



Introdução ao Python





Instrutor: Vitor Mazuco

<http://facebook.com/vitormazuco>

Email:**vitor.mazuco@gmail.com**

WebSite:

<http://vmzsolutions.com.br>



Criando Pacotes

Para ter uma estrutura melhor na hora de organizar os projetos, é preciso criar os pacotes com diversos arquivos que trabalham numa mesma função. Vamos seguir esses passos

- `mkdir modulos`
- `cd modulos`
- `touch __init__.py`



Criando Pacotes

Os módulos em Python são divididos por diretórios dentro de seu projeto. Na versão 2 do Python é preciso criar um arquivo `__init__.py`, dentro do diretório criado, pois assim o python possa entender que existem uma relação de importação. Já no python 3, esse tipo de arquivo não é necessário.



Criando Pacotes

Um exemplo de nossa estrutura poderia ser assim:

Projeto/

 / model

 / __init__.py

 / model.py

 / template

 / index.html

 / view

 / views.py



Criando Pacotes

Nesse caso, é necessário informar o nome do pacote e depois o nome do script. Vamos criar uma pasta com o nome de Módulos, aí dentro dele, mover o arquivo `utils.py` para dentro dela, criar o arquivo `__init__.py` assim o arquivo `main.py` ficaria diferente. (Veja o arquivo **main2.py**)



Criando Pacotes

Você pode perceber que ao executar esse comando, foi gerado arquivos `.pyc` que contêm o *bytecode* compilado de arquivos de origem Python. O interpretador Python carrega arquivos `.pyc` antes de arquivos `.py`, portanto, se eles estiverem presentes, ele pode economizar algum tempo ao não ter que recompilar o código fonte do Python novamente.



Criando Pacotes

Você pode apagar eles se quiser, mas eles não causam problemas, eles não são grandes, e eles podem economizar algum tempo ao executar programas, isso ajuda muito na hora de executar vários desses arquivos ao mesmo tempo.



Criando Pacotes

Você ainda pode renomear os módulos importados, para não dar conflito na hora de programar em python. Ex:

```
from modulos import utils as outro_nome
```

```
outro_nome.escreverArquivo()
```

```
outro_nome.lerArquivo()
```