

---

# 15- Jupyter Notebooks: Usando e Desenvolvendo

Demo tutorial

Adriano, Michel

A few topics:

- Cloning Jupyter notebook on LineA using GitHub;
- Some functions were applied to DES Y5-Y6 data

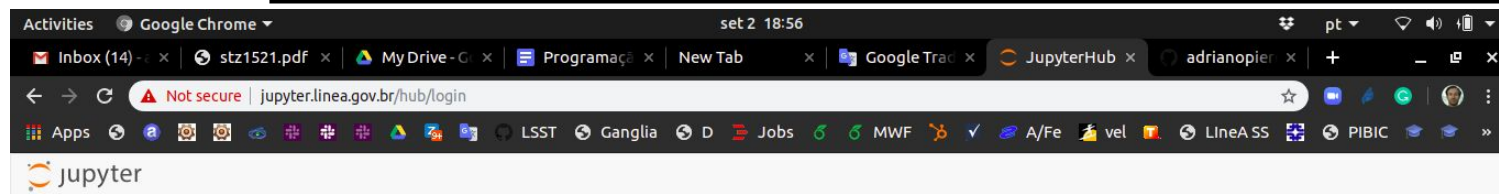
see <https://cdcv.s.fnal.gov/redmine/projects/des-y5/wiki>

and

[https://cdcv.s.fnal.gov/redmine/projects/des-y5/wiki/Y5A1MOF\\_HSC](https://cdcv.s.fnal.gov/redmine/projects/des-y5/wiki/Y5A1MOF_HSC)

Scientific topics:

- Stellar locus, SG classifier (based on the morphology);
- Stellar completeness;
- DES-DR1, HSC, stars in PanSTARRS;
- Download small samples of 2MASS data;

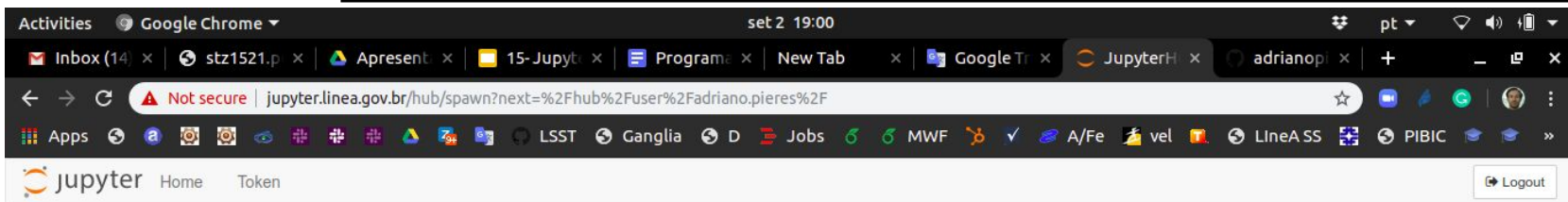


Sign in

Username:

Password:

Sign In



## Spawner Options

### Docker Image

science-0.7

### GitHub Repository to clone (OPTIONAL)

[https://github.com/adrianopierres/bootcamp\\_jn\\_qa.git](https://github.com/adrianopierres/bootcamp_jn_qa.git)

[link](#)

- Use Jupyterlab
- Use gitconfig from devel
- Use .ssh dir from devel

Spawn

---

# Parte II - Desenvolvimento de Python Software

Michel Aguená

O que é?

Bibliotecas que podem ser importadas em python  
(numpy, scipy, matplotlib, ...)

Como python importa:

```
# para python 2  
>> import mytest  
>> mytest.hw()
```

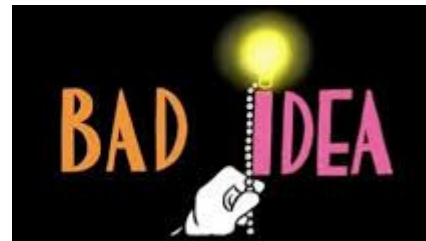
```
mytest.py  
:~::~:  
def hw():  
    print('Hello World!')
```

Exemplo:

<http://dev.linea.gov.br/~aguena/bootcamp/15-example/>

## Boas Práticas:

- Funções devem ser simples, executar 1 ação



```
def run():  
    function1()  
    function2()  
    function3()  
def function1():  
    <<rotina 1>>  
def function2():  
    <<rotina 2>>  
def function3():  
    <<rotina 3>>
```

```
def run():  
    <<rotina 1>>  
    <<rotina 2>>  
    <<rotina 3>>
```

## Boas Práticas:

- Documentação
  - Nomear claramente as funções (ações) e variáveis
  - Docstrings: documentar as informações principais de cada função - propósito, input, output, notas adicionais

```
def adds_1(x):  
    """  
    Adds 1 to input  
  
    Parameters  
    -----  
    x: int, float  
        Input value  
  
    Returns  
    -----  
    y: int, float  
        x+1  
  
    Notes  
    -----  
    """  
    y = x+1  
    return y
```

**mytest.py**

```
# pydoc mytest  
Help on function adds_1 in test:  
  
test.adds_1 = adds_1(x)  
    Adds 1 to input  
  
    Parameters  
    -----  
    x: int, float  
        Input value  
  
    Returns  
    -----  
    y: int, float  
        x+1  
  
    Notes  
    -----
```



Pacote:

```
example_package
|-- setup.py
`-- example_code
```

Instalação:

```
cd example_package
python setup.py install #--user
```

Estrutura:

```
example_code
|-- __init__.py
|-- folder1
|   |-- __init__.py
|   |-- file1.py
|   `-- file2.py
`-- main.py
```

Uso:

```
example_code.main
example_code.folder1
example_code.folder1.file1
```

## Hacking

### Baixar:

<http://dev.linea.gov.br/~aguena/bootcamp/15-example.tar.gz>

```
wget http://dev.linea.gov.br/~aguena/bootcamp/15-example.tar.gz  
tar xvf 15-example.tar.gz
```

### 1. Testar notebook hackeado

```
notebooks/hacked.ipynb
```

### 2. Instalar pacote e rodar notebook simples

```
notebooks/installed.ipynb
```



[my jupyter lab](#)