

Python Fundamentals



Jupyter

Anotações		

Jupyter Notebooks

Objetivos da Aula

- ✓ Compreender o que é Jupyter notebooks.
- ✓ Conhecer suas funcionalidades.
- ✓ Como instalar e executar.
- Configurar primeiro acesso.
- ✓ Como utilizar.

Jup	vter	Notebooks

Nesta aula apresentaremos o jupyter notebooks, como realizar sua instalação e configuração e utilização.

Anotações			

4LINUX

Jupyter Notebooks

Trata-se de uma aplicação web, que facilita o entendimento e visualização de dados e resultados de análises, juntamente com o código.



Jupyter Notebooks

Facilita a experimentação, colaboração e publicação online. Anteriormente conhecido como IPython notebooks, atualmente permite o uso de outras linguagens, Python porém é o default. Jupyter Notebook constitui uma das ferramentas ideais, como um aplicativo da web, para ajudá-lo a obter as habilidades de ciência de dados necessárias, permitindo criar e compartilhar documentos que contenham código ativo, equações, visualizações e texto.

Anotações		

4LINUX Caderno Jupyter?

- ✔ Produzidos pelo aplicativo, "Notebook" ou "documentos de notebook", denotam registros contendo elementos de código, texto, figuras, links, equações, etc.
- ✔ Devido à combinação de código e elementos textuais, esses documentos consistem meio ideal para reunir uma descrição de análise e seus resultados.
- ✓ Também podem executar a análise de dados em tempo real.

Anotações	



- ✓ Ao executar o documento, o kernel associado, inicia-se automaticamente executa o cálculo, produzindo os resultados.
- ✔ Dependendo do tipo de computação, consumirá significativamente CPU e RAM.
- /
- ✔ Observe que a RAM não é liberada até que o kernel seja desligado.

K	Д	r	n	Р	ı
$I \setminus$	$\overline{}$	ı.	ш	ᆫ	ı

O kernel de notebook é um "mecanismo computacional" que executa o código contido em um documento de Notebook, sendo acionando automaticamente na iniciação do documento, pode ser executado, célula por célula ou com o menu Cell -> Run All

Anotações			

4LINUX Instalando

Por padrão o jupyter notebook rodará na porta 8888:

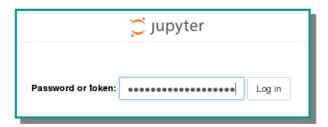
http://127.0.0.1:8888

- Instale o jupyter, execute o comando: 1
 - # pip3 install jupyter
- Execute o jupyter: 2
 - # jupyter notebook --allow-root

Anotações	

4LINUX Executando a aplicação

```
root@stark:/home/jonsnow# jupyter notebook --allow-root
[I 16:56:12.109 NotebookApp] Serving notebooks from local directory: /home/jonsnow
[I 16:56:12.109 NotebookApp] The Jupyter Notebook is running at:
[I 16:56:12.109 NotebookApp] http://localhost:8888/?token=27c6f6e80eca591db472ab21050b3dd0d1e6d7
cd7b4d7a00
    16:56:12.109 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice
to skip confirmation).
[C 16:56:12.110 NotebookApp]
     Copy/paste this URL into your browser when you connect for the first time,
     to login with a token:
           http://localhost:8888/?token=27c6f6e80eca591db472ab21050b3dd0d1e6d7cd7b4d7a00
```

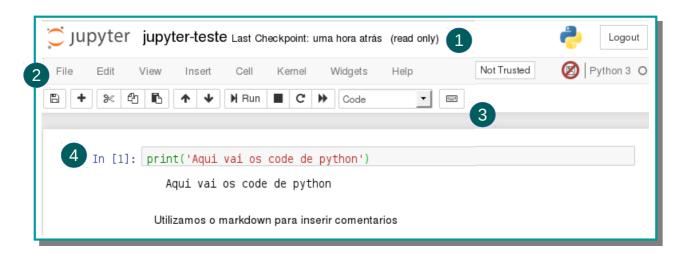


Execução do jupyter notebook

Ao iniciar a aplicação jupyter notebook dentro de um diretório, será gerado um log no terminal especificando a url de acesso e um token que será usado como chave de acesso.

Ao acessar o localhost na porta especificada, haverá um campo no qual será necessário colocar o token para logar ou, poderá acessar a url informada no log com parâmetro do token, obtendo o mesmo resultado.

A	n	0	ta	ιÇ	õ	e	S



Utilizamos cerquilha para títulos

Título

In [1]: nome do notebook e a informação de quando foi salvo pela última vez. Um notebook é qualquer arquivo de apresentação criado no Jupyter com a extensão .ipynb. A tela apresentada no slide, por exemplo, é um notebook.

In [2]: barra de menus.

In [3]: barra de botões. Atalhos para algumas das rotinas mais utilizadas do Jupyter.

In [4]: célula—ou Cell, em inglês. É aqui que a magia acontece.

OBS.: No campo da célula podemos criar scrips python, comentários, e campos de texto.

Anotações

```
# Cada bloco é executado individualmente.
In [1]: # Bloco que calcula o factorial
        resultado = 1
        factorial = 5 # número a ser calculado
        for x in range(1, factorial+1):
           resultado *= x
       print(resultado)
           120
         os blocos de codigo tem acesso as variaveis uns aos outros.
In [2]: print(resultado)
           120
In [3]: # Bloco que calcula a Raiz quadrada
        raiz = 64 # número a ser calculado
       print(raiz ** 0.5)
```

Célula

Definimos um Campo de texto como título, utilizamos # para definir um título.

In [1]: Determinamos um bloco de código que calcula o fatorial de um número inteiro, lembrando que cada bloco de código é executado individualmente.

Estabelecemos um outro bloco de texto.

In [2]: Definimos um bloco de código, embora cada bloco seja executado individualmente, podemos acessar variáveis de outros blocos.

In [3]: Determinamos um bloco de código que calcula a raiz quadrada de um número.

Obs.: Experimente executar cada bloco de código separadamente, uma das principais funções da ferramenta é que podemos criar blocos na nossa célula para analisar dados e tirar a dependência de um bloco por outro.

Anotações