

# Configurando um Projeto Flask no Padrão MVC

1. **Execute um Contêiner MySQL:** Após a instalação do Docker, você pode executar um contêiner MySQL. Você pode especificar uma imagem oficial do MySQL no Docker Hub e criar um contêiner a partir dela. Execute o seguinte comando no terminal:

bashCopy code

```
docker run -d --name mysql-container -e MYSQL_ROOT_PASSWORD=sua_senha -p 3306:3306 mysql:latest
```

Substitua `sua_senha` pela senha que deseja definir para o usuário root do MySQL. Esse comando irá baixar a imagem mais recente do MySQL e criar um contêiner com o MySQL dentro dele.

2. **Acessando o MySQL:** Com o contêiner em execução, você pode acessar o MySQL utilizando um cliente MySQL ou o próprio terminal. Para acessar o MySQL a partir do terminal, você pode executar o seguinte comando:

bashCopy code

```
docker exec -it mysql-container mysql -uroot -p
```

Ele solicitará a senha que você definiu no passo anterior.

Agora, você tem um ambiente MySQL leve em execução dentro de um contêiner Docker isolado. Isso facilita a implantação e a remoção do MySQL, além de manter seu sistema operacional mais limpo. Certifique-se de que o Docker esteja em execução sempre que você quiser acessar o MySQL. Quando não precisar mais do contêiner MySQL, você pode pará-lo e removê-lo com comandos Docker adequados.

## Iniciando com o virtualenv

Primeiro, execute esse comando em seu prompt de comando ou no terminal:

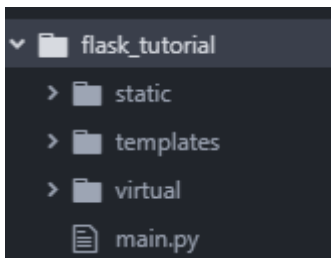
```
pip install virtualenv
```

Segundo, faça o seguinte:

```
virtualenv "nome do ambiente virtual"
```

Aqui, você pode dar um nome para o ambiente. Eu geralmente dou a ele o nome de virtual. Assim, o comando normalmente é: `virtualenv virtual **. **`

Depois de configurar o ambiente virtual, confira sua pasta do projeto. Ela deve ter essa aparência. O ambiente virtual precisa ser criado no **mesmo diretório onde estão localizados os arquivos da sua aplicação**.



Aparência do diretório

## Ativando o ambiente virtual

Agora, vá para seu terminal ou prompt de comando. Vá para o diretório que contém o arquivo chamado **activate**. O arquivo **activate** é encontrado dentro de uma pasta chamada **Scripts** no **Windows** e **bin** no **OS X** e no **Linux**.

Para os ambientes OS X e Linux:

```
$ name of virtual environmnet/bin/activate
```

Para o ambiente Windows:

```
name of virtual environment\Scripts\activate
```

Como eu sou um usuário do Windows, ao ativar o ambiente, ele terá essa aparência:

```
(virtual) C:
```

É isso que você deve ver no início de sua linha do prompt de comando

A próxima etapa é instalar o Flask em seu ambiente virtual para que possamos executar a aplicação em nosso ambiente. Execute o comando:

```
pip install flask
```

Execute a aplicação e vá para <http://localhost:5000/>

Fizemos, finalmente, nossa aplicação para a web. Agora, queremos mostrar ao mundo todo o nosso projeto (mais informações sobre o virtualenv podem ser encontradas nos guias sobre o [virtualenv](#) e na [documentação oficial do Flask](#) - textos em inglês).