

Ambientes virtuais de desenvolvimento

Virtual environments

Introdução

Quando estamos trabalhando com Python, é comum que precisemos de bibliotecas para desenvolver o nosso código. Um projeto pode depender de apenas uma ou várias bibliotecas para que seja executado. Pense nas seguintes situações:

01. Você trabalha em vários projetos simultaneamente, e alguns utilizam versão X de uma biblioteca, enquanto outros usam a Y. Isso é bem comum, pode ser que em um projeto anterior tenha usado a versão 1.1 da lib e em um novo precise usar a 1.9. Nesse cenário com VirtualEnv você consegue manter os dois funcionando em paralelo.
02. Você tem um servidor onde hospeda vários softwares. Os mais antigos precisam de versões velhas de certas bibliotecas, e você não quer ter que atualizar e arriscar quebrar as aplicações. Com VirtualEnv todos têm suas dependências isoladas, assim o projeto que foi feito 2 anos atrás pode continuar rodando sem problemas com a versão 1 da biblioteca X, e os mais novos podem usar a versão 99.

A VirtualEnv basicamente cria uma cópia de todos os diretórios necessários para que um programa Python seja executado, isto inclui:

- As bibliotecas built-in do Python
- O gerenciador de pacotes pip
- O próprio Python (Python 2.x/3.x)
- As bibliotecas baixadas pelo pip
- Seu código

Assim, ao instalar uma nova dependência dentro do ambiente criado pelo VirtualEnv, ele será colocado no diretório site-packages relativo à esse ambiente, e só esse ambiente enxergará essa dependência.

Passo a passo

1. Inicializando um ambiente virtual:

\$ python -m venv <nome da env>

2. Ativando o ambiente:

- a. No terminal do Windows:

\$ env\Scripts\activate

- b. No terminal do Linux/MacOS:

\$ source env/bin/activate