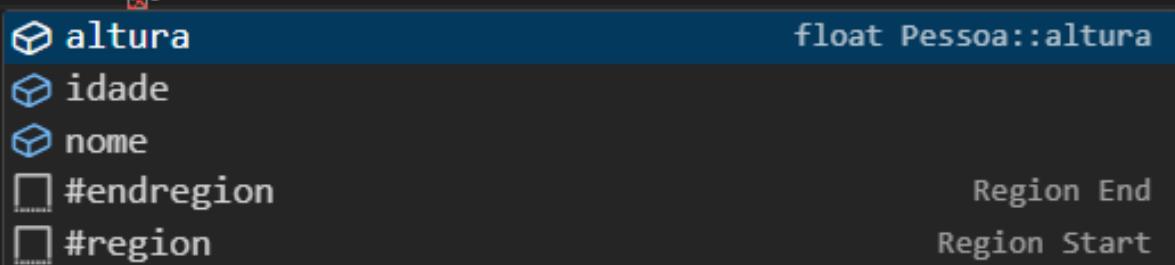
```
#include <stdio.h>

// Definição da classe Pessoa
struct Pessoa {
    char nome[50];
    int idade;
    float altura;
};

int main() {
    // Inicialização de uma instância da classe Pessoa
    struct Pessoa joao = {"João", 25, 1.75};

    // Impressão dos valores dos atributos da instância
    printf("Nome: %s\n", joao.nome);
    printf("Idade: %d\n", joao.idade);
    printf("Altura: %.2f\n", joao.altura);
}

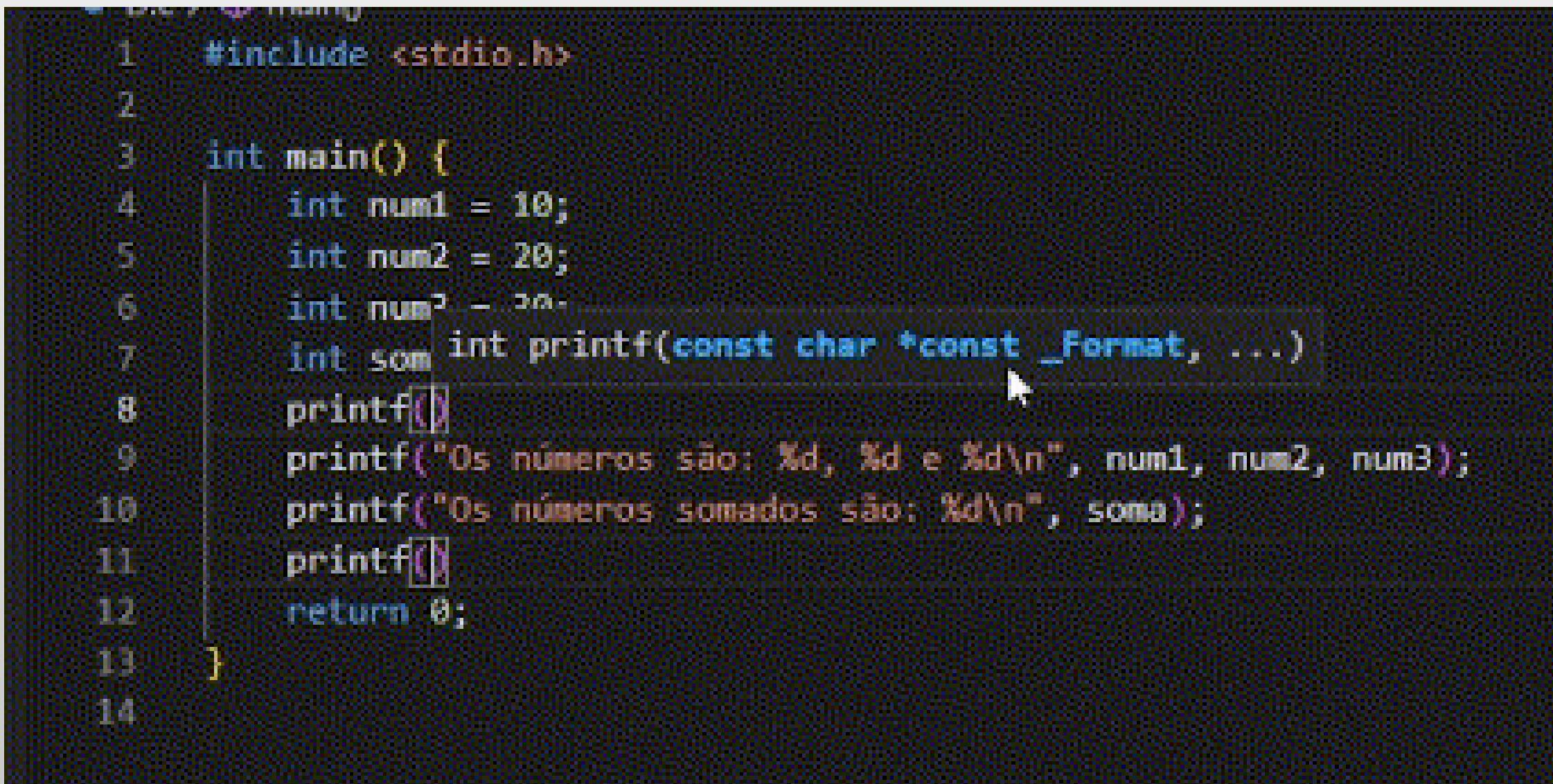
return 0;
}
```



Autopreenchimento
CTRL + ESPAÇO



Atalhos



A screenshot of a terminal window displaying a C program. The code is as follows:

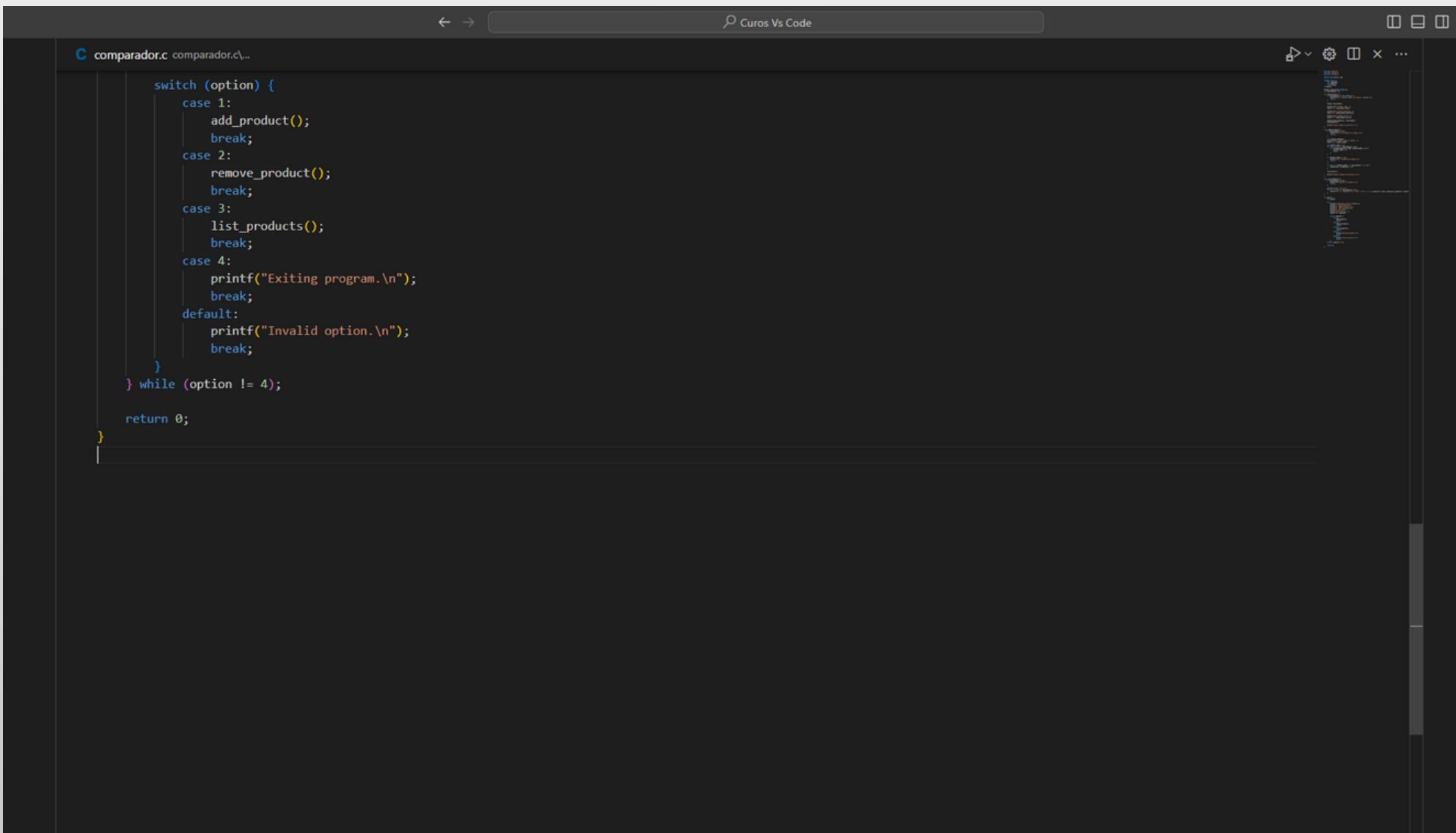
```
1 #include <stdio.h>
2
3 int main() {
4     int num1 = 10;
5     int num2 = 20;
6     int num3 = 30;
7     int soma;
8     printf();
9     printf("Os números são: %d, %d e %d\n", num1, num2, num3);
10    printf("Os números somados são: %d\n", soma);
11    printf();
12    return 0;
13 }
14
```

The cursor is positioned at the end of the word "printf" in line 7. There are also two other cursors: one at the end of the opening parenthesis in line 8, and another at the end of the closing parenthesis in line 11. This illustrates the use of the multi-cursor feature in the terminal.

Multi cursor
ALT + Click



Atalhos



```
C comparador.c comparador.c...
switch (option) {
    case 1:
        add_product();
        break;
    case 2:
        remove_product();
        break;
    case 3:
        list_products();
        break;
    case 4:
        printf("Exiting program.\n");
        break;
    default:
        printf("Invalid option.\n");
        break;
}
} while (option != 4);

return 0;
}
```

modo Zen
CTRL + K Z



Atalhos

```
Área de transferência ↔ comparador.c comparador.c\add_product()
```

```
int main() {
    int option;

    do {
        printf("\nInventory Control System\n");
        printf("1. Add product\n");

        printf("4. Exit\n");
        printf("Enter option: ");
        scanf("%d", &option);

        switch (option) {
            case 1:
                add_product();
                break;
            case 2:
                remove_product();
                break;
            case 3:
                list_products();
                break;
            default:
                printf("Invalid option.\n");
                break;
        }
    } while (option != 4);

    return 0;
}
```

```
int main() {
    int option;

    do {
        printf("\nInventory Control System\n");
        printf("1. Add product\n");
        printf("2. Remove product\n");
        printf("3. List products\n");
        printf("4. Exit\n");
        printf("Enter option: ");
        scanf(" %d", &option);

        switch (option) {
            case 1:
                add_product();
                break;
            case 2:
                remove_product();
                break;
            case 3:
                list_products();
                break;
            case 4:
                printf("Exiting program.\n");
                break;
            default:
                printf("Invalid option.\n");
                break;
        }
    } while (option != 4);

    return 0;
}
```

Comparador de código com área de transferência
CTRL + K C



Atalhos

```
for(i=0;i< 10;i++){  
    x = 10;  
    int x;  
    x = 50;
```

este for NAO DEVE FAZ ABSOLUTAMENTE NADA

```
printf("numero %d", k);
```

VOCE PRECISA PRINTAR A VARIAVEL i 10 VEZES NESSE CODIGO ALTERNANDO ENTRE i = 10 e i = 50

```
k=0:::::  
int k;  
};  
i = x;  
int i;
```

VAMOS APRENDER HABILIDADES MAGICAS PARA MANIPULAR NOSSOS CODIGOS NO VSCADE

SEU AMIGO NAO VAI ENTENDER NADA VENDO VOCE PROGRAMAR no VSCADE

BEM VINDOS AO MINICURSO DOMINANDO O VSCADE



Atalhos

```
// BEM VINDOS AO MINICURSO DOMINANDO O VS CODE
// VAMOS APRENDER HABILIDADES MAGICAS PARA MANIPULAR NOSSOS CODIGOS NO VS CODE
// SEU AMIGO NAO VAI ENTENDER NADA VENDO VOCE PROGRAMAR no VS CODE
// VOCE PRECISA PRINTAR A VARIABEL i 5 VEZES NESSE CODIGO ALTERNANDO ENTRE i = 10 e i = 50

int x;
int i;
int k;
x = 10;
i = x;
printf("numero %d", i);

x = 50;
i = x;
printf("numero %d", i);

x = 10;
i = x;
printf("numero %d", i);

x = 50;
i = x;
printf("numero %d", i);

x = 50;
i = x;
printf("numero %d", i);

for(i=0;i< 10;1++){
// este for NAO DEVE FAZ ABSOLUTAMENTE NADA
};

// k=0:::::
```

ALT + SETAS

ALT + SHIFT + SETAS

CTRL + ALT + SETAS

CTRL + ;

ALT + Z

ALT + CLICK

CTRL + D

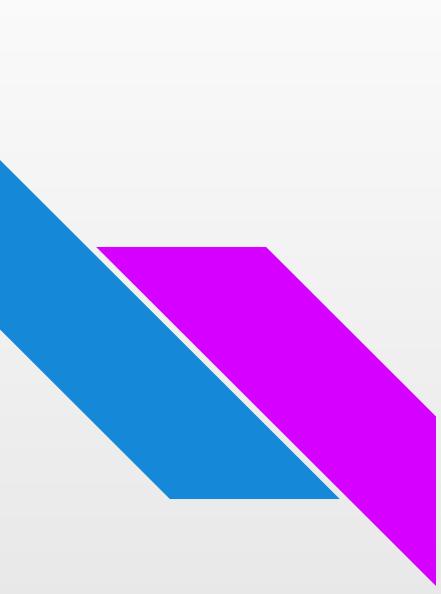




Customização

vs Code

PET.COMP

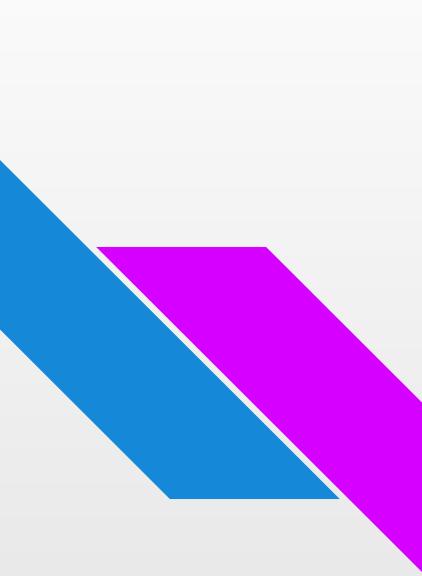


Abrindo a settings.json

- Vamos apertar F1 ou CTRL+SHIFT+P para abrir a paleta de comandos.
- E vamos digitar pesquisar por settings.json e apertar ENTER.



Abrindo a settings.json

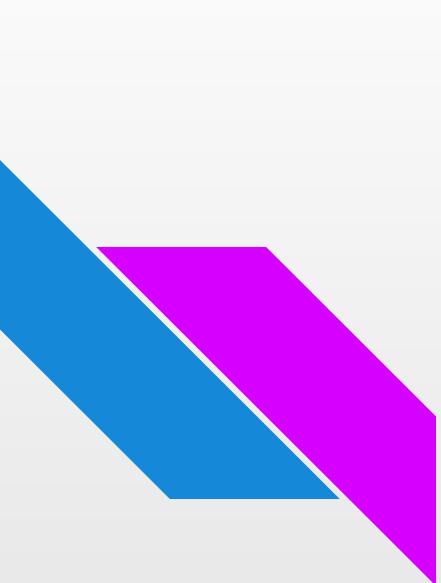


```
{ } settings.json ×  
C: > Users > victo > AppData > Roaming > Code > User > { } settings.json > ...  
1  {  
2      "workbench.startupEditor": "none",  
3      "files.autoSave": "afterDelay",  
4      "workbench.colorTheme": "Visual Studio Dark",  
5      "tabnine.experimentalAutoImports": true,  
6      "window.zoomLevel": 1,  
7      "workbench.iconTheme": "vscode-icons",  
8      "editor.fontLigatures": false,  
9  
10 }
```

Obs: Cada pasta settings.Json de cada pessoa
vai estar de uma maneira.



Abrindo a settings.json



```
{  
    "workbench.startupEditor": "none",  
    "files.autoSave": "afterDelay",  
    "workbench.colorTheme": "Visual Studio Dark",  
    "tabnine.experimentalAutoImports": true,  
    "window.zoomLevel": 1,  
    "workbench.iconTheme": "vscode-icons",  
    "editor.fontLigatures": false,  
  
    "workbench.colorCustomizations": {  
        // Your color customization code here  
    },  
}
```

**Obs: Adicionaremos
"workbench.colorCustomizations"
para começar a customizar.**



Customizando VS code

```
"workbench.colorCustomizations": {  
    "activityBar.background": "#1d77e4", //ActivityBar.background: cor de fundo da barra de atividades.  
    "activityBar.foreground": "#1a0129", // ActivityBar.foreground: cor de primeiro plano da barra de atividades (por exemplo, usada para os ícones).  
    "foreground": "#7ecddb61", // foreground: cor geral de primeiro plano. Esta cor é usada apenas se não for substituída por um componente.  
    "selection.background": "#4d07ff", // selection.background: cor de fundo da seleção de texto em um editor.  
    "icon.foreground": "#ff00f2", // icon.foreground: cor de primeiro plano dos ícones do editor.  
    "focusBorder": "#00ff15", // focusBorder: cor da borda para destacar elementos focados.  
    "textBlockQuote.background": "#b91db2", // textBlockQuote.background: cor de fundo da citação em bloco.  
    "toolbar.activeBackground": "#2528e4", // toolbar.activeBackground: cor de fundo da barra de ferramentas quando ela é clicada.  
    "toolbar.hoverBackground": "#ff0000", // toolbar.hoverBackground: cor de fundo da barra de ferramentas quando ela é passada com o mouse.  
    "button.background": "#ff0000", // button.background: cor de fundo do botão.  
    "button.foreground": "#e6f704", // button.foreground: cor de primeiro plano do botão.  
    "button.hoverBackground": "#03f803", // button.hoverBackground: cor de fundo do botão quando ele é passado com o mouse.  
    "scrollbarSlider.activeBackground": "#00e9faef", // scrollbarSlider.activeBackground: cor de fundo do controle deslizante da barra de rolagem quando ele é clicado.  
    "scrollbarSlider.background": "#ff0000", // scrollbarSlider.background: cor de fundo do controle deslizante da barra de rolagem.  
    "scrollbarSlider.hoverBackground": "#d0ff00" // scrollbarSlider.hoverBackground: cor de fundo do controle deslizante da barra de rolagem quando ele é passado com o mouse.  
},
```

Algumas customizações.



Customizando VS code

The screenshot shows the Visual Studio Code interface with a dark theme. On the left is the sidebar with various icons for file operations like Open, Save, Find, and Settings. The main area has three tabs at the top: settings.json (selected), comparador.c, and Bem-vindo. Below the tabs is a file path: C: > Users > victo > AppData > Roaming > Code > User > {} settings.json > The main content area displays the contents of the settings.json file:

```
1  {
2    "workbench.startupEditor": "none",
3    "files.autoSave": "afterDelay",
4    "workbench.colorTheme": "Visual Studio Dark",
5    "tabnine.experimentalAutoImports": true,
6    "workbench.iconTheme": "vscode-icons",
7    "editor.fontLigatures": false,
8
9    "workbench.colorCustomizations": {
10      "activityBar.background": "#1d77e4", //ActivityBar.background: cor de fundo da barra de atividades.
11      "activityBar.foreground": "#1a0129", // ActivityBar.foreground: cor de primeiro plano da barra de atividades (por exemplo, usada para os ícones).
12      "foreground": "#7ecddb61", // foreground: cor geral de primeiro plano. Esta cor é usada apenas se não for substituída por um componente.
13      "selection.background": "#4d07ff", // selection.background: cor de fundo da seleção de texto em um editor.
14      "icon.foreground": "#ff00f2", // icon.foreground: cor de primeiro plano dos ícones do editor.
15      "focusBorder": "#00ff15", // focusBorder: cor da borda para destacar elementos focados.
16      "textBlockQuote.background": "#b91db2", // textBlockQuote.background: cor de fundo da citação em bloco.
17      "toolbar.activeBackground": "#2528e4", // toolbar.activeBackground: cor de fundo da barra de ferramentas quando ela é clicada.
18      "toolbar.hoverBackground": "#ff0000", // toolbar.hoverBackground: cor de fundo da barra de ferramentas quando ela é passada com o mouse.
19      "button.background": "#ff0000", // button.background: cor de fundo do botão.
20      "button.foreground": "#e6f704", // button.foreground: cor de primeiro plano do botão.
21      "button.hoverBackground": "#03f803", // button.hoverBackground: cor de fundo do botão quando ele é passado com o mouse.
22      "scrollbarSlider.activeBackground": "#00e9faef", // scrollbarSlider.activeBackground: cor de fundo do controle deslizante da barra de rolagem quando ele é clicado.
23      "scrollbarSlider.background": "#ff0000", // scrollbarSlider.background: cor de fundo do controle deslizante da barra de rolagem.
24      "scrollbarSlider.hoverBackground": "#d0ff00" // scrollbarSlider.hoverBackground: cor de fundo do controle deslizante da barra de rolagem quando ele é passado com o mouse.
25    },
26
27
28 }
```

Customizando VS code

```
"editor.tokenColorCustomizations": {  
    "comments": "#FF0000",  
    "variables": "#229977"  
}
```

editor.tokenColorCustomizations é
usado para customizar as cores do seu
código em si.



Customizando VS code

The screenshot shows the Visual Studio Code interface with a dark theme. On the left is the sidebar with various icons for file operations like Open, Save, Find, and Settings. The main area has three tabs at the top: settings.json (selected), comparador.c, and Bem-vindo. Below the tabs is a file path: C: > Users > victo > AppData > Roaming > Code > User > {} settings.json > The main content area displays the contents of the settings.json file:

```
1  {
2    "workbench.startupEditor": "none",
3    "files.autoSave": "afterDelay",
4    "workbench.colorTheme": "Visual Studio Dark",
5    "tabnine.experimentalAutoImports": true,
6    "workbench.iconTheme": "vscode-icons",
7    "editor.fontLigatures": false,
8
9    "workbench.colorCustomizations": {
10      "activityBar.background": "#1d77e4", //ActivityBar.background: cor de fundo da barra de atividades.
11      "activityBar.foreground": "#1a0129", // ActivityBar.foreground: cor de primeiro plano da barra de atividades (por exemplo, usada para os ícones).
12      "foreground": "#7ecddb61", // foreground: cor geral de primeiro plano. Esta cor é usada apenas se não for substituída por um componente.
13      "selection.background": "#4d07ff", // selection.background: cor de fundo da seleção de texto em um editor.
14      "icon.foreground": "#ff00f2", // icon.foreground: cor de primeiro plano dos ícones do editor.
15      "focusBorder": "#00ff15", // focusBorder: cor da borda para destacar elementos focados.
16      "textBlockQuote.background": "#b91db2", // textBlockQuote.background: cor de fundo da citação em bloco.
17      "toolbar.activeBackground": "#2528e4", // toolbar.activeBackground: cor de fundo da barra de ferramentas quando ela é clicada.
18      "toolbar.hoverBackground": "#ff0000", // toolbar.hoverBackground: cor de fundo da barra de ferramentas quando ela é passada com o mouse.
19      "button.background": "#ff0000", // button.background: cor de fundo do botão.
20      "button.foreground": "#e6f704", // button.foreground: cor de primeiro plano do botão.
21      "button.hoverBackground": "#03f803", // button.hoverBackground: cor de fundo do botão quando ele é passado com o mouse.
22      "scrollbarSlider.activeBackground": "#00e9faef", // scrollbarSlider.activeBackground: cor de fundo do controle deslizante da barra de rolagem quando ele é clicado.
23      "scrollbarSlider.background": "#ff0000", // scrollbarSlider.background: cor de fundo do controle deslizante da barra de rolagem.
24      "scrollbarSlider.hoverBackground": "#d0ff00" // scrollbarSlider.hoverBackground: cor de fundo do controle deslizante da barra de rolagem quando ele é passado com o mouse.
25    },
26
27
28 }
```

Customizando VS code

```
"workbench.colorCustomizations": {  
    "[Dracula)": {  
        "sideBar.background": "#5c3deb"  
    }  
},
```

No exemplo acima estou alterando o cor da side bar de um tema em específico por meio da [nome do tema].



Customizando VS code

```
"workbench.colorCustomizations": {  
    "[Dracula)": {  
        "sideBar.background": "#5c3deb"  
    }  
},
```

No exemplo acima estou alterando o cor da side bar
de um tema em específico.

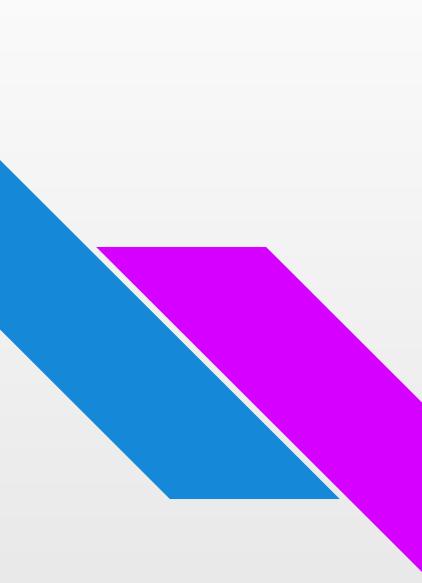


Customizando VS code

```
"editor.tokenColorCustomizations": {  
    "[Dracula Soft)": {  
        "comments": "#FF0000",  
        "variables": "#229977"  
    },  
}
```



Site de referência para customização



OVERVIEW

GET STARTED

EXTENSION CAPABILITIES

EXTENSION GUIDES

UX GUIDELINES

LANGUAGE EXTENSIONS

TESTING AND PUBLISHING

ADVANCED TOPICS

REFERENCES

VS Code API

Contribution Points

Activation Events

Extension Manifest

Built-in Commands

When clause contexts

Theme Color

Product Icon Reference

Document Selector

Theme Color

You can customize your active Visual Studio Code color theme with the [workbench.colorCustomizations](#) user setting.

```
{  
  "workbench.colorCustomizations": {  
    "activityBar.background": "#00AA00"  
  }  
}
```

Note: If you want to use an existing color theme, see [Color Themes](#) where you'll learn how to set the active color theme through the Preferences: Color Theme dropdown ([Ctrl+K Ctrl+T](#)).

Theme colors are available as CSS variables in [webviews](#), and [an extension](#) is available which provides IntelliSense for them.

Color formats

Color values can be defined in the RGB color model with an alpha channel for transparency. As format, the following hexadecimal notations are supported: `#RGB`, `#RGBA`, `#RRGGBB` and `#RRGGBAA`. R (red), G (green), B (blue), and A (alpha) are hexadecimal characters (0-9, a-f or A-F). The three-digit notation (`#RGB`) is a shorter version of the six-digit form (`#RRGGBB`) and the four-digit RGB notation (`#RGBA`) is a shorter version of the eight-digit form (`#RRGGBAA`). For example `#e35f` is the same color as `#ee3355ff`. If no alpha value is defined, it defaults to `ff` (opaque, no transparency). If alpha is set to `00`, the color is fully transparent.

Some colors should not be opaque in order to not cover other annotations. Check the color descriptions to see to which colors this applies.

IN THIS ARTICLE

Color formats

Contrast colors

Base colors

Window border

Text colors

Action colors

Button control

Dropdown control

Input control

Scrollbar control

Badge

Progress bar

Lists and trees

Activity Bar

Profiles

Side Bar

Minimap

Editor Groups & Tabs

Editor colors

Diff editor colors

Chat colors

Inline Chat colors

Editor widget colors

Peek view colors

Merge conflicts colors

Panel colors

<https://code.visualstudio.com/api/references/theme-color>

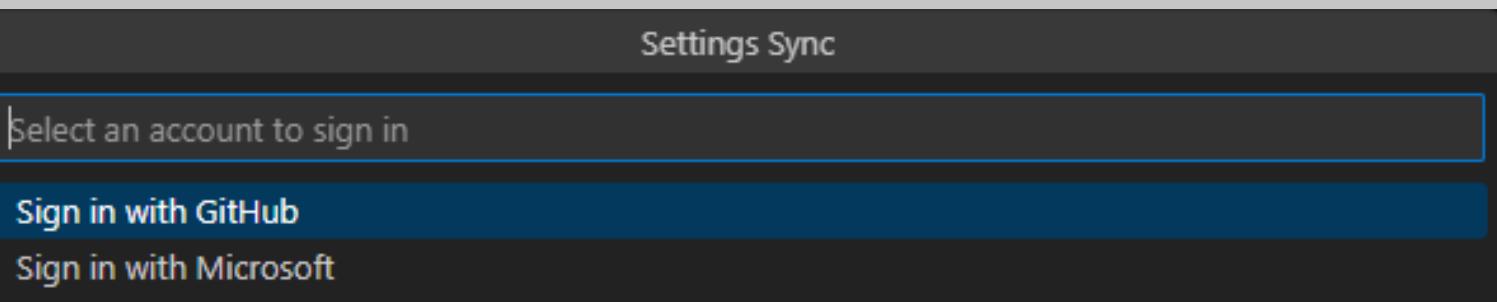
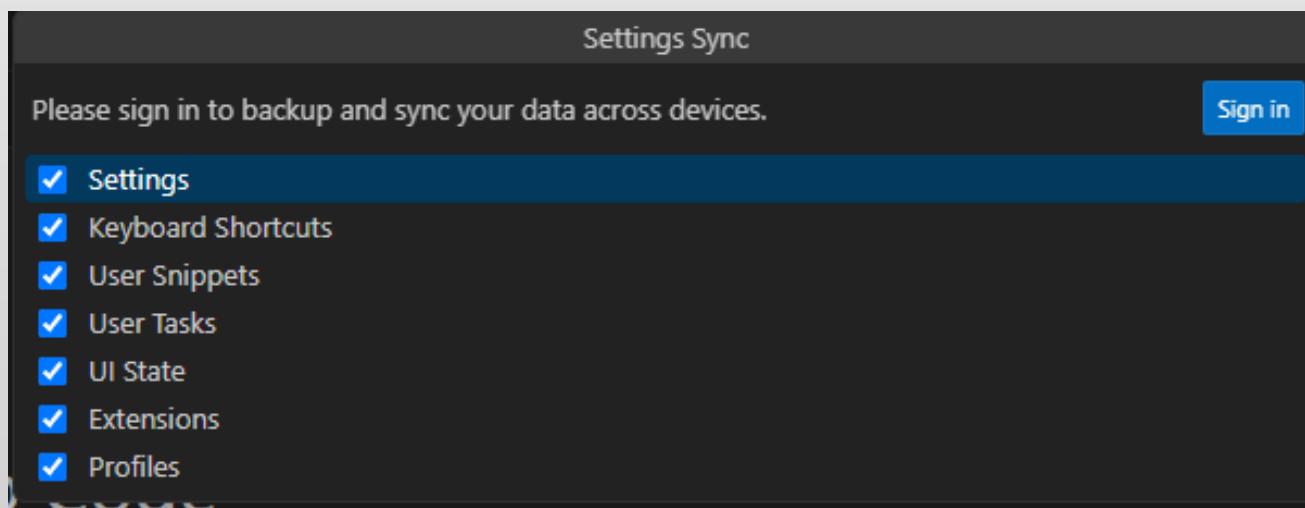
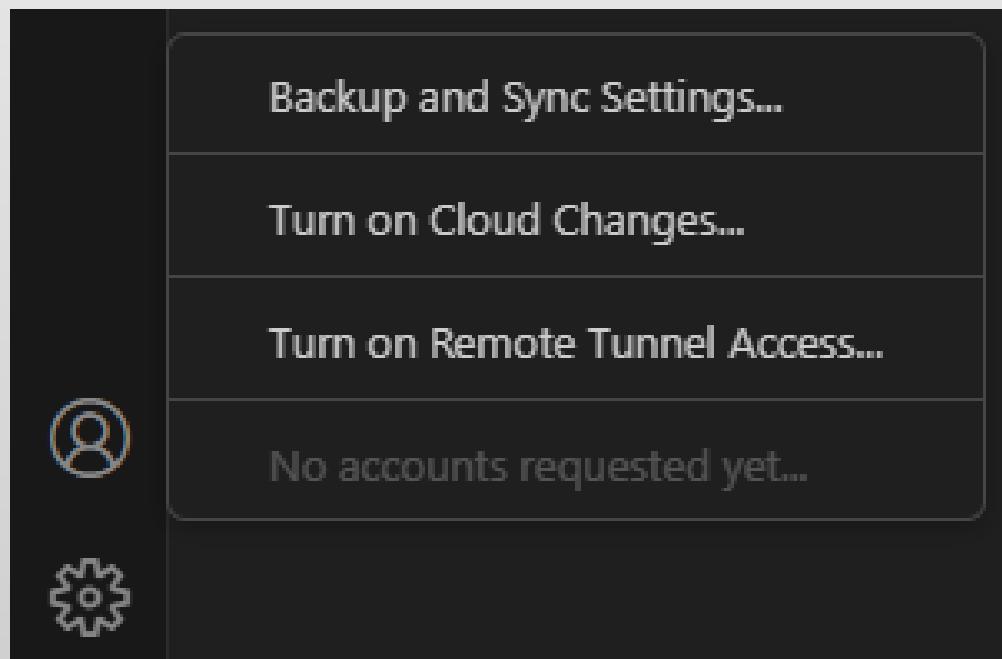


Extensão git no VS Code

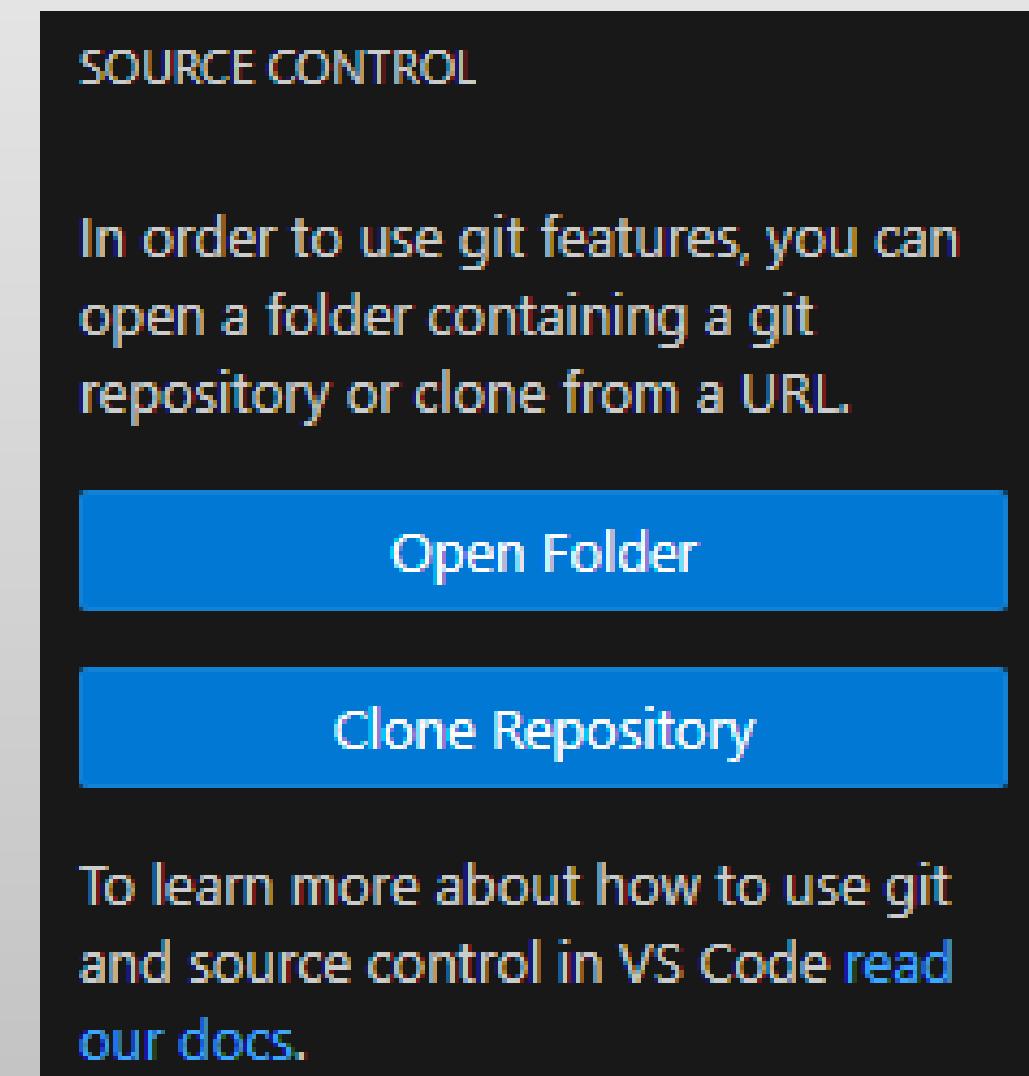
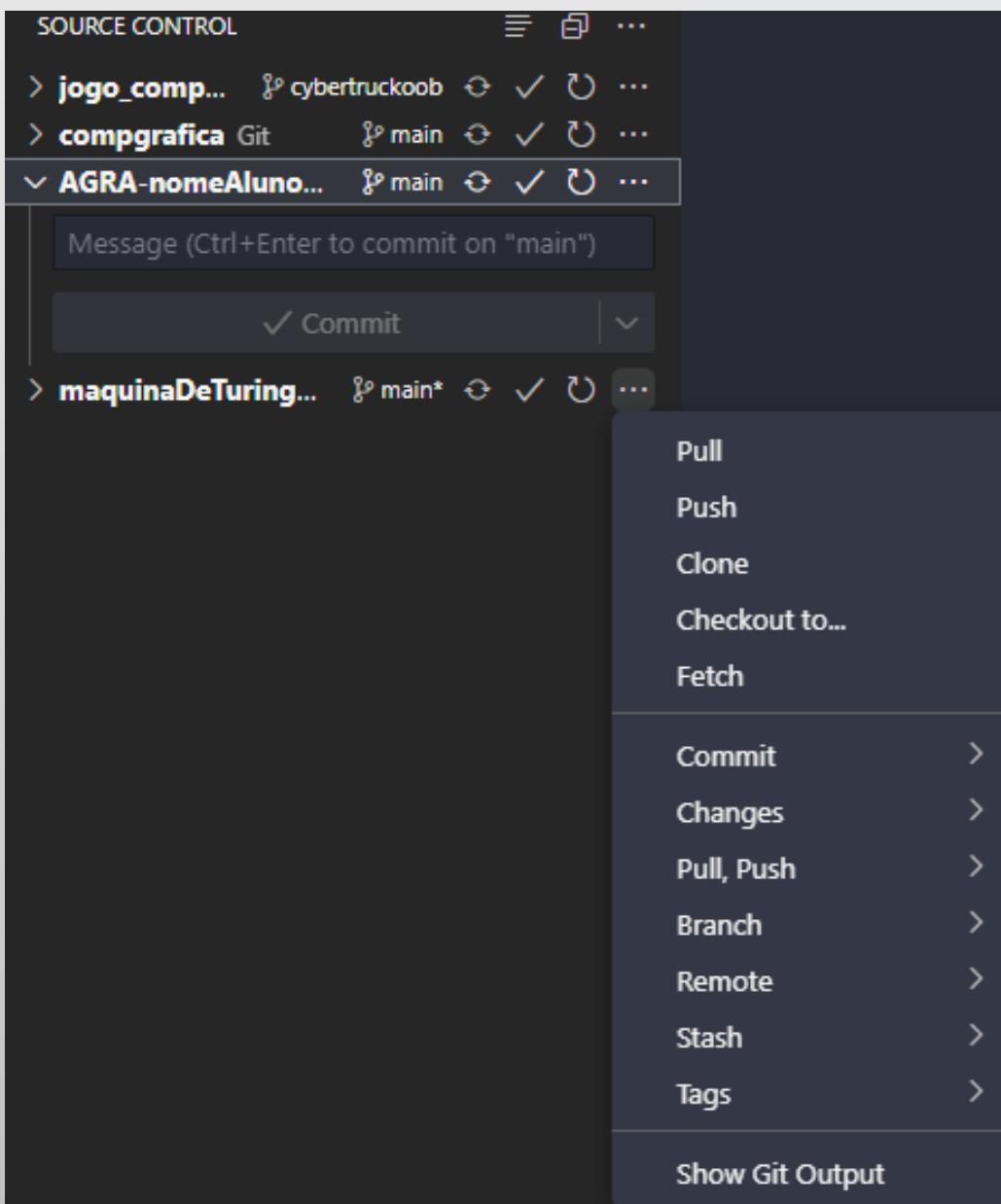


PET.COMP

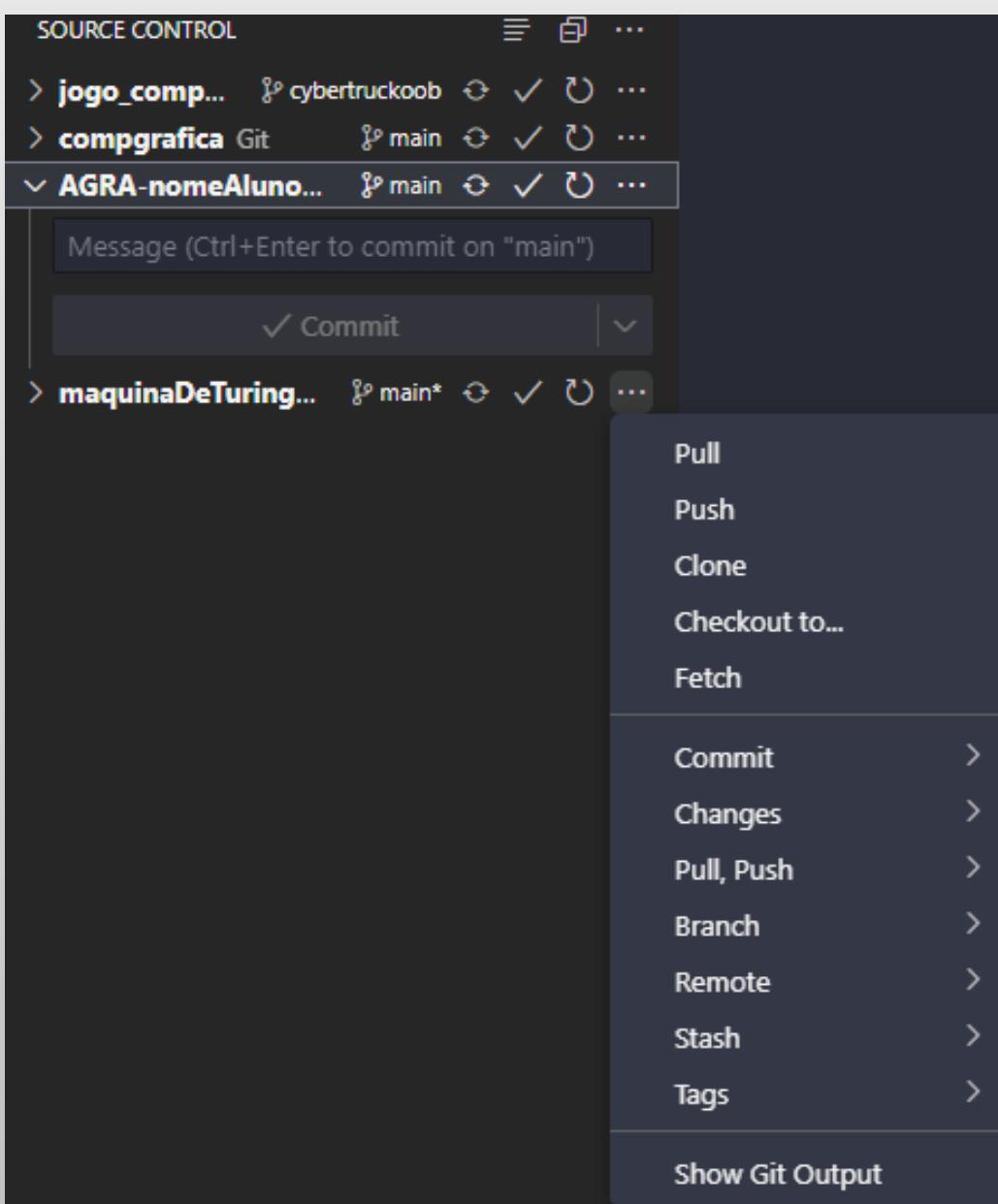
Extensão git no VS Code



Extensão git no VS Code



Extensão git no VS Code



Pull: Traz as alterações de um repositório remoto para o local;

Push: Leva as alterações do repositório local para o remoto;

Clone: Copia um repositório do github em seu computador;

Checkout to: Faz a troca entre as branchs do repositório em seu computador;

Fetch: informa seu git local que ele deve obter as informações de metadados mais recentes do repositório original.





PET.COMP

GitHub Copilot



GitHub Copilot

GitHub Copilot inclui recursos para auxiliar programadores, como:

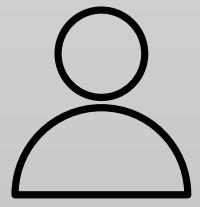
- Conversão de comentários de código em código executável
- Preenchimento automático para trechos de código, seções repetitivas de código e métodos e/ou funções inteiros.



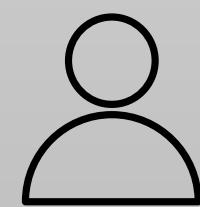
Obrigado



[pet.comp.cefetmg](https://www.instagram.com/pet.comp.cefetmg/)



mateuspereirafernandes@gmail.com



victorsouzavilelasilva@gmail.com



PET.COMP

