

# Instalação

Essa página vai te guiar através dos passos para a instalação, e tudo que você precisa para começar a programar.

## O que você precisa?

Para começar você vai precisar de:

- Um computador com Windows 10 ou Mac Os
- Acesso a internet e permissões de administrador
  - o Você só vai precisar dessas permissões para fazer a instalação. Você não vai precisar de nenhum acesso especial para escrever nem executar os códigos.
- Um cartão microSD
  - o Você vai precisar de no mínimo 4GB e no máximo 32GB de capacidade de armazenamento. Esse tipo de cartão microSD também é conhecido como microSDHC. Nós recomendamos cartões com Performance de Aplicação Classe A1.
- Um leitor de cartão microSD.
  - o Se o seu computador não tiver um leitor de cartão microSD, você pode usar um leitor externo USB cartão microSD.
- Um cabo Mini-USB, assim como o que vem incluso no conjunto EV3.

A configuração típica desse equipamento é descrita na figura 1.

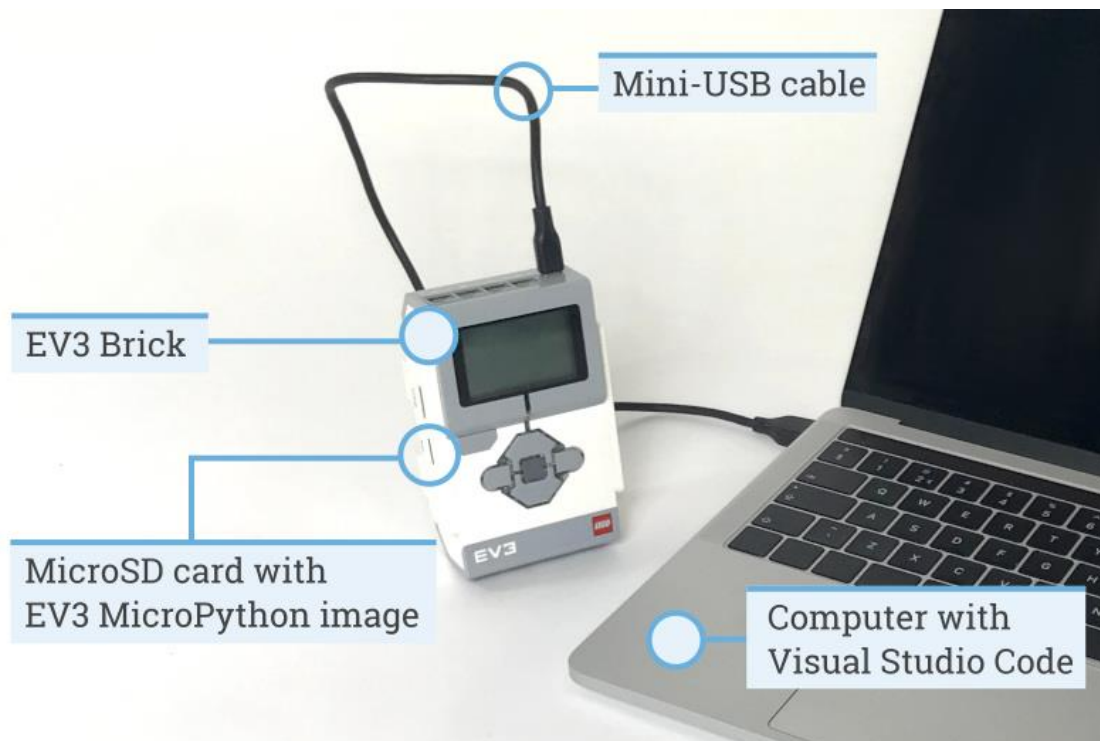


Figura 1: Visão Geral do Equipamento

## Preparando o seu computador.

Você vai escrever seus códigos em MicroPython usando Visual Studio Code. Siga os passos abaixo para baixar, instalar e configurar essa aplicação:

1. Baixe o Visual Studio Code.
2. Siga as instruções na tela para instalar o programa
3. Execute Visual Studio Code.
4. Abra a aba de extensões.
5. Instale a extensão EV3 MicroPython como mostrado na Figura 2.

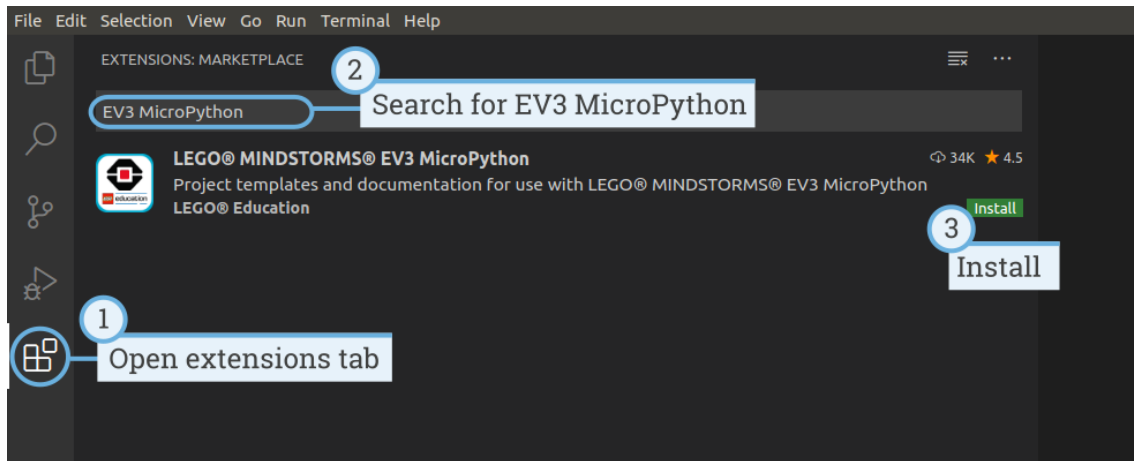


Figura 2: Instalando a extensão da aba de extensões do Visual Studio Code.

## Preparando o MicroSD.

Para fazer com que seja possível executar programas MicroPython no seu EV3 Brick, agora você vai aprender como instalar as ferramentas necessárias no seu cartão MicroSD.

Se o cartão MicroSD contém arquivos que você deseja guardar, tenha certeza de criar um backup antes. Acesse managing files on the EV3 para aprender a fazer um backup dos seus programas MicroPython anteriores se necessário.

Esse processo apaga tudo no seu cartão MicroSD. Incluindo programas MicroPython anteriores.

Para instalar as ferramentas MicroPython no seu cartão MicroSD:

1. Instale o EV3 MicroPython microSD card image e salve ele em uma localização conveniente. Esse arquivo contém aproximadamente 360MB. É necessário descompactar o arquivo para selecionar a imagem mais à frente.
2. Baixe e instale a ferramenta de para transformar seu cartão MicroSD em bootavel chamada Etcher.
3. Insira o cartão MicroSD no seu computador ou no leitor de cartão.
4. Execute a ferramenta e siga os passos para a instalação. Se você for usar o Etcher, você pode seguir as instruções abaixo, assim como mostrado na figura abaixo.

5. Selecione o arquivo de imagem EV3 MicroPython microSD que você acabou de baixar.
6. Selecione seu cartão microSD. Tenha certeza de que esse dispositivo e capacidade correspondam ao seu cartão microSD.
7. Comece o processo para inserir a imagem no cartão microSD. Esse processo pode demorar alguns minutos. Não retire o cartão até que o processo tenha acabado.

Figura 3: Usando o Etcher para inserir o arquivo de

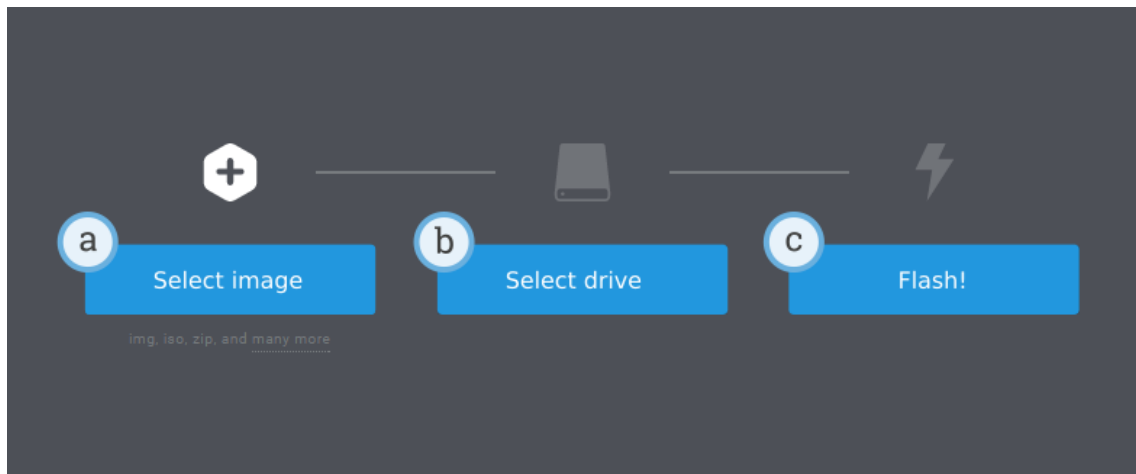


imagem EV3 MicroPython microSD no cartão microSD

## Atualizando o cartão microSD.

Para atualizar o cartão microSD, baixe um novo arquivo de imagem usando o arquivo de imagem insira no seu cartão microSD como descrito acima. Tenha certeza de fazer backup dos arquivos MicroPython que você deseja salvar.

Você não precisa de apagar o conteúdo do cartão microSD antes. Isso é feito automaticamente ao inserir a nova imagem.

# Usando o EV3 Brick.

Tenha certeza de que o EV3 Brick esteja desligado. Insira o cartão MicroSD que você preparou na abertura de cartão MicroSD no EV3 Brick, assim como mostrado na Figura 4:

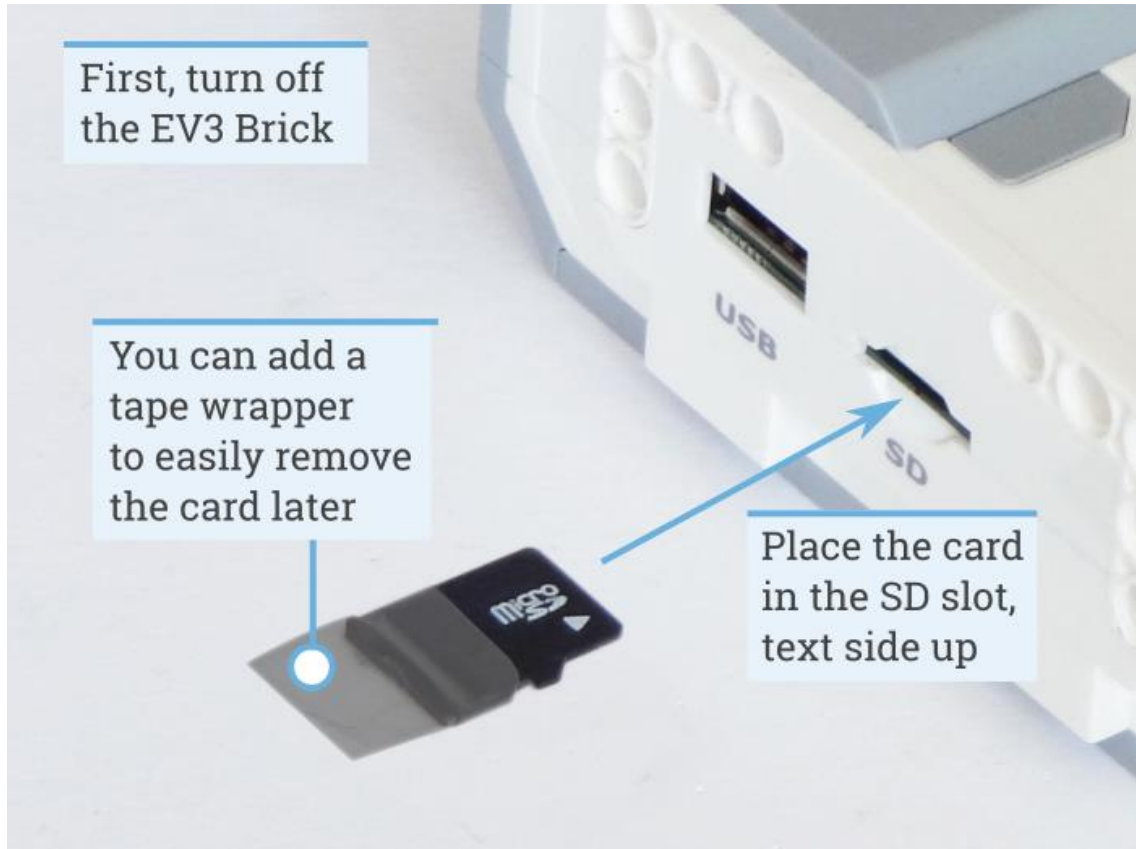


Figura 4: Inserindo o cartão MicroSD no EV3 Brick.

# Ligando e Desligando o EV3 Brick.

Para ligar o EV3 Brick pressione o botão cinza escuro no centro.

O processo de iniciação pode demorar alguns minutos. Enquanto liga o EV3 Brick a luz de status liga na cor laranja e pisca intermitentemente e você pode ver muitos textos na tela do EV3. O EV3 Brick está pronto para o uso quando a luz de status ficar verde.

Para desligar o EV3 Brick, abra o menu de desligamento pressionando o botão mais afastado, e selecione Power Off usando o botão central. Assim como mostrado na Figura 5:

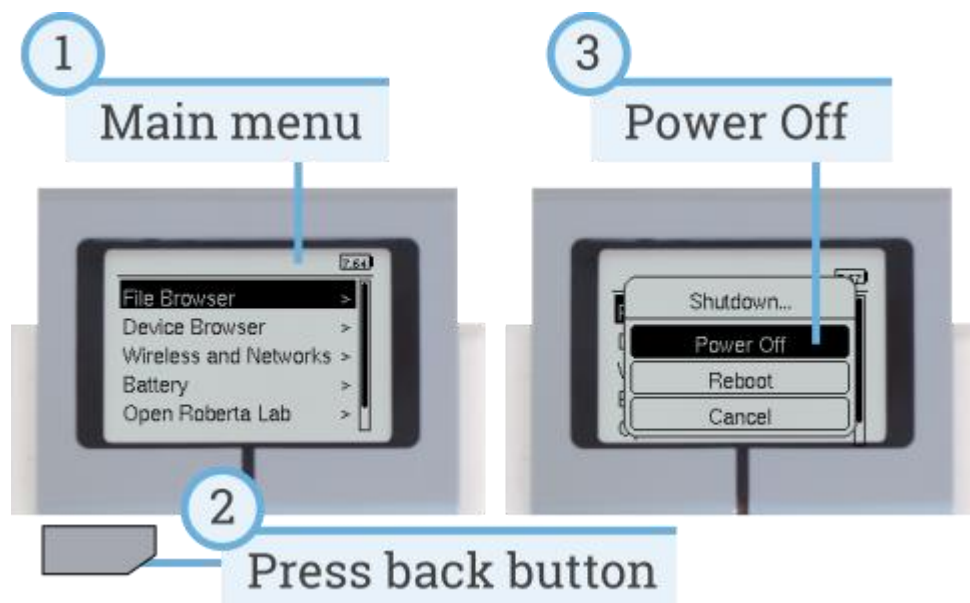


Figura 5: Desligando o EV3 Brick.

## Enxergando os valores dos motores e Sensores.

Quando você não está executando nenhum programa, você consegue enxergar os valores dos sensores e dos motores o Device Browser, como mostrado na Figura 6:

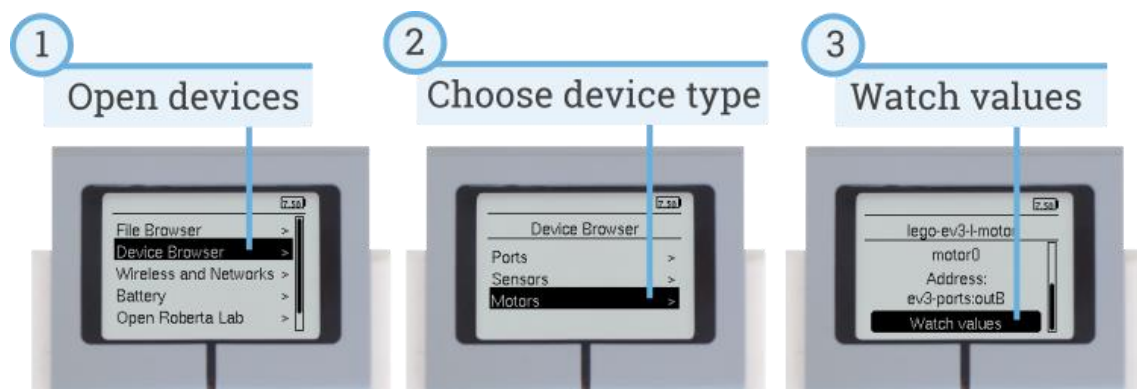


Figura 6: Vendo os valores dos motores e sensores.

# Executando o programa sem um computador.

Você pode executar programas previamente baixados direto do EV3 Brick.

Para fazê-lo, encontre o programa usando o File Browser na tela do EV3 e pressionar o botão central para começar o programa assim como mostrado na Figura 7:

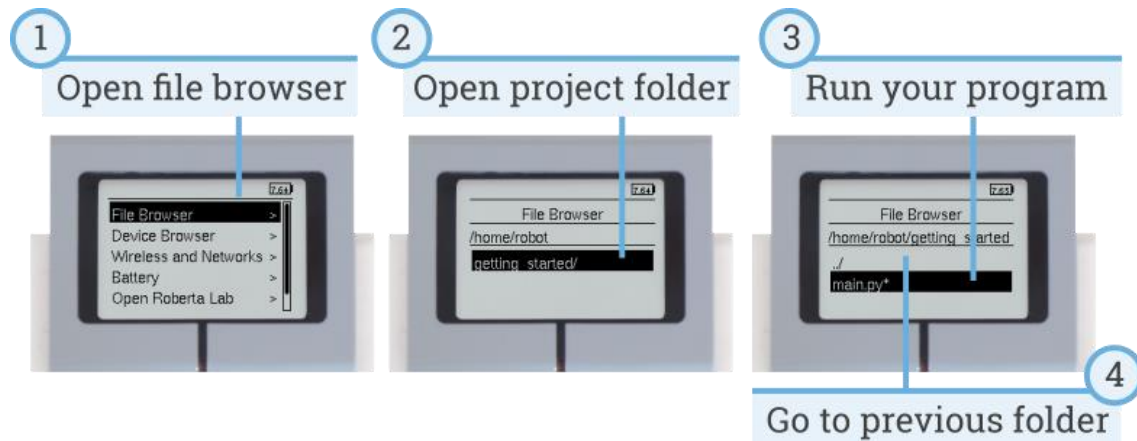


Figura 7: Executando um programa usando os botões do EV3 Brick.

# Voltando para o Firmware original.

Você pode voltar para o firmware original do Lego e os seus programas Lego a qualquer hora. Para fazê-lo:

1. Desligue o EV3 Brick como mostrado acima.
2. Espere até que a tela e a luz de status se apaguem.
3. Remova o cartão MicroSD.
4. Ligue o EV3.

# Criando e Executando os programas.

Agora você configurou seu computador e seu EV3 Brick, você está pronto para começar a escrever os seus programas.

Para fazer isso mais fácil de criar e gerenciar seus programas, vamos primeiro dar uma rápida olhada em como projetos e programas MicroPython são organizados para o seu robô EV3.

Programas são organizados em *pastas de projeto*, assim como mostrado na Figura 8. Uma pasta de projeto, é um diretório no seu computador que contém os principais programas (**main.py**) e outros arquivos e scripts opcionais. Esta pasta de projetos e tudo que está dentro dela são copiados para dentro do EV3 Brick, onde o programa principal vai ser executado.

Essa página vai te mostrar como criar e transferir seu projeto para dentro do EV3 Brick.

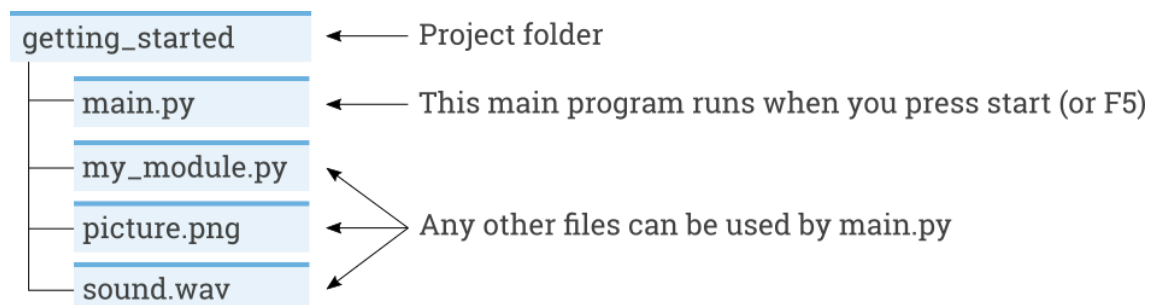


Figura 8: Um projeto contém um programa chamado **main.py** e recursos adicionais como sons ou módulos MicroPython.



# Criando um projeto

Para criar um projeto, abra a aba EV3 MicroPython e clique em criar, assim como mostrado na Figura 9. Insira um nome para o projeto no campo de texto e pressione Enter. Quando pressionado, escolha uma localização para seu programa e confirme clicando em Escolher Pasta.

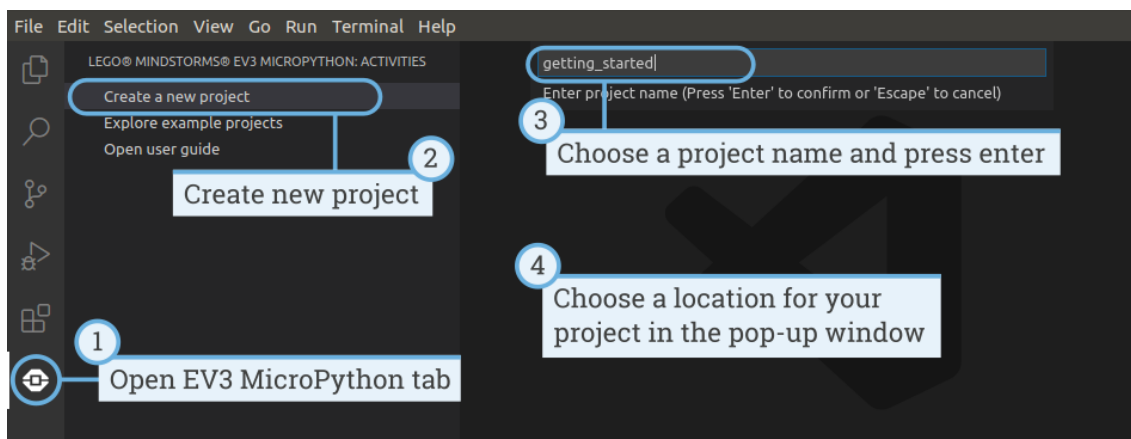


Figura 9: Criando um projeto. Esse exemplo é chamado de `getting_started`, mas você pode escolher qualquer nome.

Quando você cria um novo projeto, ele já possui o arquivo `main.py`. Para ver o conteúdo e modificá-lo, abra-o pelo buscador assim como mostrado na Figura 10. É assim que você vai escrever seus códigos.

Se você é novo na programação em MicroPython, nos recomendamos que você comece pelos códigos existentes e depois adicione o seu.

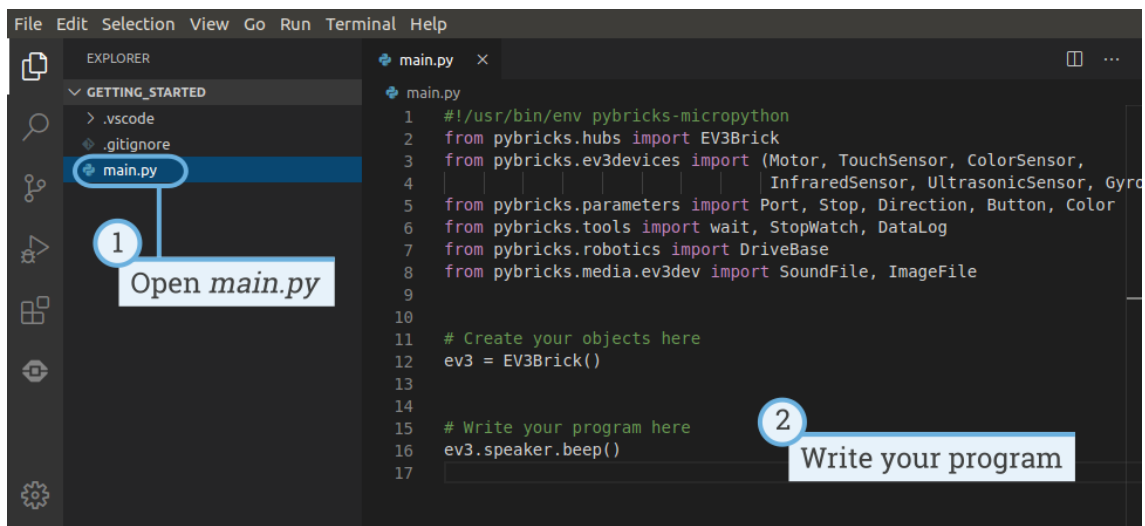


Figura 10: Abrindo o programa padrão main.py.

## Abrindo um programa existente.

Para abrir um programa que você criou anteriormente, clique em File e clique em Open Folder, assim como mostrado na Figura 11. Depois, navegue para seu projeto criado anteriormente e clique em OK. Você também pode abrir seu projeto recém usado usando o Open Recent no menu de Opções.

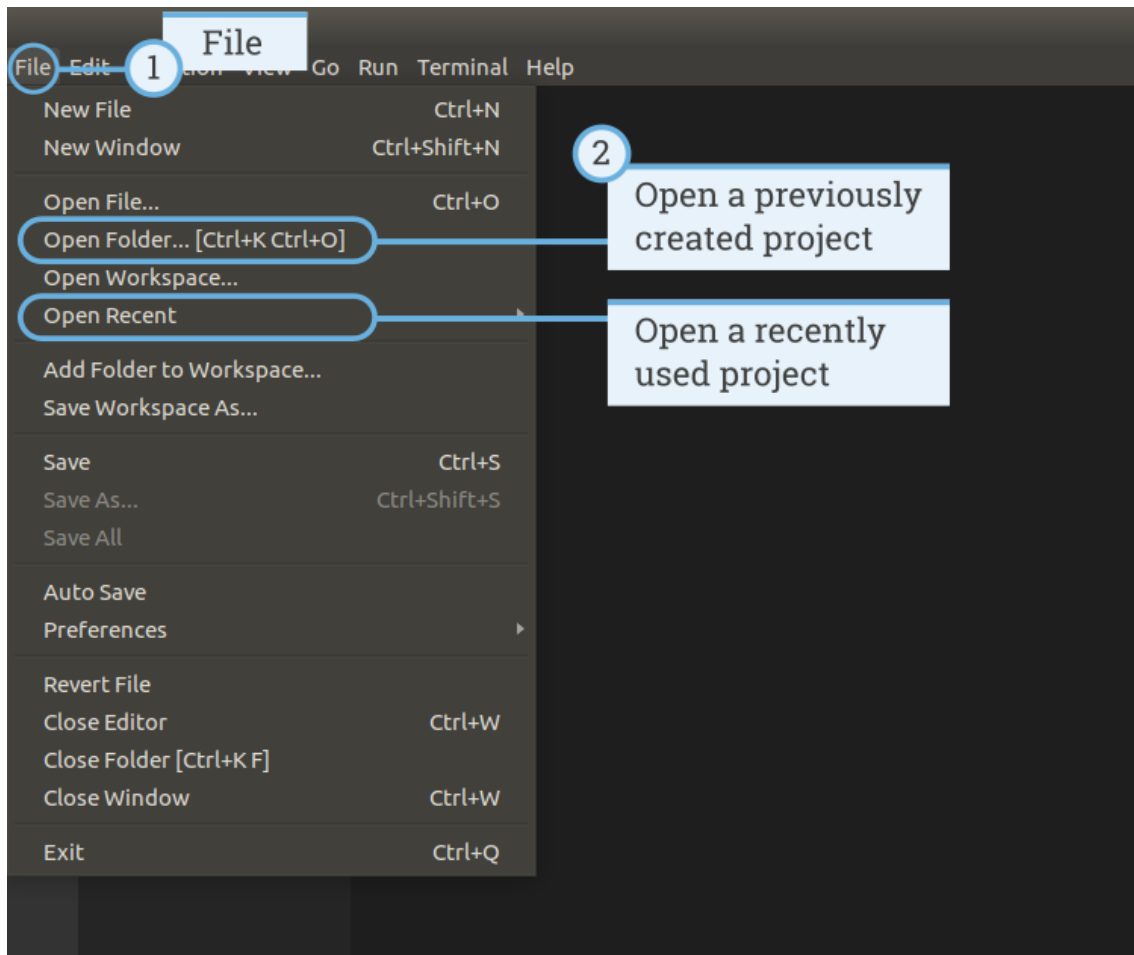


Figura 11: Abrindo um projeto já criado.

# Conectando o EV3 Brick com o Visual Studio Code.

Para ser permitida a transferência do seu código para o Ev3 Brick, você vai primeiro precisar conectar o EV3 Brick ao seu computador com o cabo Mini-USB e configurar a conexão com o Visual Studio Code. Para fazê-lo:

- Ligue o EV3 Brick.
- Conecte o EV3 Brick ao seu computador com o cabo Mini-USB
- Configure a conexão USB como mostrado na Figura 12.

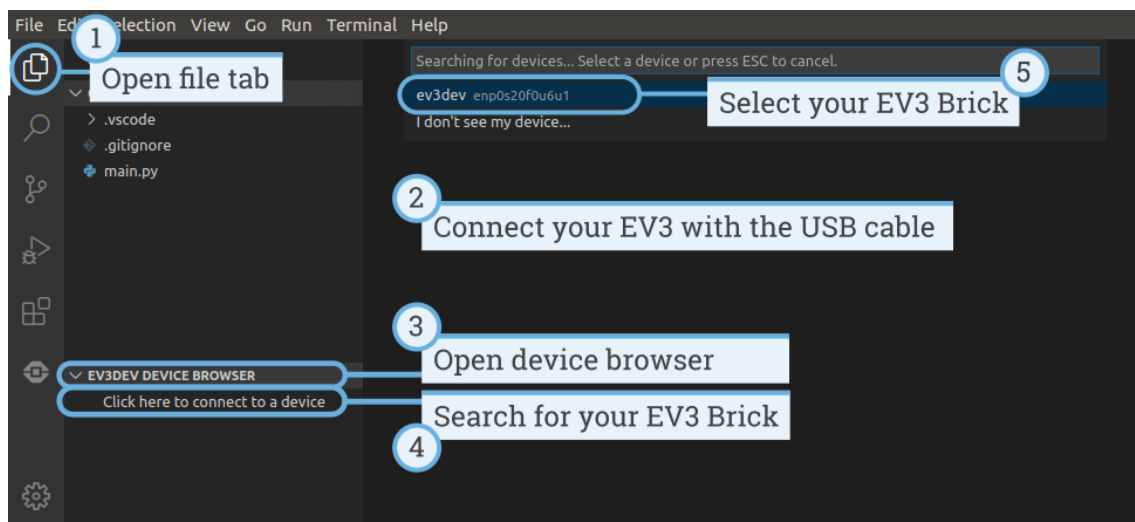


Figura 12: Configurando a conexão USB entre o computador e o EV3 Brick.

# Baixando e executando um programa.

Você pode pressionar o botão F5 para executar o programa. Ou também, você pode executá-lo manualmente, indo a aba de *debug* e clicando na seta verde de start, como mostrado na Figura 13.

Quando o programa começar, uma barra de ferramentas irá aparecer na tela permitindo que você pare o programa se necessário. Você também pode pausar o programa a qualquer hora pressionando o botão mais afastado no EV3 Brick.

Se seu programa produz qualquer saída com o comando `print`, ele será mostrado na janela de saída.

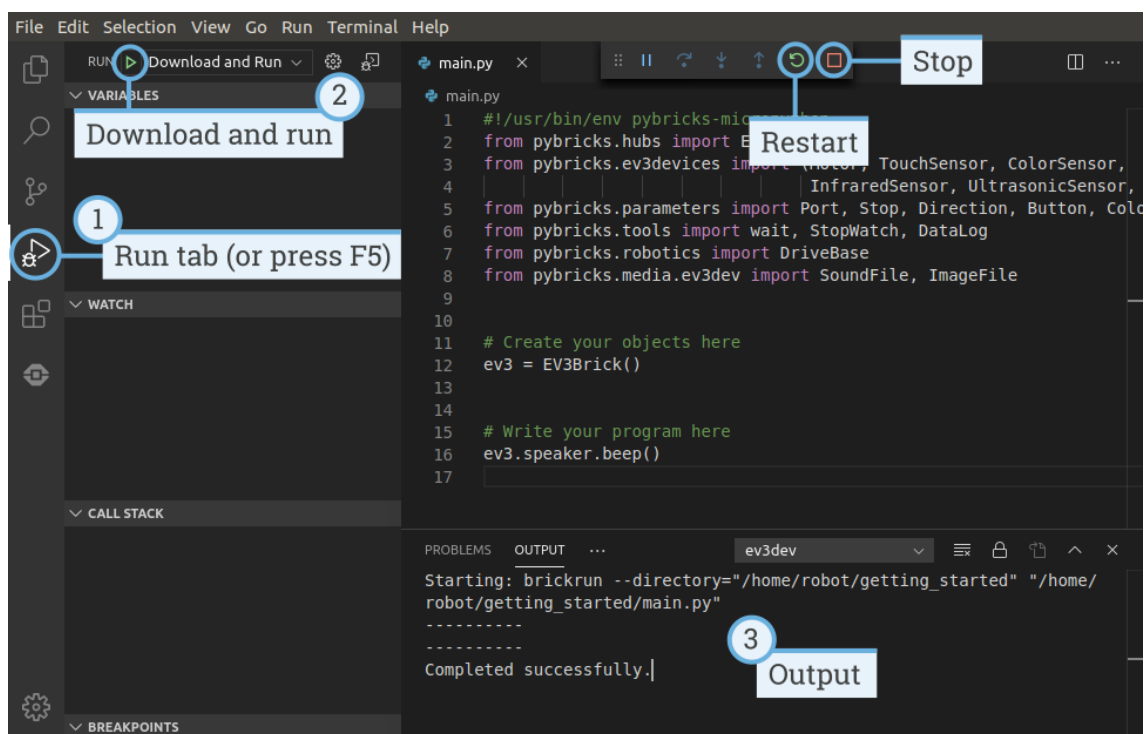


Figura 13: Executando um programa

# Expandindo o programa de exemplo.

Agora que você executou um código básico, você pode expandir o programa fazendo um motor se mover. Primeiramente, conecte um Motor Grande na Porta B no EV3 Brick, como mostrado na Figura 14.

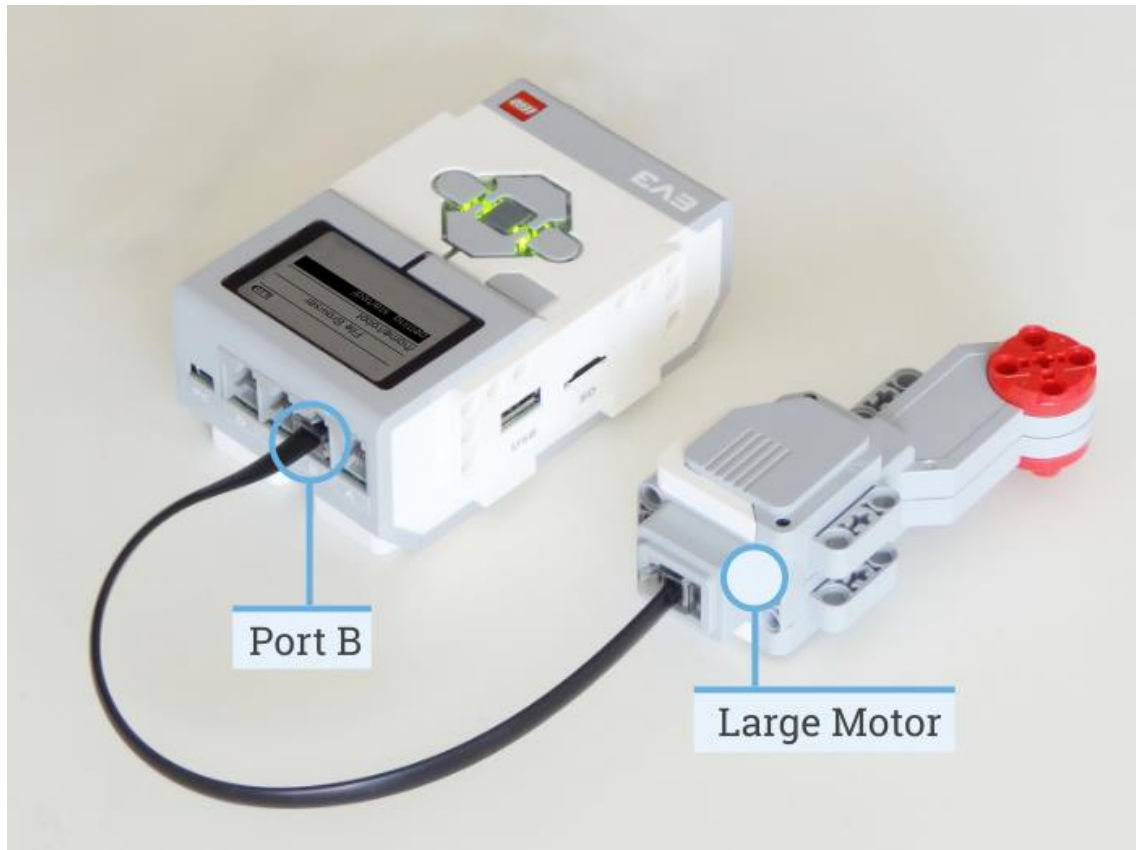


Figura 14: o EV3 Brick com um Motor Grande conectado na Porta B.

A seguir, editar o código `main.py` para fazê-lo se parecer com esse:

```
#!/usr/bin/env pybricks-micropython
from pybricks.hubs import EV3Brick
from pybricks.ev3devices import Motor
from pybricks.parameters import Port

# Crie seu próprio objeto aqui.

# Inicializar o EV3 Brick.
ev3 = EV3Brick()

# Inicializar um motor na Porta B.
test_motor = Motor(Port.B)

# Escreva seu programa aqui.

# Toque um som.
ev3.speaker.beep()

# Acione o motor em 500 graus por segundo. Para um angulo alvo de 90 graus.
test_motor.run_target(500, 90)

# Toque outro som de beep.
ev3.speaker.beep(frequency=1000, duration=500)
```

Esse programa faz o robô tocar um som de beep, move o motor e faz outro som de beep com um tom mais agudo. Execute o programa para ter certeza de que ele funcionou como esperado.

## Organizando arquivos no EV3 Brick.

Depois de baixar um projeto para o EV3 Brick, você pode executá-lo, deletá-lo ou fazer backup e programas guardados usando o Device Browser como mostrado na figura 15.

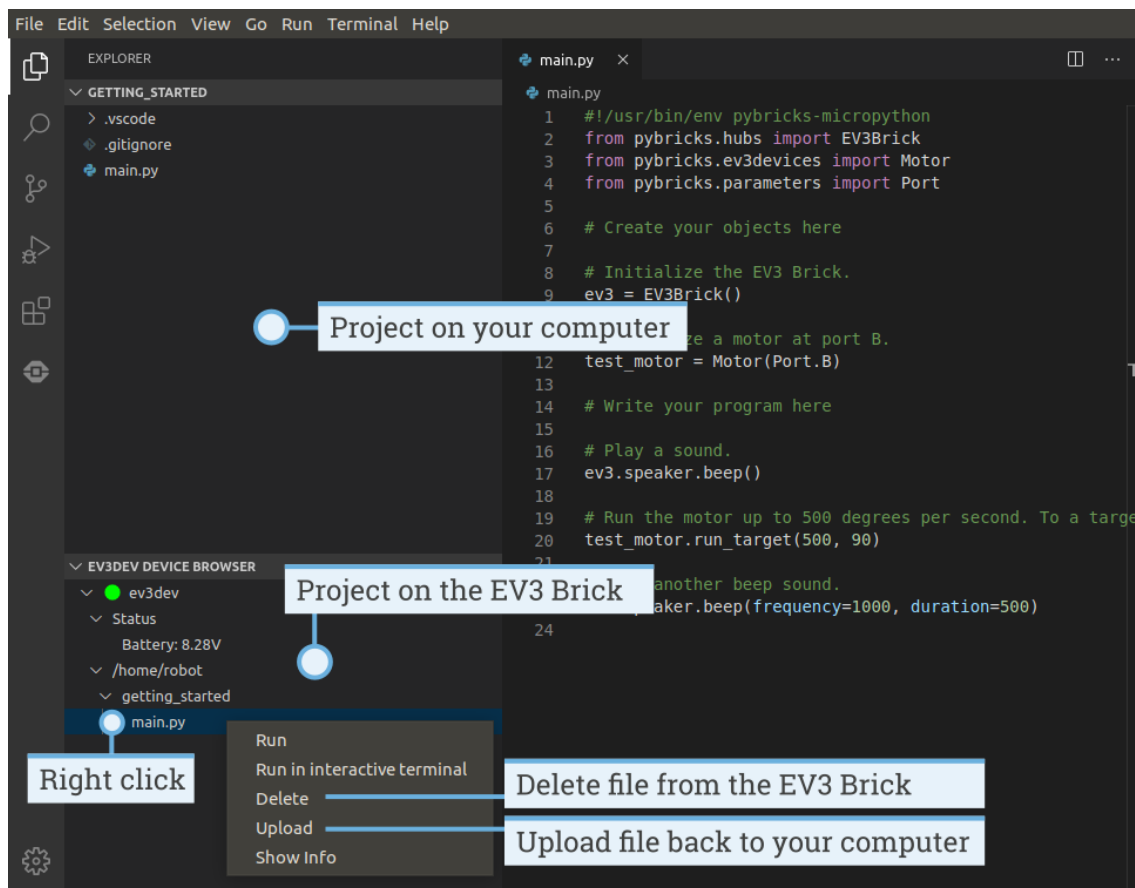


Figura 15: Usando o Device Browser para organizar os arquivos no seu EV3 Brick.