

h00_BJ

August 7, 2017

1 Tutorial de instalação e uso do Jupyter Notebook em GNU/Linux (Ubuntu)

2 Objetivos

As metas desse tutorial são ajudar o usuário a: - Instalar de pacotes de suporte ao Jupyter Notebook (Anaconda, nbconvert, pandoc, extensions) - Baixar, abrir, editar e criar Notebooks que executem blocos de código de Python (Python 3) - Fazer um primeiro uso da linguagem Python 3 dentro dos Notebooks - Converter .ipynb para PDF

3 Downloads necessários

O primeiro download a ser feito é do Anaconda, um conjunto de pacotes para o Python que traz consigo editores de texto e outros recursos. Usaremos isso como recurso de otimização de tempo e complexidade: basta baixar esse pacote para usar as bibliotecas mais comuns **e o próprio Jupyter**



Notebook vem incluso.

```
[//]: # (- Comando opcional antes de começar, para atualizar os pacotes do próprio Ubuntu:) [//]: # (bash) [//]: # (sudo apt-get update && sudo apt-get upgrade) [//]: # ()
```

Passo 01: Por via das dúvidas, confira a versão adequada. Abra um terminal pelo atalho **Ctrl+Alt+T**. No terminal, digite:

```
uname -m.
```

Normalmente retorna 'i686', 'x86' ou 'x86_64'. Os dois primeiros retornos indicam 32 bits, e o último indica um sistema de 64 bits.

Passo 02: Acesse a [página de download](#) - Clique na aba Download for Linux - Baixe a versão adequada do Python 3 a seu sistema operacional (32 ou 64 bits).

Passo 03: Quando o download for concluído, entre na pasta Downloads (ou a pasta que você baixou):

```
cd Downloads
```

Passo 04: No terminal, execute um dos comandos abaixo, para 64 ou 32 bits, respectivamente.

```
""bash bash Anaconda3-4.3.1-Linux-x86_64.sh
```

ou

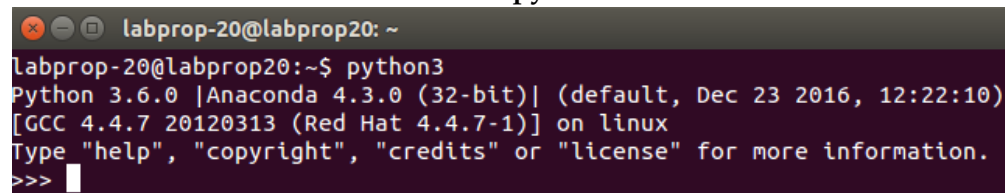
```
> ````bash
bash Anaconda3-4.3.1-Linux-x86_64.sh
```

Passo 05: A instalação começará, e o usuário deve seguir confirmando os requests com 'Enter' ou 'yes'.

Passo 06 (só é necessário se você respondeu 'no' para a última pergunta do passo anterior): Por fim, digite (substituindo o termo 'username' pelo seu usuário):

```
export PATH="/home/username/anaconda3/bin:$PATH"
```

Passo 07: Confira se o Anaconda está devidamente instalado abrindo o console de Python 3 no terminal. Basta usar o comando **python3**. O retorno deve ser como indicado na imagem:



```
labprop-20@labprop20: ~
labprop-20@labprop20:~$ python3
Python 3.6.0 |Anaconda 4.3.0 (32-bit)| (default, Dec 23 2016, 12:22:10)
[GCC 4.4.7 20120313 (Red Hat 4.4.7-1)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

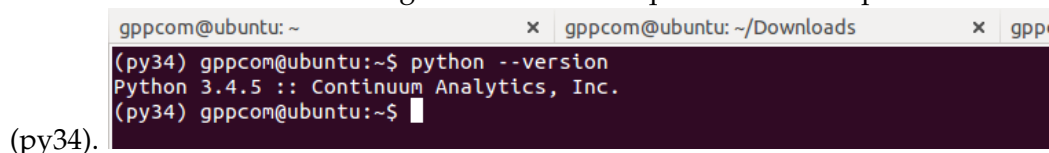
Passo 08: Após isso, em outro terminal, é só executar o jupyter notebook pelo comando: > bash jupyter notebook

Será aberta uma janela no navegador com os arquivos do seu computador. - **A instalação está concluída e o Jupyter pronto para uso!!!**

Passo 09 (muito importante): Como iremos utilizar o kernel do Matlab que utiliza a versão 3.4 do Python (é um requisito), criaremos um ambiente virtual para o Python 3.4. Para isso, digite em um terminal a seguinte sequência de comandos: - Para criar um ambiente chamado py34, digite:

```
conda create --name py34 python=3.4
```

- Para ativar o ambiente py34, digite: > bash source activate py34
- Você verá uma tela similar a figura abaixo. Note que o ambiente aparece identificado como



```
gppcom@ubuntu: ~
(py34) gppcom@ubuntu:~$ python --version
Python 3.4.5 :: Continuum Analytics, Inc.
(py34) gppcom@ubuntu:~$ █
```

- Para ativar o ambiente py34 sempre que chamar um terminal, digite:

```
bash echo 'source activate py34' >> ~/.bashrc
```

3.0.1 Pacotes adicionais

Passo 10: Instalação do Jupyter Notebook Extensions. Ele é um pacote de extensões que adiciona funcionalidade à aplicação. Instale através do comando: `> bash conda install -c conda-forge jupyter_contrib_nbextensions`

Para mais informações, acesse a página oficial: https://github.com/ipython-contrib/jupyter_contrib_nbextensions

Passo 11: Instalação do Jupyter Nbextensions configurator. Ele é um gerenciador para ativar e desativar extensões instaladas. Sua instalação não contém extensões embutidas, e é feita através do comando (via terminal): `> bash conda install -c conda-forge jupyter_nbextensions_configurator`

Passo 12: Para utilizar o gerenciador (e checar se tudo está indo bem com a instalação do ambiente), digite em um terminal:

```
""bash jupyter notebook
```

```
- Vá para a tela aberta no seu navegador (dependendo de sua máquina, demora a aparecer)
! [fig_nbextensions] (./FIGS/HD_00_BJ/f9.png)
```

```
- Uma tela com as extensões instaladas será exibida:
! [fig_nbextensions] (./FIGS/HD_00_BJ/f10.png)
```

```
- Para mais informações acesse: https://github.com/Jupyter-contrib/jupyter\_nbextensions
```

```
**Passo 13:** Instalação do Pandoc. Ele é uma ferramenta responsável pela conversão de arquivos de texto para PDF.
> ``bash
sudo apt-get install pandoc
```

Renderização de imagens: Problemas com exportação de arquivos (pdf e latex) e renderização de imagens são comuns: as imagens não aparecem no arquivo de saída, porque o pandoc tem uma sintaxe própria para adição de imagens. Use código `![titulodaimagem](caminho/da/imagem.extensao)` para que a imagem seja renderizada no notebook e no arquivo de saída. Em caso de dúvidas, acesse a página oficial: <http://pandoc.org/>. **O JUPYTER AINDA POSSUI PROBLEMAS COM EXPORTAÇÃO DE IMAGENS, QUE PODEREM SER CONTORNADOS COM UMA CONVERSÃO PARA LATEX SEGUIDA DE UMA COMPILAÇÃO E GERAÇÃO DE PDF. AINDA ASSIM, AS IMAGENS PODEREM PRECISAR DE AJUSTES DIMENSIONAIS.**

Passo 14: Alguns pacotes precisam ser instalados no seu novo ambiente. Para instalá-los, digite em um terminal a sequência de comandos a seguir:

- Atualizar os pacotes do próprio Ubuntu: `> bash sudo apt-get update && sudo apt-get upgrade`
- Instalar o **Spyder** (IDE para o Python), editor **vim**, editor de Latex **Kile**, comparador de arquivos **meld** e compressor/descompressor **RAR**: `> bash sudo apt-get install spyder vim kile meld rar`

- Instalar pacotes adicionais do Python: `> bash pip install numpy numpydoc pandas matplotlib graphviz scipy`

- Instalar mais alguns pacotes do Python: `> bash pip install babel bitarray bokeh bsdiff4 bz2file colorama decorator gitpython html5lib humanize imagesize ipyparallel iso8601 ipywidgets ipyparallel ipykernel jedi lxml markdown markdown2 meld3 memory_profiler natsort nbconvert numexpr openpyxl packaging plotly py pyopengl pyopengl-accelerate python-utils pyyaml pyzmq scikit-image scikit-learn seaborn sympy xlsxwriter xlrd`

- Instalar pacotes do Latex:

```
sudo apt-get install texlive-full okular
sudo add-apt-repository "deb http://archive.canonical.com/ precise partner"
sudo apt-get update
sudo apt-get install acroread
sudo add-apt-repository -r "deb http://archive.canonical.com/ precise partner"
sudo apt-get update
```

4 Baixar o Repositório do Curso

Passo 15: Para **Clonar** todo o repositório do curso, i.e., baixar todos os arquivos para uma máquina local, execute no terminal: `> bash git clone https://github.com/vicentesousa/DCO2004_LabPSC`

Em qualquer momento, **por exemplo, no começo de cada aula**, você pode fazer uma atualização dos arquivos do repositório e baixar atualizações. Para tal, execute no terminal:

```
git pull
```

5 Jupyter Notebook: o que é, vantagens e desvantagens



Antes conhecida como IPython Notebook, é uma ferramenta de código aberto (*open source*) utilizada para visualizar e interagir com informações e códigos. O pacote Jupyter permite que o usuário exponha textos e códigos em blocos de conteúdo específicos, gerando um arquivo que pode ser usado para visualização, apenas, ou execução de *scripts* em tempo real. Várias linguagens de programação

são suportadas pelo Jupyter: C/C++, Matlab, Python, R, Julia, entre outras. Sobre o arquivo gerado (*notebook*), pode-se dizer que é um apanhado de funções de HTML com elementos interativos em JavaScript.

- O usuário pode ver os notebooks através do [NBViewer](#), inserindo a URL do arquivo ou acessando o repositório do GitHub onde o notebook desejado se encontra;
- Para interagir (editar, executar células de código) com os notebooks, é necessário iniciar o pacote via terminal (para sistemas Linux) com o comando: `> bash jupyter notebook nome_do_notebook.ipynb`
- Para executar apenas o pacote, sem necessariamente abrir um notebook específico, basta usar: `> bash jupyter notebook`

5.0.2 Vantagens

A grande portabilidade e interatividade concedida aos artigos científicos, bem como projetos de programação, resume bem o que o Jupyter oferece aos seus usuários. Ao invés de artigos estáticos, cujos leitores são meramente passivos, os Notebooks possibilitam a execução das etapas propostas pelo autor e um acompanhamento dinâmico dos resultados. Recursos interessantes, por exemplo, são:

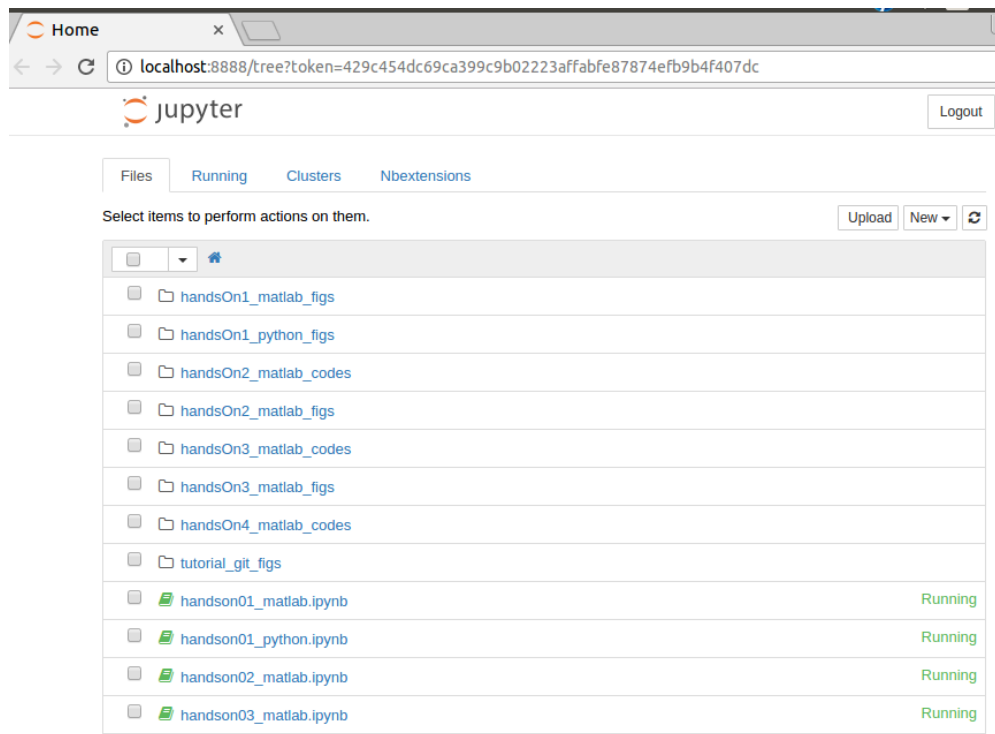
- Criação de gráficos estáticos e/ou interativos (com interface);
- Exportação dos notebooks em formatos diferentes: .pdf, .py, .html, etc;
- Várias linguagens operadas num mesmo notebook;
- Código aberto (*open source*).

5.0.3 Desvantagens

- O NBViewer não renderiza a todo instante os notebooks que você quiser acessar, o que pode retornar notebooks desatualizados e com problemas de visualização não corrigidos;
- Para que haja a interatividade oferecida, o usuário precisa ter instalado as dependências de Python, além de todas as linguagens utilizadas no notebook em questão (além dos kernels das linguagens). Isso traz um trabalho adicional no momento da instalação.

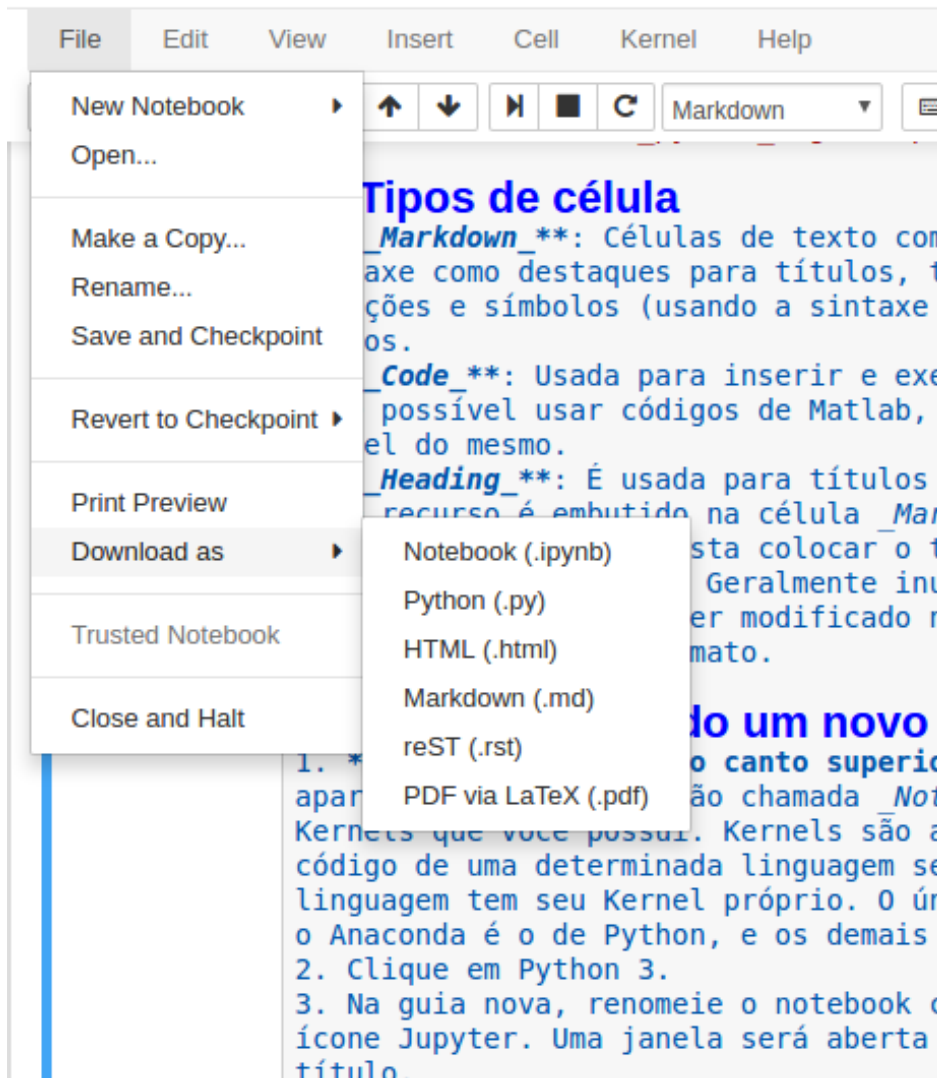
6 Interface no navegador

Quando o Jupyter é iniciado, como mostrado acima, é aberta uma nova guia em seu navegador padrão (neste caso, o Google Chrome) com a interface inicial exibindo os arquivos do diretório atual (dentro do qual você executou o comando ‘jupyter notebook’).



6.1 Tipos de célula

- **Markdown:** Células de texto comum. É possível usar recursos de sintaxe como destaques para títulos, texto em negrito e/ou itálico, equações e símbolos (usando a sintaxe do LaTeX), imagens, links e vídeos;
- **Code:** Usada para inserir e executar códigos. Vale salientar que só é possível usar códigos de Matlab, por exemplo, se você possuir o Kernel do mesmo e ativá-lo;
- **Heading:** É usada para títulos de seções e subseções. Contudo, esse recurso é embutido na célula *Markdown*, já que também é possível criar títulos nesta, basta colocar o texto entre *hashtags* (#).
- **Raw NBConvert :** Geralmente inutilizada, serve para inserir conteúdo que não deve ser modificado numa eventual exportação do notebook para outro formato, listados abaixo:



6.2 Iniciando um novo notebook

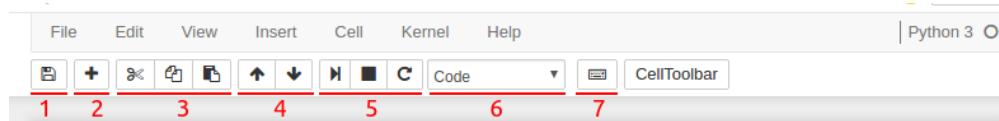
1. **Clique em “New”, no canto superior direito da tela.** Na aba que aparece, existe uma seção chamada *Notebooks*. Esta contém todos os Kernels que você possui. Kernels são as ferramentas que possibilitam um código de uma determinada linguagem ser executado no notebook, e cada linguagem tem seu Kernel próprio. O único Kernel que vem instalado com o Anaconda é o de Python, e os demais são instalados à parte.
2. Clique em Python 3.
3. Na guia nova, renomeie o notebook clicando em *Untitled*, ao lado do ícone Jupyter. Uma janela será aberta para que você insira o novo título.
4. Use Ctrl+s para salvar as alterações, ou clique no disquete (primeiro ícone da barra de ferramentas).

6.3 Editando o notebook

1. Clique no ícone ‘+’ para abrir uma nova célula.

2. **Selecione o tipo de célula que você quiser.** Por padrão, as células novas são do tipo *Code*, na qual se inserem códigos.
3. Execute a célula com um **Shift + Enter**.
4. Salve as alterações, ou edite as células novamente com dois cliques na mesma.

6.4 Barra de ferramentas



fig_figura4

1. Salvar (criar checkpoint).
2. Adicionar nova célula.
3. Cortar (usada para remover), copiar e colar célula, respectivamente.
4. Deslocar célula para cima ou para baixo.
5. Executar, parar execução ou reiniciar Kernel, respectivamente.
6. Selecionar tipo de célula.
7. Abrir paleta de comandos. Muitas funções/ferramentas podem ser executadas pelo teclado.

7 Prática 01

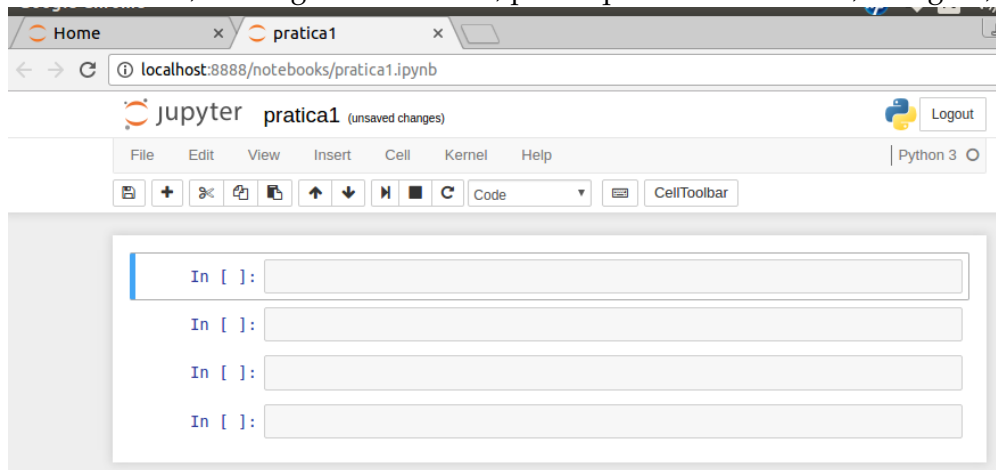
Para exercitar o uso de notebook e testar a instalação, siga os seguintes passos para elaborar um notebook simples.

Passo 1: Inicie o Jupyter notebook, digitand em um terminal: `> bash jupyter notebook`

Passo 2: Clique em *New* para criar um novo notebook.

Passo 3: Mude o título do notebook para *pratica1*.

Passo 4: Crie 4 células, sem digitar nada nelas, por enquanto. O resultado, até agora, deve ser



o seguinte:

Passo 5: Selecione o tipo da primeira célula para *Heading*, e coloque o seguinte título: Exercício da prática 1.

Passo 6: Selecione o tipo da segunda célula para *Markdown*, e esta conterá os seguintes itens: 1. " O Processo de modulação explora os três parâmetros da senoide portadora:" em negrito. 2. " Amplitude, Frequência e Fase" em itálico. 3. Um hiperlink para a seguinte página: https://pt.wikipedia.org/wiki/Modula%C3%A7%C3%A3o_em_amplitude 4. Uma imagem, de preferência a primeira que aparece no link acima. Digite:

![fig_figura] (<https://upload.wikimedia.org/wikipedia/commons/thumb/d/d5/Amplitude-modulation.png>)



Até agora, o resultado é:

Passo 7: Selecione o tipo da terceira célula para *Code*, e insira e execute o seguinte script de Python:

```
a = int(input("Escolha a primeira variável"))
b = int(input("Escolha a segunda variável"))
print("A soma das variáveis é:", a+b) ``
```

O resultado deve ser:

![fig_figura7] (./FIGS/HD_00_BJ/f7.png)

Passo 8: A última célula ser do tipo `_Raw NBConvert_`, e você pode inserir qualquer texto. O resultado final, após a execução das células, deve ser:

![fig_figura8] (./FIGS/HD_00_BJ/f8.png)

Passo 9: Como mostrado na seção 4, faça o download do notebook como `_.html_` (le

Passo 10: Abra o arquivo que você baixou em seu navegador e veja o resultado de OBS: Os dois últimos passos mostram uma opção para visualizar seu trabalho, mas est

Conversão para outros formatos

É possível exportar para PDF, LaTeX, HTML e outros formatos utilizando a opção `_Fil`

`jupyter nbconvert -to FORMATO nome_do_notebook.ipynb` - Para usar essa funcionalidade, é recomendado que a documentação do `nbconvert` seja lida: <https://media.readthedocs.org/pdf/nbconvert/latest/nbconvert.pdf>. O capítulo 2 (página 9) diz respeito aos formatos de saída

8 Prática 02

Para exercitar a exportação dos notebooks e um uso inicial de arquivos em Latex no Linux, siga os seguintes passos para exportar este notebook para PDF.

Passo 01: entre na pasta do repositório local que você baixou. Em um terminal digite: `> bash`
`cd ~/DCO2004_LabPSC`

Passo 02: Abra esse notebook em modo de edição. Em um terminal digite: `> bash jupyter notebook h00_BJ.ipynb`

```
(py34) gppcom@ubuntu:~$ cd DCO2004_LabPSC/
(py34) gppcom@ubuntu:~/DCO2004_LabPSC$ jupyter notebook h00_BJ.ipynb
[W 11:17:53.740 NotebookApp] Widgets are unavailable. Please install widgetsnbextension or ipywidgets 4.0
[I 11:17:53.768 NotebookApp] The port 8888 is already in use, trying another port.
[I 11:17:53.858 NotebookApp] Loaded server extension jupyter_nbextensions_configurator
[I 11:17:53.868 NotebookApp] Serving notebooks from local directory: /home/gppcom/DCO2004_LabPSC
[I 11:17:53.868 NotebookApp] 0 active kernels
[I 11:17:53.868 NotebookApp] The Jupyter Notebook is running at: http://localhost:8889/?token=9a5041636560b140764ef0631ab2d8bfa5660caa6bb
[I 11:17:53.868 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).

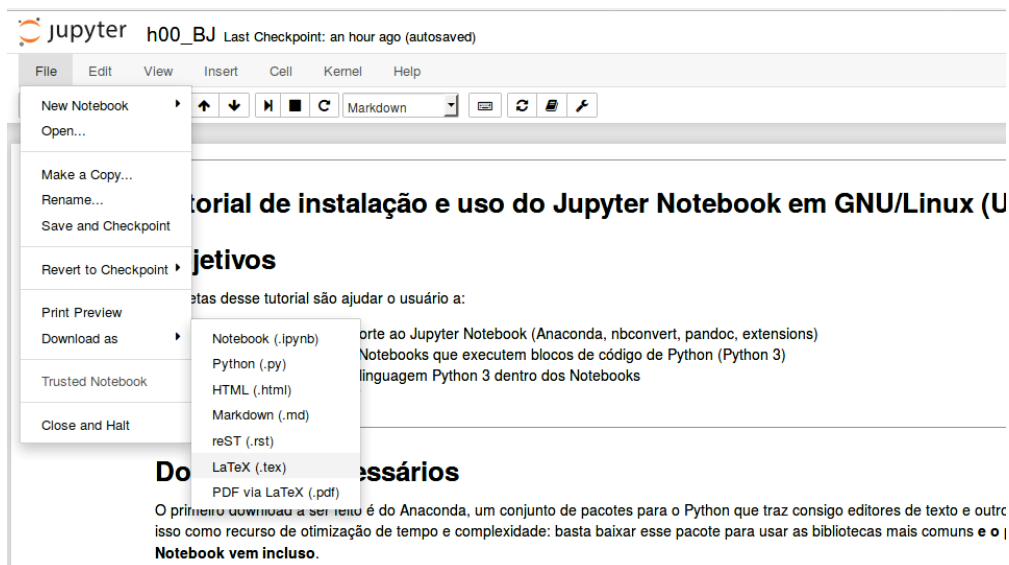
Copy/paste this URL into your browser when you connect for the first time,
to login with a token:
http://localhost:8889/?token=9a5041636560b140764ef0631ab2d8bfa5660caa6bb179c6
[I 11:17:53.570 NotebookApp] Accepting one-time-token-authenticated connection from 127.0.0.1
[I 11:17:58.581 NotebookApp] 302 GET /notebooks/HD_00_BJ/f2.png (127.0.0.1) 3.06ms
[I 11:17:58.586 NotebookApp] 302 GET /notebooks/HD_00_BJ/f3.png (127.0.0.1) 2.59ms
[I 11:17:58.591 NotebookApp] 302 GET /notebooks/HD_00_BJ/f4.png (127.0.0.1) 2.66ms
[I 11:17:59.192 NotebookApp] Kernel started: 2a97418d-bdfb-4488-9f38-6d5c26b56887
[I 11:17:59.924 NotebookApp] 302 GET /notebooks/HD_00_BJ/f0.png (127.0.0.1) 4.67ms
[I 11:17:59.944 NotebookApp] 302 GET /notebooks/HD_00_BJ/console.png (127.0.0.1) 2.58ms
[I 11:17:59.954 NotebookApp] 302 GET /notebooks/HD_00_BJ/ambienteipy34.png (127.0.0.1) 7.51ms
[I 11:17:59.991 NotebookApp] 302 GET /notebooks/HD_00_BJ/f9.png (127.0.0.1) 1.88ms
[I 11:17:59.993 NotebookApp] 302 GET /notebooks/HD_00_BJ/f10.png (127.0.0.1) 1.70ms
[I 11:18:00.055 NotebookApp] 302 GET /notebooks/HD_00_BJ/f1.png (127.0.0.1) 2.64ms
[I 11:18:00.096 NotebookApp] 302 GET /notebooks/HD_00_BJ/f5.png (127.0.0.1) 2.63ms
[I 11:18:00.101 NotebookApp] 302 GET /notebooks/HD_00_BJ/f6.png (127.0.0.1) 2.66ms
[I 11:18:00.144 NotebookApp] 302 GET /notebooks/HD_00_BJ/f7.png (127.0.0.1) 24.40ms
[I 11:18:00.153 NotebookApp] 302 GET /notebooks/HD_00_BJ/f8.png (127.0.0.1) 5.61ms
[W 11:18:00.650 NotebookApp] 404 GET /nbextensions/widgets/notebook/js/extension.js?v=20170720111752 (127.0.0.1) 58.95ms referer=http://localhost:8889/notebooks/h00_BJ.ipynb?token=c998406a20f29b384e60d5f666b736f869c53bffa8d7bf5
```

fig_prat02_fig01

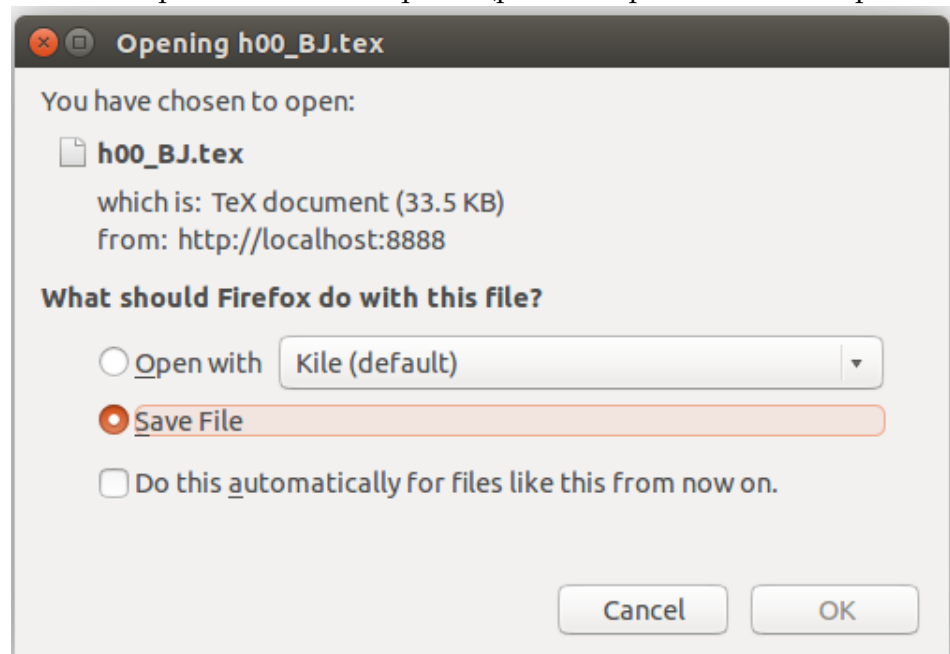


- O seu browser irá aparecer com a seguinte tela:

Passo 03: Exporte o notebook para um arquivo Latex (com extensão *tex*).

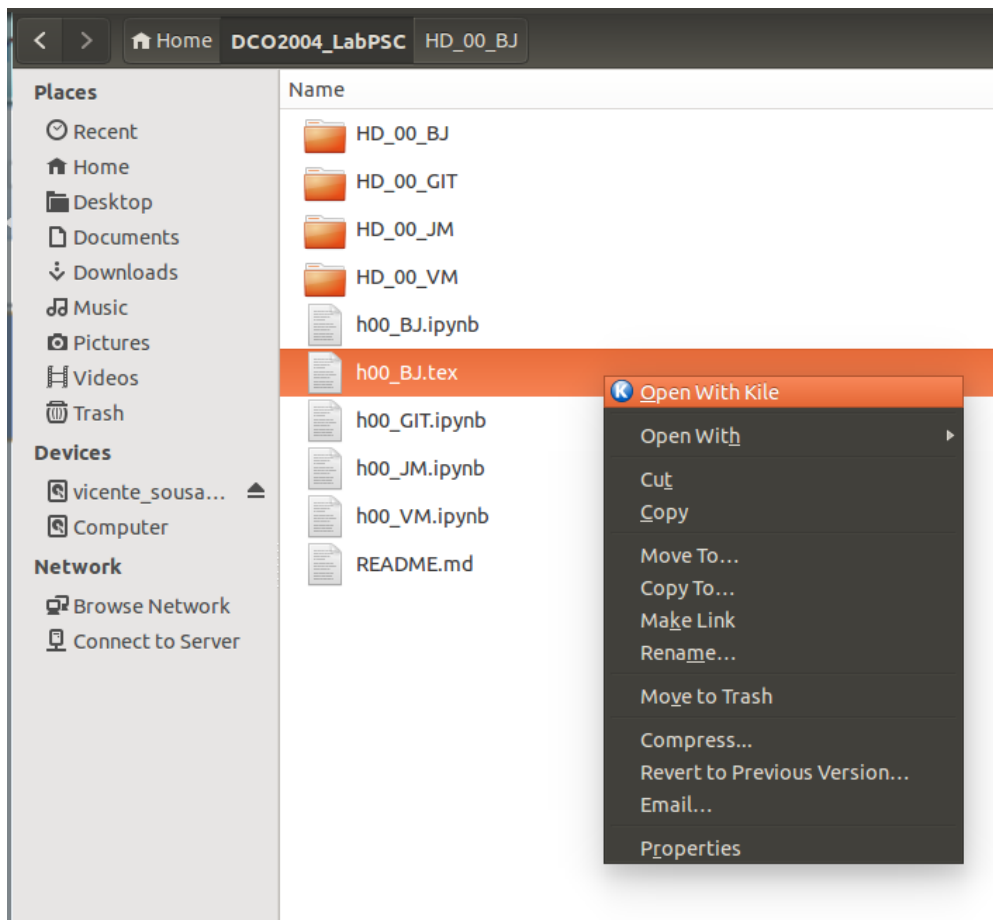


- Escolha um local conhecido para salvar o arquivo (por exemplo, dentro da pasta

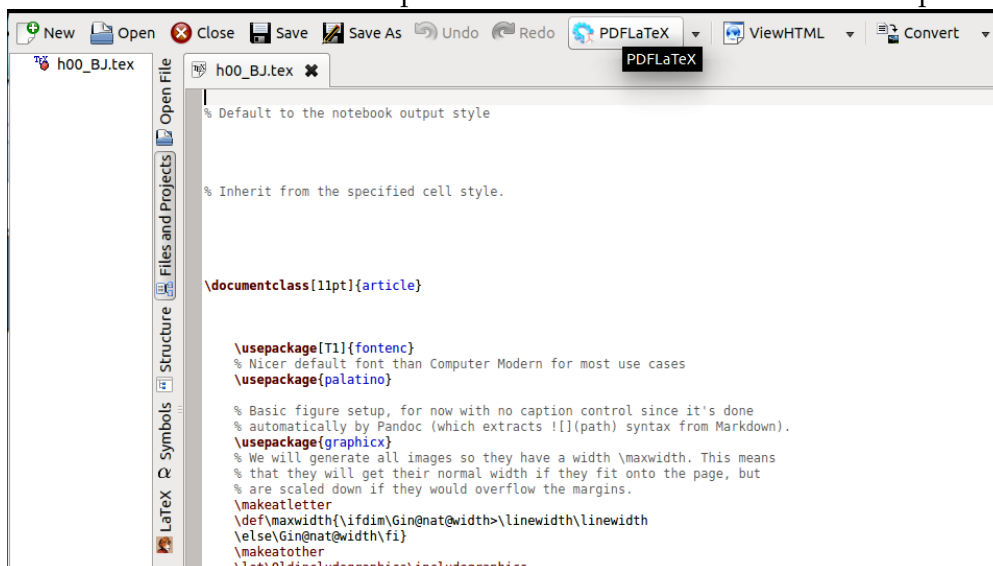


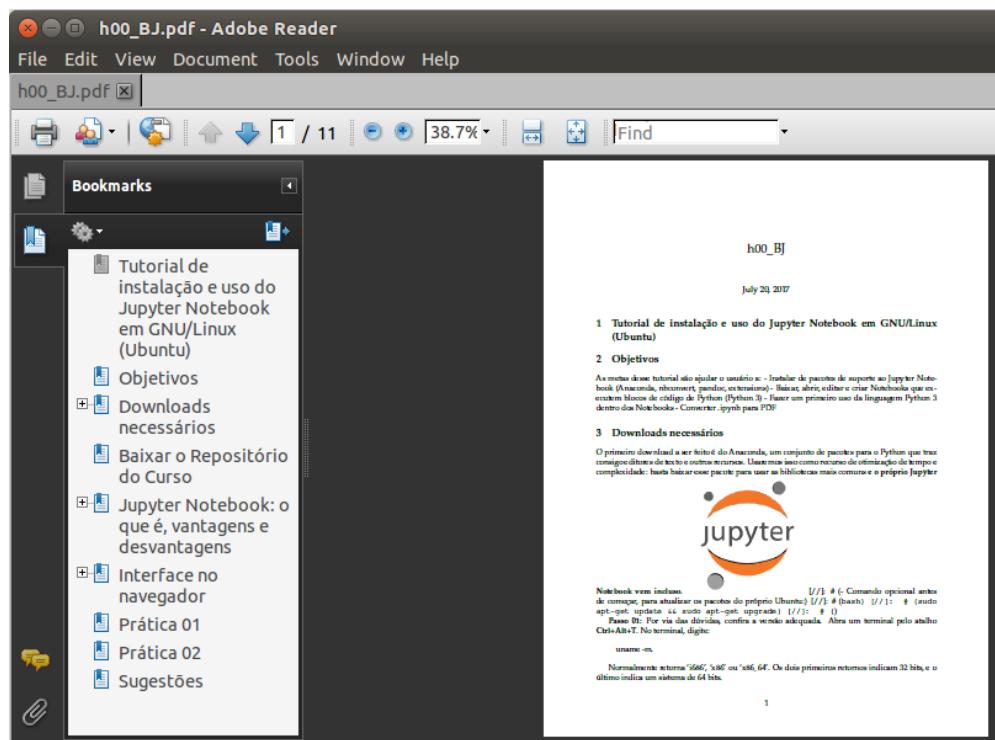
DCO2004_LabPSC).

Passo 04: Vá na pasta e abra o arquivo .tex com o Kile.



Passo 05: Compile o documento no menu *PDFLaTeX*. Seu documento será aberto pelo visualizador de PDF padrão do sistema.





9 Sugestões

- Baixe notebooks (inclusive este) para explorar a sintaxe com a qual eles foram editadas: isso ajuda a aprender recursos novos.
- Clique [aqui](#) para ver uma galeria com vários notebooks .
- Clique [aqui](#) para conferir 28 dicas, truques e atalhos do Jupyter Notebook.