

# MÓDULO IV

## Introdução à Programação I

A Educação é o primeiro passo para um futuro melhor...

# MÓDULO IV

## O Que é um Ambiente de Desenvolvimento? Como ele Difere de um Ambiente de Desenvolvimento Integrado (IDE)?

Um ambiente de desenvolvimento é um espaço de trabalho que permite aos desenvolvedores criar um aplicativo ou fazer alterações nele sem afetar a versão ativa do produto de software.

Essas alterações podem incluir manutenção, depuração e aplicação de patches.

Enquanto isso, um ambiente de desenvolvimento integrado (IDE) é usado dentro do ambiente de desenvolvimento para facilitar várias ferramentas de programação em um aplicativo.

Isso ajuda os desenvolvedores a simplificar o processo de desenvolvimento de software.

# MÓDULO IV

De um modo geral, o termo ambiente de desenvolvimento inclui todo o ambiente, desde o desenvolvimento e preparação até os servidores de produção, enquanto um IDE refere-se simplesmente ao aplicativo usado para codificação.

Como o desenvolvimento de aplicativos de software é uma tarefa complexa e demorada, um ambiente de desenvolvimento pode ser uma boa solução para aumentar a eficiência e a produtividade dos desenvolvedores. Por esse motivo, explicaremos um ambiente de desenvolvimento junto com um IDE com mais detalhes, incluindo os motivos para usá-lo e seus diversos tipos.

# MÓDULO IV

## Por Que Usar Um Ambiente De Desenvolvimento?

Dependendo das necessidades do site, a maioria dos ambientes de desenvolvimento contém quatro camadas de servidor, incluindo um servidor de desenvolvimento e um servidor de teste. Todas as alterações são implantadas em diferentes ambientes antes de lançar um site ou aplicativo ao vivo.

Como resultado, muitos desenvolvedores usam ambientes de desenvolvimento para criar um código-fonte e testar se ele funciona no aplicativo sem se preocupar com o impacto nos usuários. Afinal, eles não perceberão nenhuma alteração ou atualização que você fez enquanto navega no site.

# MÓDULO IV

Os desenvolvedores também podem usar esse ambiente de outras maneiras, dependendo do projeto.

Outra vantagem é que ajuda **empresas de desenvolvimento web** e freelancers a conseguirem mais clientes.

Por exemplo, usar um ambiente de desenvolvimento local como XAMPP ou **Docker** ajudará você a criar software para clientes sem serviço de **hospedagem** ou sem acessar o servidor. Isso ocorre porque o aplicativo do conjunto de servidores é armazenado em um dispositivo local, como um computador.

# MÓDULO IV

Além de garantir que o aplicativo funcione conforme o esperado para os usuários finais, outros benefícios que você pode obter de um ambiente de desenvolvimento incluem:

- **Simplificar o fluxo de trabalho.** Um ambiente de desenvolvimento ajuda os desenvolvedores a configurar facilmente cada ferramenta de desenvolvimento — evitando a necessidade de configurá-la separadamente. Isso acelera o processo de desenvolvimento de software.
- **Minimizar possíveis erros.** Os ambientes de teste permitem que os desenvolvedores testem cada código, facilitando a localização e correção de erros.
- **Melhorar a produtividade.** O processo de desenvolvimento é mais direto, permitindo que os desenvolvedores executem várias tarefas com mais rapidez e eficiência. Por exemplo, eles podem analisar o código e a sintaxe durante a edição.
- **Padronização do processo de desenvolvimento.** O uso de uma interface de ambiente de desenvolvimento ajuda vários desenvolvedores a colaborar e economizar tempo.

# MÓDULO IV

## O que é Ambiente de Desenvolvimento Integrado (IDE)?

Um ambiente de desenvolvimento integrado (IDE) é um conjunto de software que combina todas as ferramentas de desenvolvimento em uma **única interface gráfica do usuário (GUI)**. Como resultado, isso torna o processo de desenvolvimento mais rápido e eficiente.

Alguns exemplos de ambientes de desenvolvimento integrado populares são **NetBeans**, **Microsoft Visual Studio**, **Adobe Flex Builder** e **Eclipse**.

# MÓDULO IV

Agora, vamos ver os seguintes recursos em um IDE que podem melhorar a produtividade geral:

- **Editor de código** – usado principalmente para escrever e editar o código-fonte. Muitos IDEs também se integram a um editor de texto, como um **editor de HTML**, que pode destacar palavras-chave e erros de sintaxe.
- **Conclusão de código** – conhecido como prompt de código, esse recurso é usado para analisar todo o código para identificar e inserir componentes de código ausentes. Usá-lo economizará tempo e minimizará bugs ou erros.
- **Compilador** – traduz o texto simples em uma **linguagem** de programação como **JavaScript** e **Python** para que um computador possa processá-lo.

# MÓDULO IV

- **Depurador** – ajuda os desenvolvedores a encontrar e corrigir códigos de erro em um aplicativo ou site durante a fase de teste. Por exemplo, **depurar seu site WordPress** garantirá que sua funcionalidade e desempenho funcionem bem antes de publicá-lo para os usuários.
- **Ferramentas de automação de compilação** – são usadas para automatizar os processos de criação e desenvolvimento de software, como compilar o código-fonte em código de máquina, empacotar o código binário e executar testes automatizados.
- **Suporte à linguagem de programação** – a maioria dos IDEs é dedicada a uma linguagem de programação específica. Por exemplo, o **Xcode** é usado para desenvolver o macOS com as linguagens Objective-C e Swift. No entanto, alguns IDEs também oferecem vários idiomas, como **Eclipse** e **Komodo IDE**.

# MÓDULO IV

- **Controle de versão** – permite que os desenvolvedores rastreiem todas as alterações feitas.
  - Também pode ajudar o IDE a se conectar com o repositório de origem que eles usam.

Além disso, alguns IDEs também podem incluir um navegador de **classes**, um **navegador de objetos** e um **diagrama de classes**.

Observe que um IDE pode ser um aplicativo independente, embora possa fazer parte de um ou mais aplicativos.

# MÓDULO IV

## Tipos de Ambiente de Desenvolvimento Integrado

Como cada projeto pode exigir IDEs diferentes, é importante considerar vários aspectos, como linguagem de programação, facilidade de uso, confiabilidade e tipo. Muitos tipos de IDEs oferecem recursos diferentes para gerar um aplicativo de alta qualidade com eficiência.

Aqui estão vários tipos de ambientes de desenvolvimento integrados:

- **Multilíngue** – suporta programas em vários idiomas que podem ser benéficos para iniciantes melhorarem suas habilidades. Por exemplo, o **Visual Studio** é um dos IDEs multilíngues mais populares que oferece recursos sólidos e oferece suporte a configuração fácil para extensões e atualizações.

# MÓDULO IV

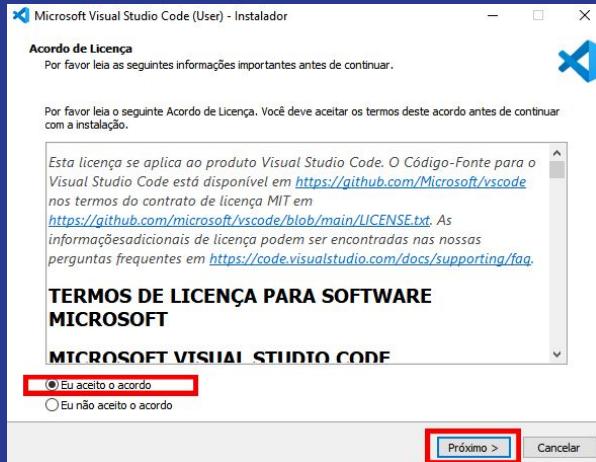
- **Desenvolvimento móvel** – desenvolvido especificamente para desenvolvimento móvel, como **AppCode** e **Android Studio**. Observe que alguns IDEs, especialmente IDEs multilíngues, suportam plug-ins de desenvolvimento móvel, como bate-papo em tempo real e Gerenciador de arquivos.
- **Específico do idioma** – projetado para desenvolvedores de software que trabalham em um idioma. Por exemplo, **Jikes** e **Jcreator** são construídos para Java e **Idle** para Python.
- **Baseado em nuvem** – fornece um recurso integrado para programação em pares em tempo real com feedback instantâneo que permite que os desenvolvedores criem software apenas com um navegador, permitindo que eles acessem o código remotamente.
- **HTML** – usado para desenvolver aplicativos **HTML**, como Notepad++, Atom e Adobe DreamWeaver CC. Essas ferramentas automatizam várias tarefas no desenvolvimento web.

# MÓDULO IV

## Como baixar e instalar o VS Code no Windows?

Para baixar o VS Code, acesse o site oficial da ferramenta e clique em **Download for Windows**.

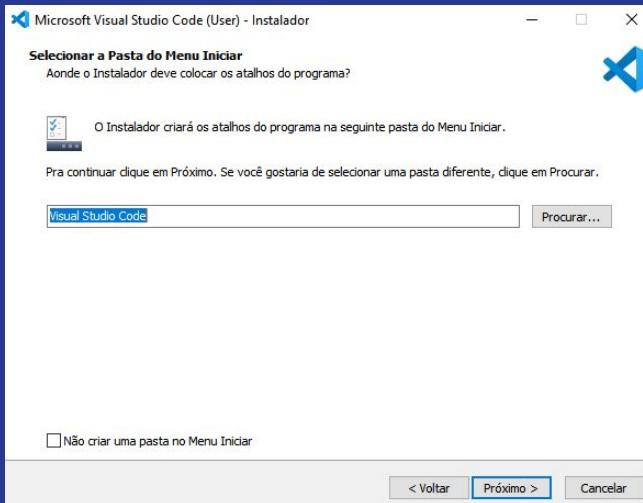
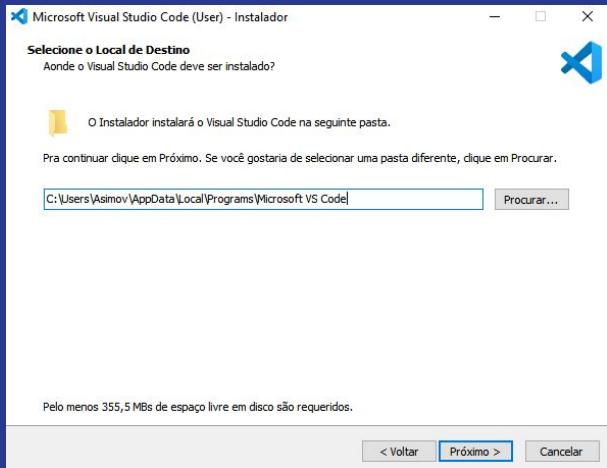
Assim que terminar o download, clique no arquivo para realizar a instalação. Para prosseguir, você precisará aceitar os **termos de uso**.



# MÓDULO IV

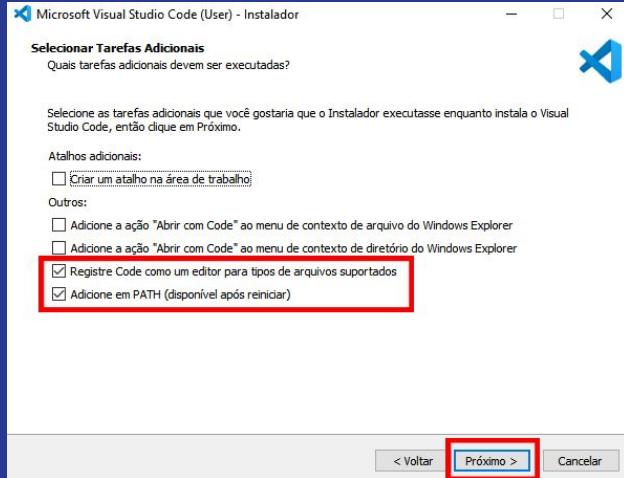
Em seguida, escolha a pasta do seu computador onde será instalado o VS Code.

Na tela seguinte, o instalador notifica que irá criar os atalhos do programa. Se não quiser que isso ocorra, marque a opção **Não criar uma pasta no Menu Iniciar**. Do contrário, apenas clique em **Próximo**.

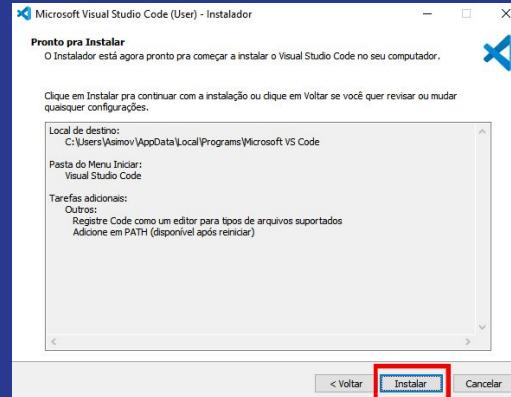


# MÓDULO IV

Agora, você deverá se atentar a um **ponto importante**: marcar a opção **Adicione em PATH**. Isso é necessário para que o software fique disponível nas suas variáveis de ambiente, portanto copie as configurações do exemplo abaixo para funcionar.

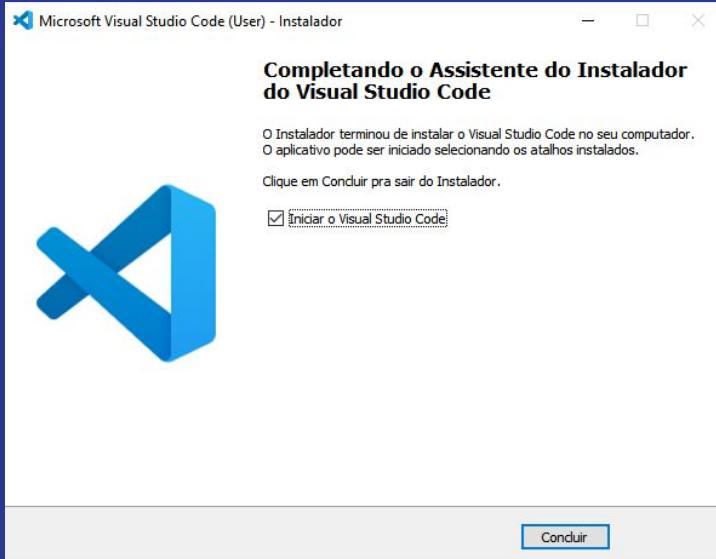


O próximo passo é revisar se todas as configurações estão corretas. Caso sim, clique em **Instalar**; mas, do contrário, clique em **Voltar** e corrija o que for necessário.



# MÓDULO IV

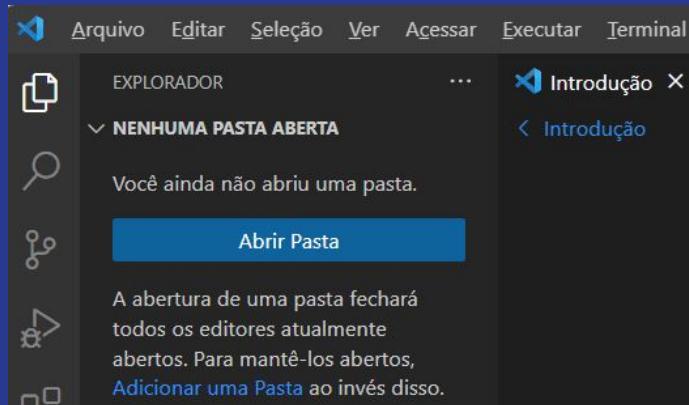
Após a instalação, se tudo ocorrer bem, clique em **Concluir** para abrir o editor.



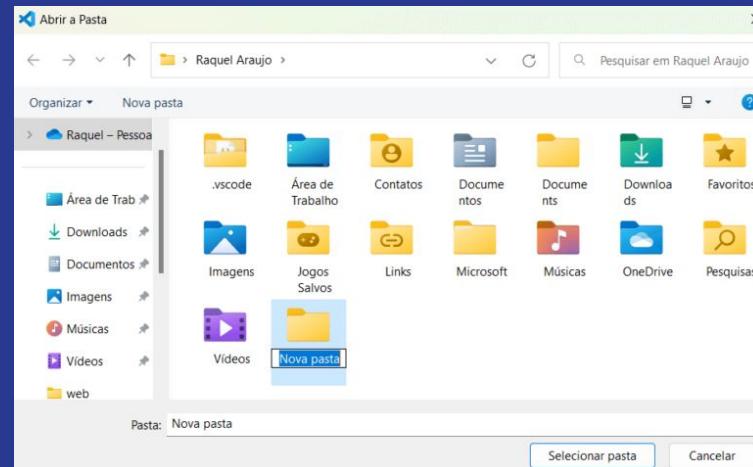
# MÓDULO IV

Concluída a instalação, vai aparecer uma tela simples.

No canto superior esquerdo desta tela temos um botão chamado “Abrir Pasta” nele vamos abrir a pasta para começar a escrever um novo código.



Abrir pasta.

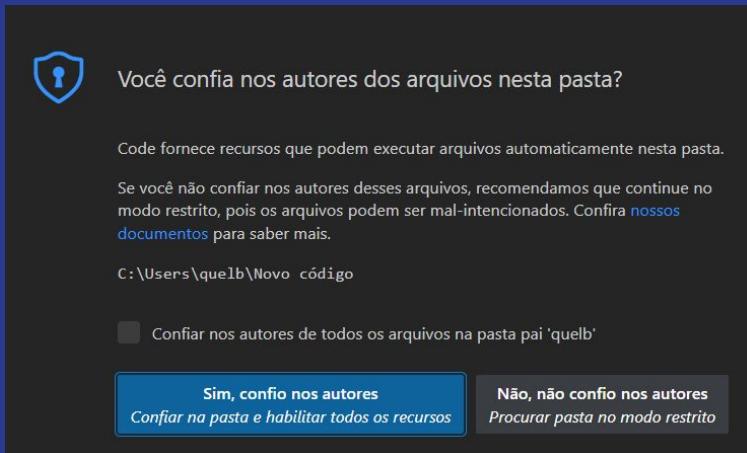


Criando uma pasta.

# MÓDULO IV

Clicando no botão vai se abrir esta imagem acima, crie uma pasta e renomeie, nela vamos começar um novo código.

Após renomear clique em “Selecionar Pasta”.



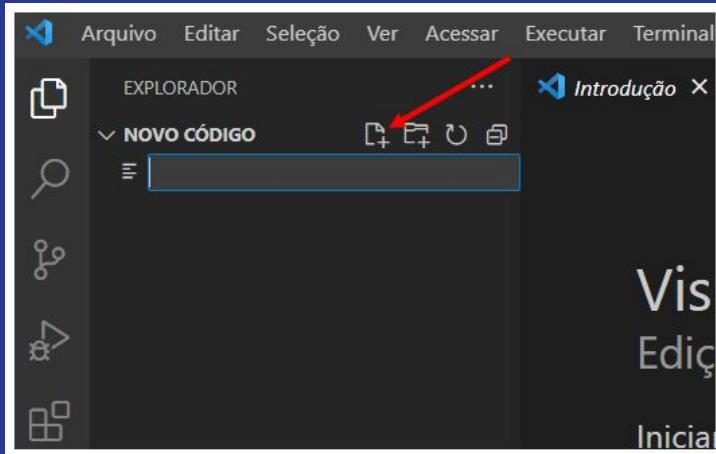
Aceitar.

Vai aparecer esta mensagem, clique em “sim, confio nos autores” porque esta é uma pasta que nós mesmos estamos criando.

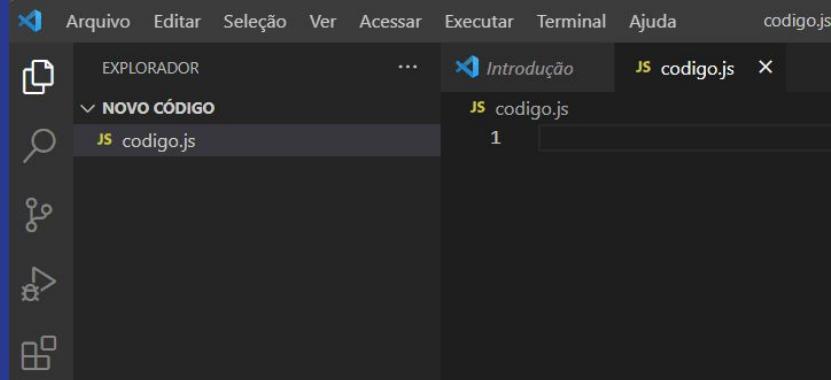
# MÓDULO IV

Na parte superior esquerda passando o mouse vão aparecer algumas opções de ícones como mostra a imagem abaixo.

Clique no ícone indicado para ter acesso a barra de fórmulas.



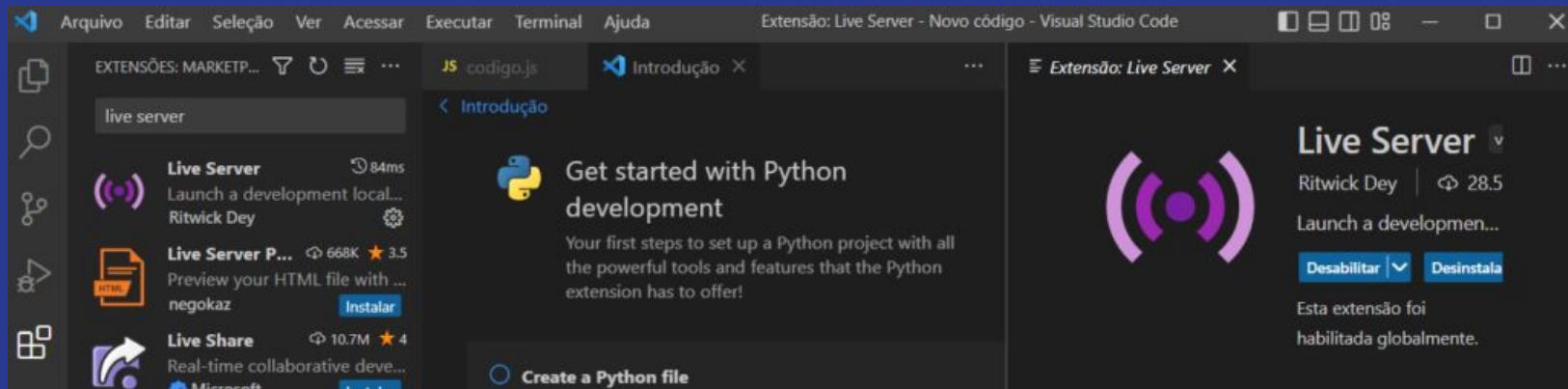
Se escrevermos “código.js” o VS Code já consegue identificar que este código pertence à linguagem JavaScript.



# MÓDULO IV

- Live Server

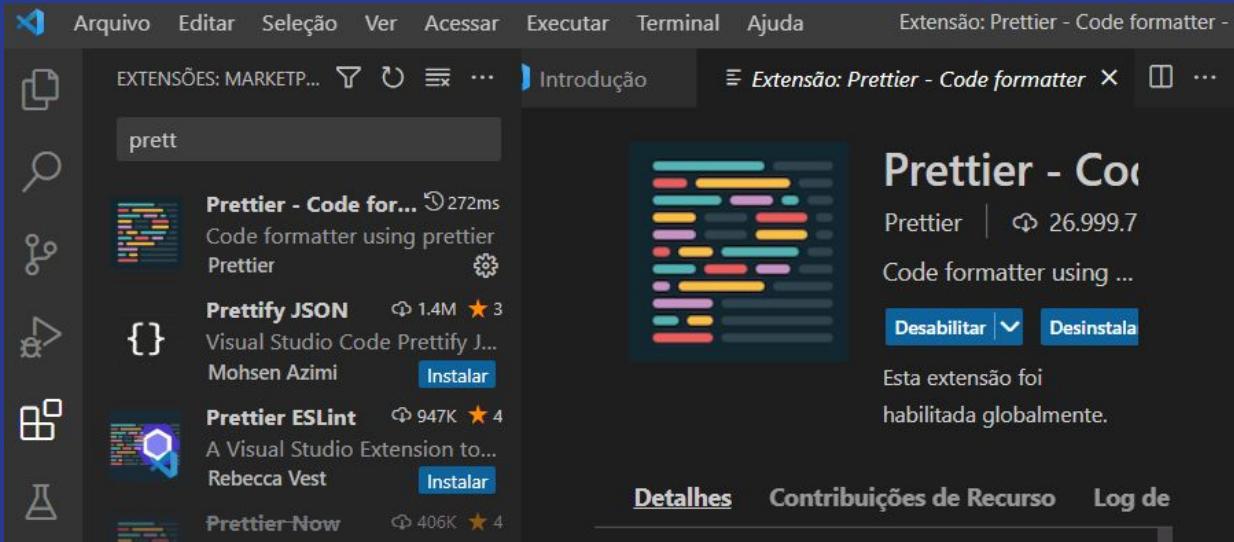
Esta extensão é vital para visualizar o site durante a sua criação de uma forma rápida e dinâmica.



# MÓDULO IV

- Prettier

Por último deixo a dica de baixar a extensão do Prettier, esta aplicação deixa o código mais bonito e organizado, mais fácil de ler.



Pronto! Agora está mais fácil iniciar ou aplicar seus conhecimentos em programação!

# MÓDULO IV

EXERCÍCIOS...

# MÓDULO IV

FIM...