The University of Queensland
School of Earth and Environmental Sciences

Introduction to geophysical data processing and modeling using Python

Lutz Gross
E-mail : l.gross@uq.edu.au

Zhengguang Zhao

E-mail : zhengguang.zhao@uq.net.au

Aims of the Course

- Hands-on introduce to basic concepts of python as a tool for computational sciences
- Show concepts for geophysical data processing
- Introduction to numerical modeling & inversion in python in Geophysics

Course Outline

- Day 1: Basic Concepts of Python
- Day 2: Arrays and data visualization:
  - matplotlib & numpy
- Day 3: Numerical modeling for Geophysics
  - esys.escript
- Day 4: Seismic Data Processing
- Day 5: Advanced Topics: Inversion

Program (somehow)

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00-10:30	`Prelude`	`Files`	`Numerical Modeling`	`Modeling Waves`	`Advanced topics`
10:30-11:00	break	break	break	break	
11:00-12:30	`Basics`	`Data Visualization`	`Geophysical modeling`	`Seismic data processing I`	
12:30-14:00	lunch	lunch		lunch	lunch
14:00-15:30	`Programming`	`Arrays`	`Geophysical modeling`	`Seismic data processing II`	exercises
15:30-16:00	break	break	break	break	
16:00-18:00	exercises	exercises		exercises	

Course Presentation

- Through Jupyter notebook

  → Hands-on work during lecture

- Course through a Virtual Machine Guest:
  - Debian Linux
  - With Anaconda3 python distribution:

    https://www.anaconda.com/
  - With esys-script package

    https://anaconda.org/conda-forge/esys-escript

Virtual Box

- Install VM host from

  https://www.virtualbox.org/wiki/Downloads

- Import OVF file for course

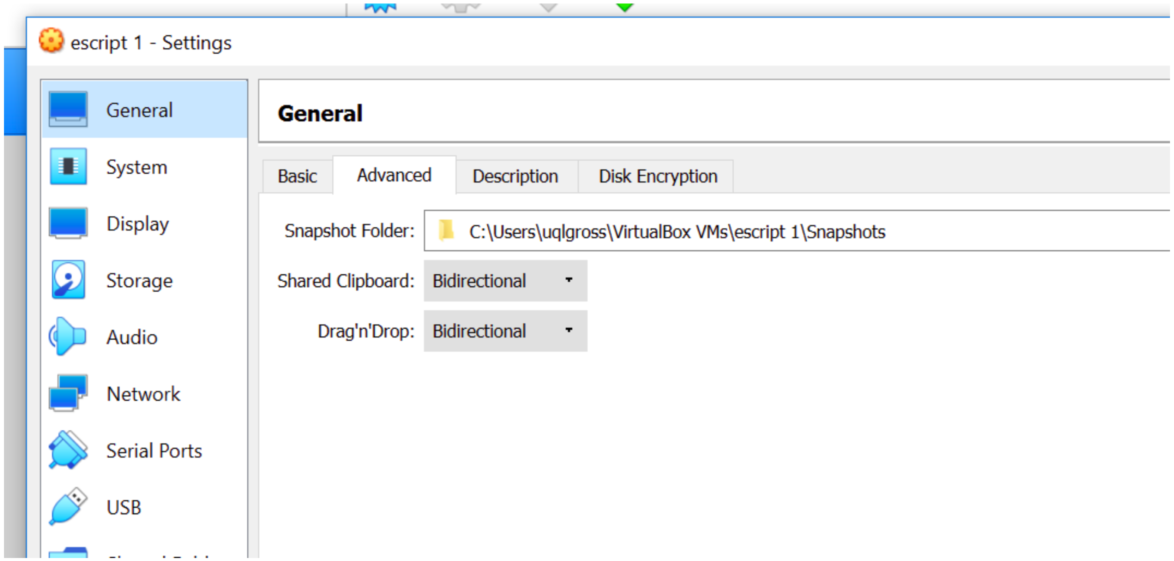
  click on OVF file

  Or

  select file through File->Import

Check Guest Settings

- Copy & past between Host & Guest:



escript 1 - Settings

General

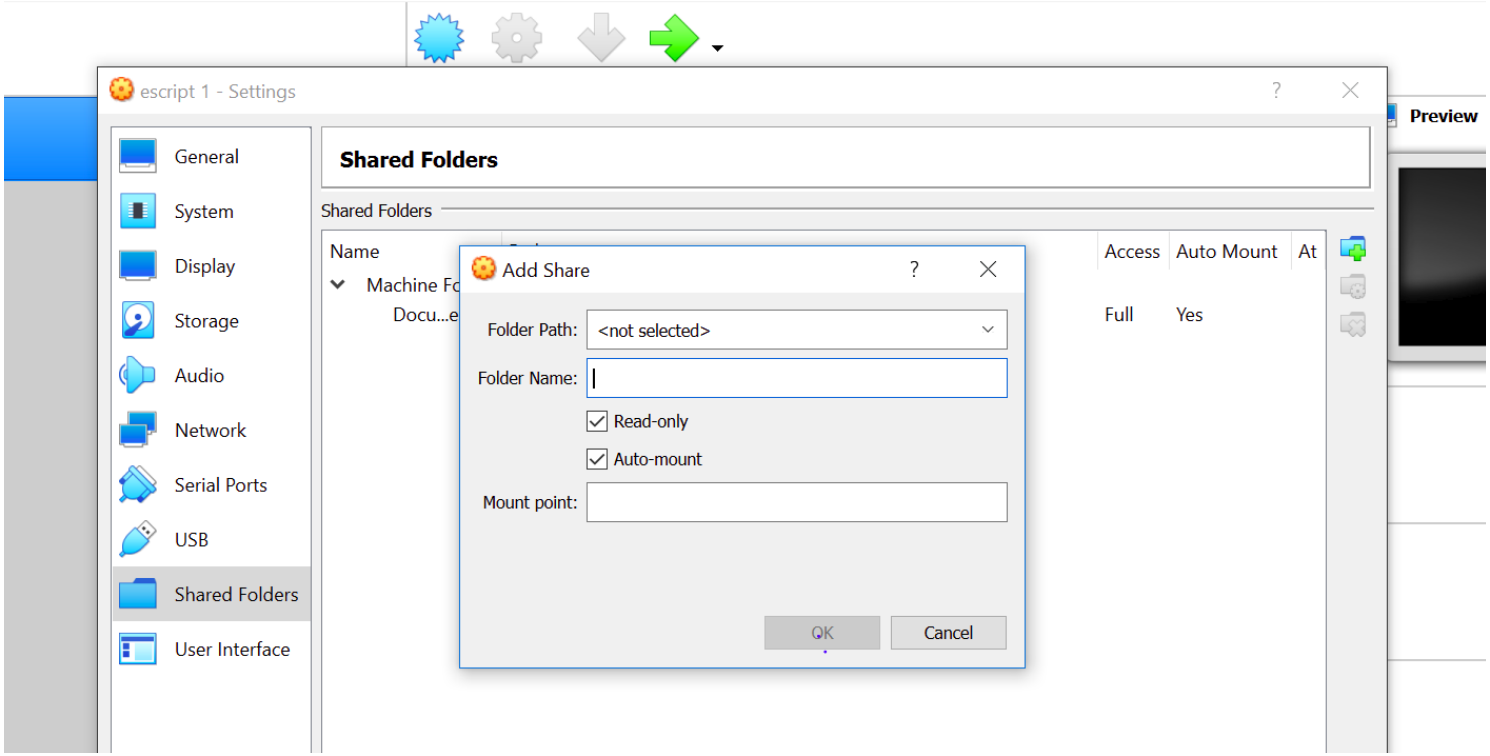
Snapshot Folder: C:\Users\uqlgross\VirtualBox VMs\escript 1\Snapshots

Shared Clipboard: Bidirectional

Drag'n'Drop: Bidirectional

Access Host Folder in Guest

- To transfer files between guest and host
- Settings → Shared Folders



escript 1 - Settings

Shared Folders

Shared Folders

Name	Access	Auto Mount	At
Machine Fo...			
Docu...e	Full	Yes	

Add Share

Folder Path: <not selected>

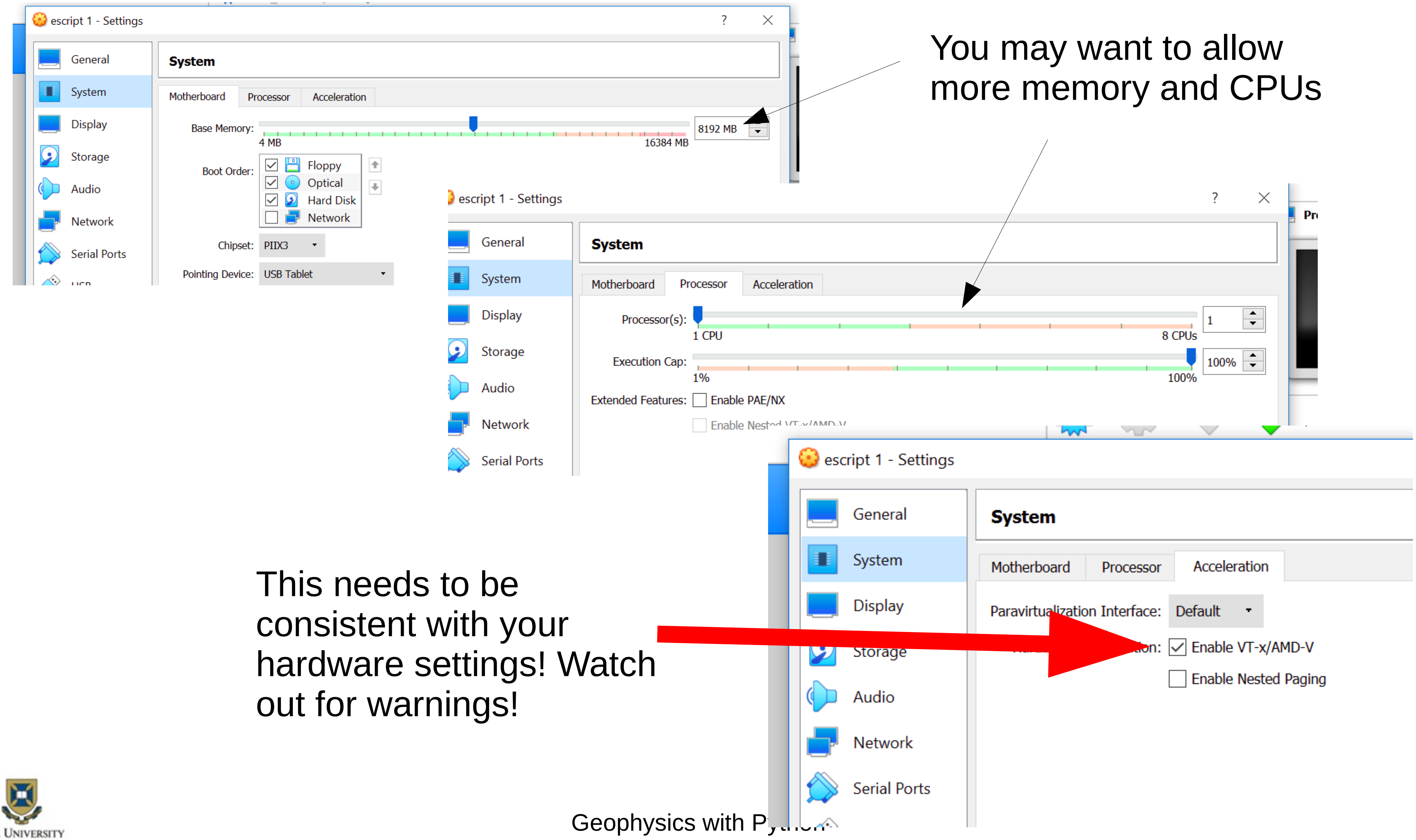
Folder Name:

☑ Read-only

☑ Auto-mount

Mount point:

Guest System Setting



escript 1 - Settings

General, System, Display, Storage, Audio, Network, Serial Ports

System

Motherboard | Processor | Acceleration

Base Memory: 4 MB – 16384 MB: 8192 MB

Boot Order:
- ☑ Floppy
- ☑ Optical
- ☑ Hard Disk
- □ Network

Chipset: PIIX3

Pointing Device: USB Tablet

escript 1 - Settings

System

Motherboard | Processor | Acceleration

Processor(s): 1 CPU – 8 CPUs: 1

Execution Cap: 1% – 100%: 100%

Extended Features: □ Enable PAE/NX

□ Enable Nested VT-x/AMD-V

escript 1 - Settings

System

Motherboard | Processor | Acceleration

Paravirtualization Interface: Default

Hardware Virtualization: ☑ Enable VT-x/AMD-V

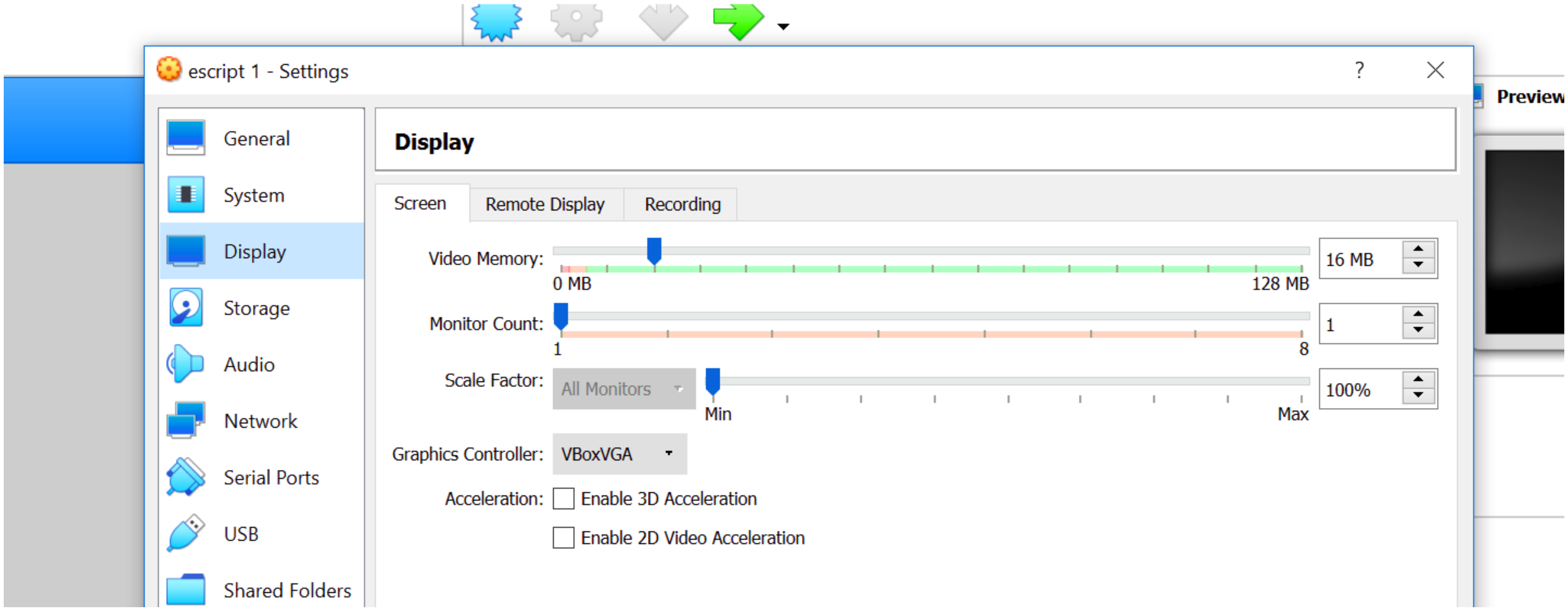
□ Enable Nested Paging

You may want to allow more memory and CPUs

This needs to be consistent with your hardware settings! Watch out for warnings!

Guest Display Settings

- You may want to allow more Video Memory
- “Vb



Display

Video Memory: 16 MB

Monitor Count: 1

Scale Factor: All Monitors 100%

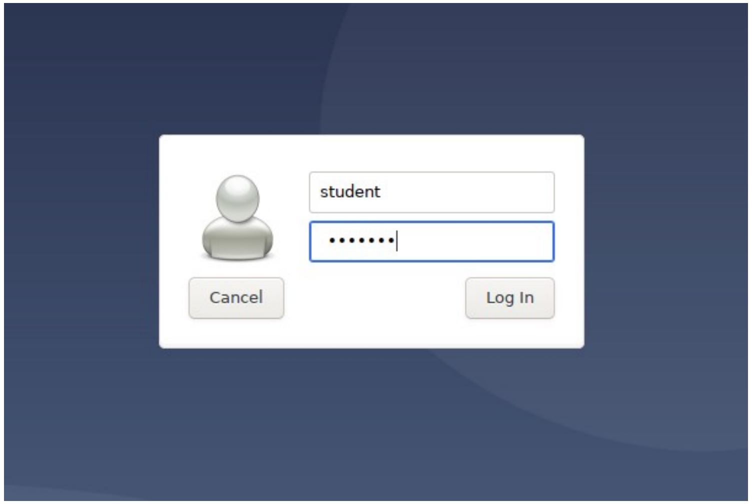
Graphics Controller: VBoxVGA

Acceleration: □ Enable 3D Acceleration

□ Enable 2D Video Acceleration

Fire up the guest machine

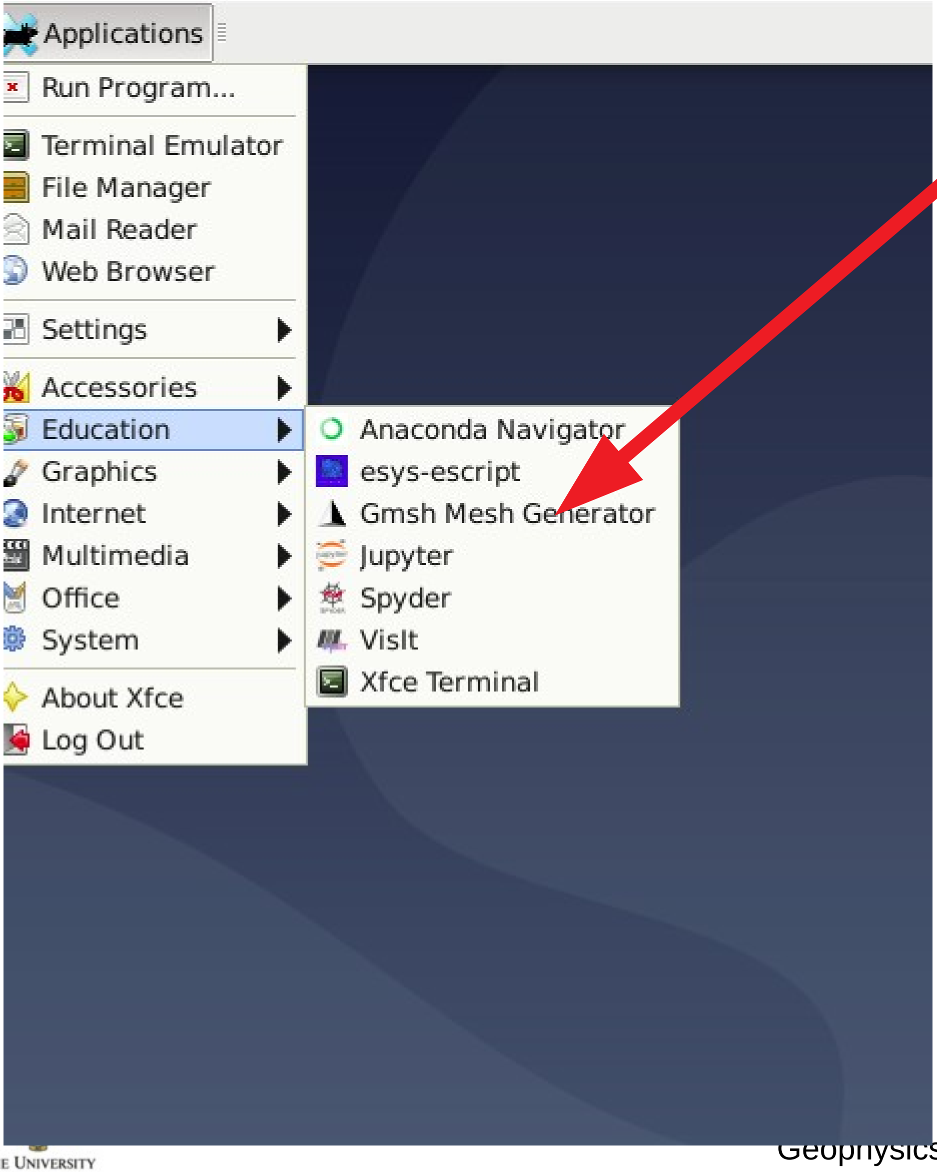
- Machine → Start → Normal Start
  - Or double click



student

Login:
Name: *student*
Password: *escript*

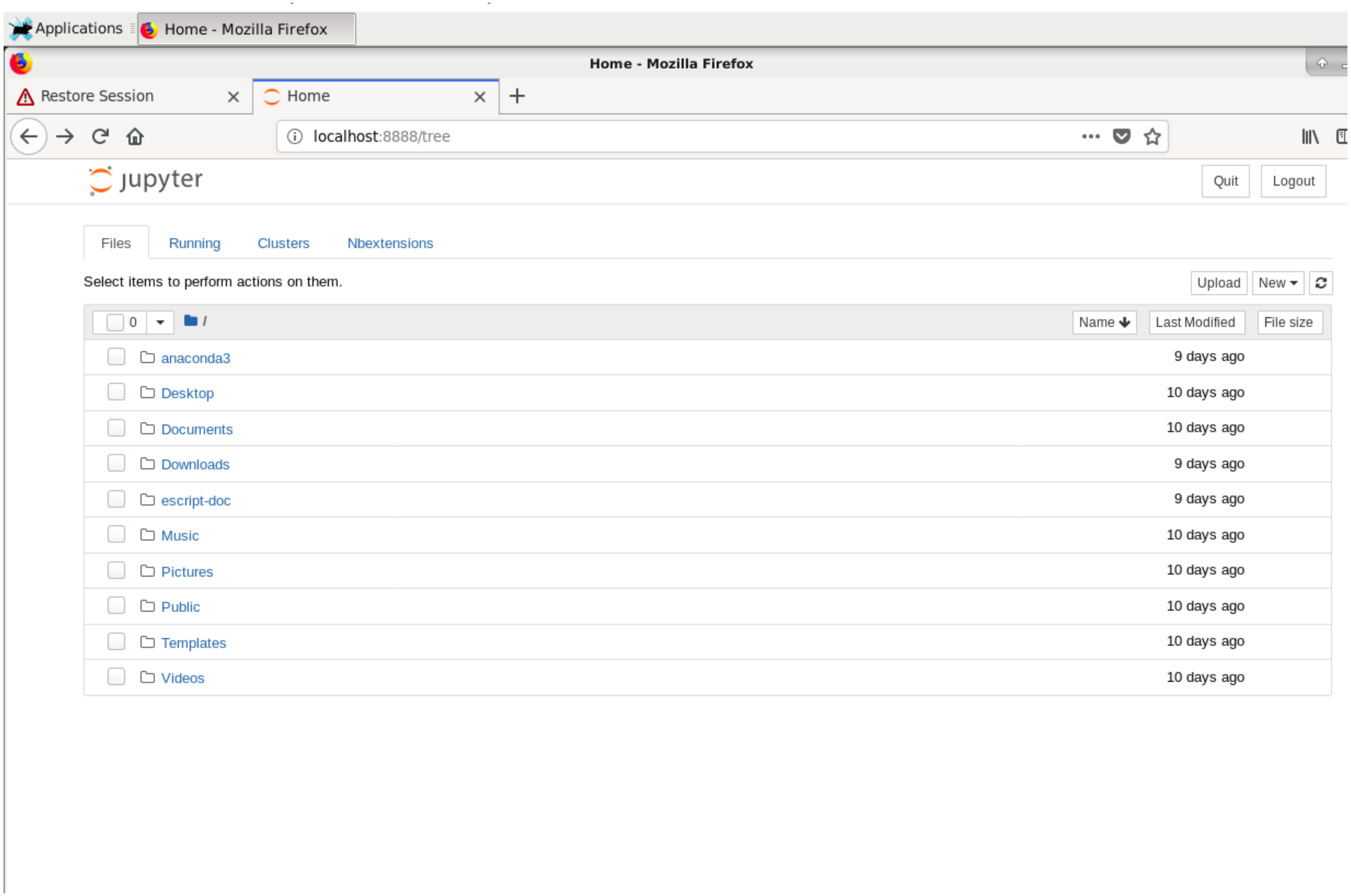
Start Jupyter Server



- **Anaconda**: python development environment
- **Esys-escript** interactive shell
- **Gmsh** 3D FEM mesh generator
- **Jypyter** notebook server
- **Spyder** python code editor
- **VisIt** 3D visualization
- New **Terminal**

Jupyter Portal

- In the Firefox browser



Select items to perform actions on them.

Name	Last Modified	File size
anaconda3	9 days ago	
Desktop	10 days ago	
Documents	10 days ago	
Downloads	9 days ago	
escript-doc	9 days ago	
Music	10 days ago	
Pictures	10 days ago	
Public	10 days ago	
Templates	10 days ago	
Videos	10 days ago	

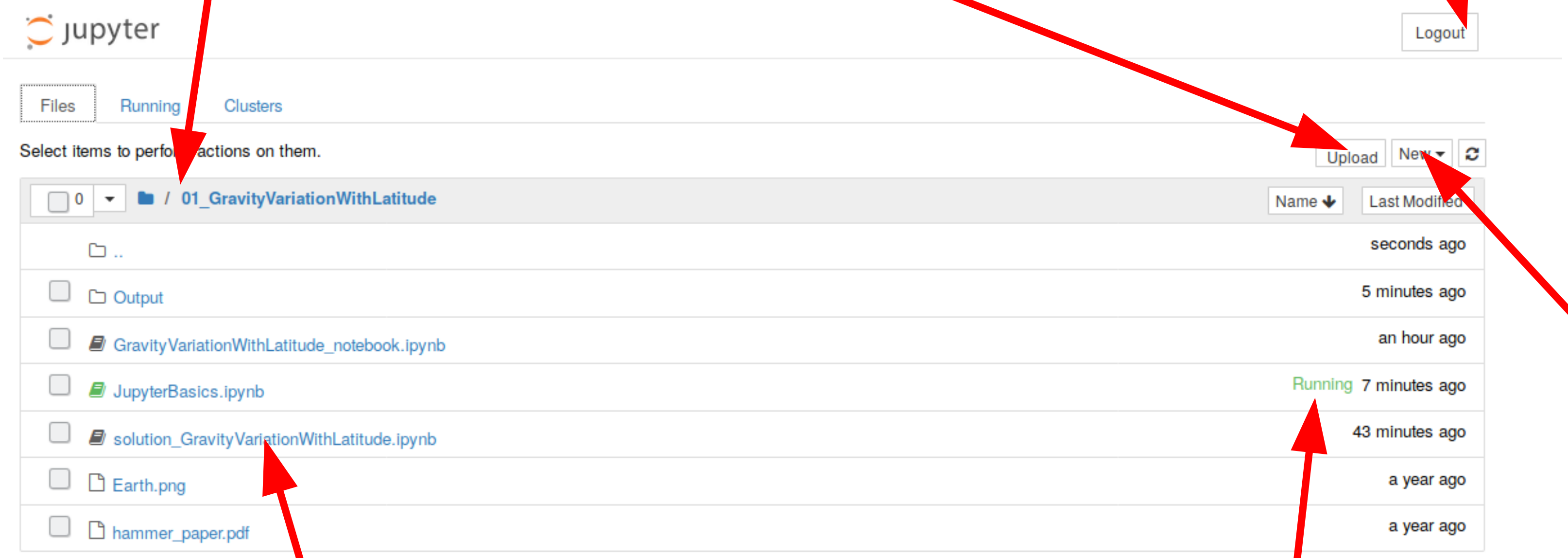
Jupyter notebook

- web application → use your favorite browser
  - contain live python code
  - includes narrative text
  - contains equations
  - includes visualizations
  - to create and share documents
- See https://jupyter.org/

A jupyter notebook

- is two things:
  - A document containing text and python code
    - Extension: ipynb
  - A session
    - to render the text
    - to run the code/sections of the code

After login



Select items to perform actions on them.

☐ 0 / 01_GravityVariationWithLatitude

Name	Last Modified
..	seconds ago
☐ Output	5 minutes ago
☐ GravityVariationWithLatitude_notebook.ipynb	an hour ago
☐ JupyterBasics.ipynb	Running 7 minutes ago
☐ solution_GravityVariationWithLatitude.ipynb	43 minutes ago
☐ Earth.png	a year ago
☐ hammer_paper.pdf	a year ago

Your files and directories

Upload file

Leave Jupyter

Create new notebook, directory

Click to start Notebook in new tab

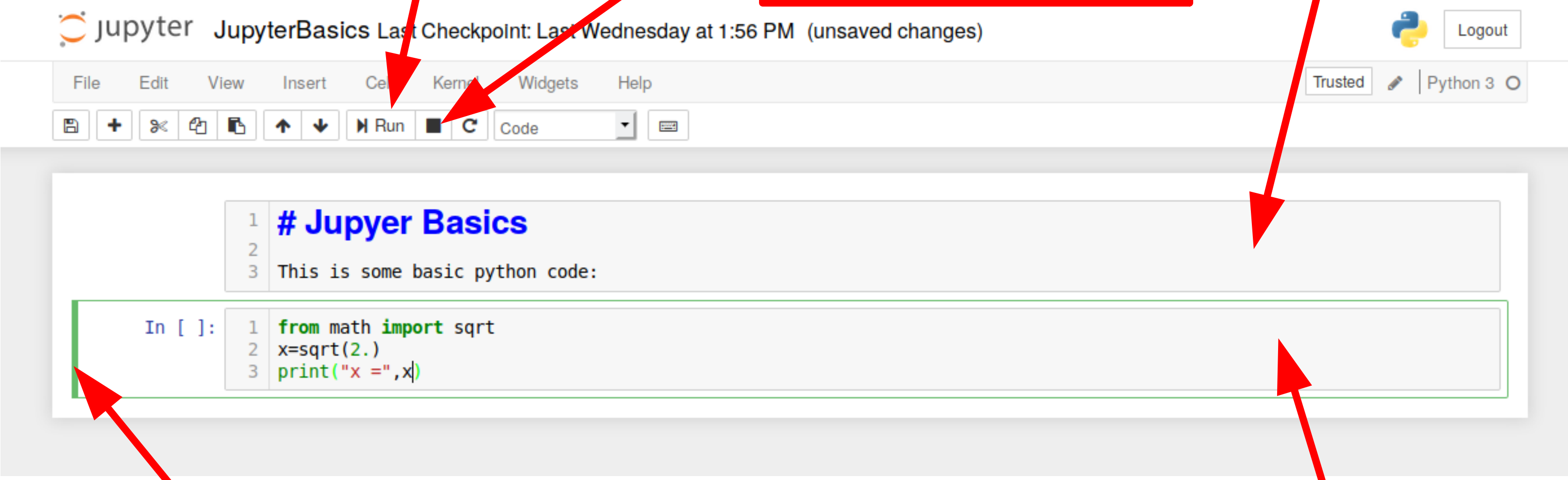
Notebook running in other browser tab

notebook page

Run python code
+ render cells
→ active cell only

Cell with
text

Stop Running
python code



JupyterBasics Last Checkpoint: Last Wednesday at 1:56 PM (unsaved changes)

```
# Jupyer Basics

This is some basic python code:
```

In [ ]:

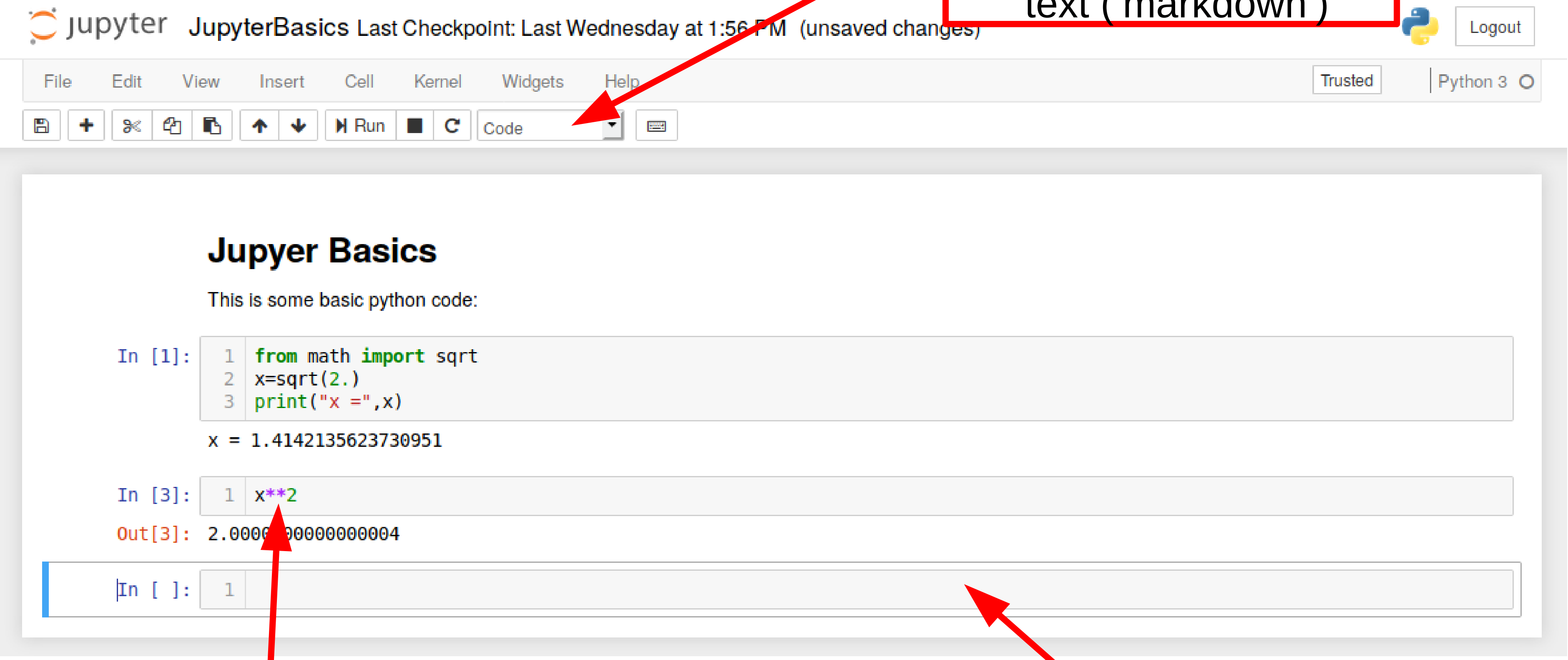
```
from math import sqrt
x=sqrt(2.)
print("x =",x)
```

active cell with code

Cell with
python code

Cells after ‘run’

Change cell type Code or text (‘markdown’)


jupyter
JupyterBasics Last Checkpoint: Last Wednesday at 1:56 PM (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code

Jupyer Basics

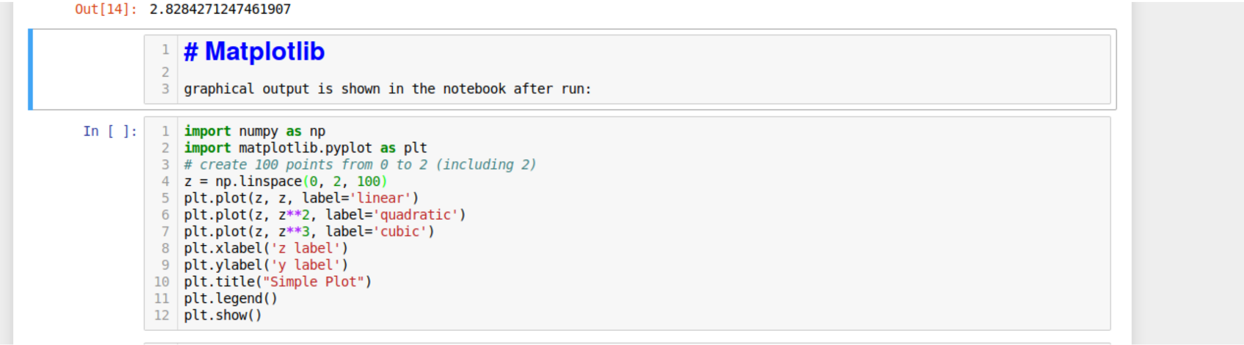
This is some basic python code:

```
In [1]: from math import sqrt
        x=sqrt(2.)
        print("x =",x)

        x = 1.4142135623730951

In [3]: x**2

Out[3]: 2.0000000000000004

In [ ]:
```

All cells treated as single python code

Cell code cell added

Matplotlib output is integrated

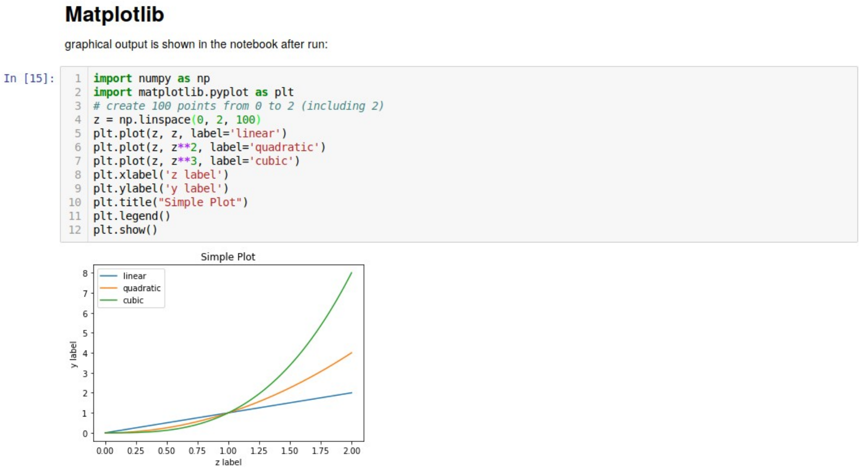


```
Out[14]:  2.8284271247461907
```

Matplotlib

```
graphical output is shown in the notebook after run:
```

```
In [ ]:
import numpy as np
import matplotlib.pyplot as plt
# create 100 points from 0 to 2 (including 2)
z = np.linspace(0, 2, 100)
plt.plot(z, z, label='linear')
plt.plot(z, z**2, label='quadratic')
plt.plot(z, z**3, label='cubic')
plt.xlabel('z label')
plt.ylabel('y label')
plt.title("Simple Plot")
plt.legend()
plt.show()
```

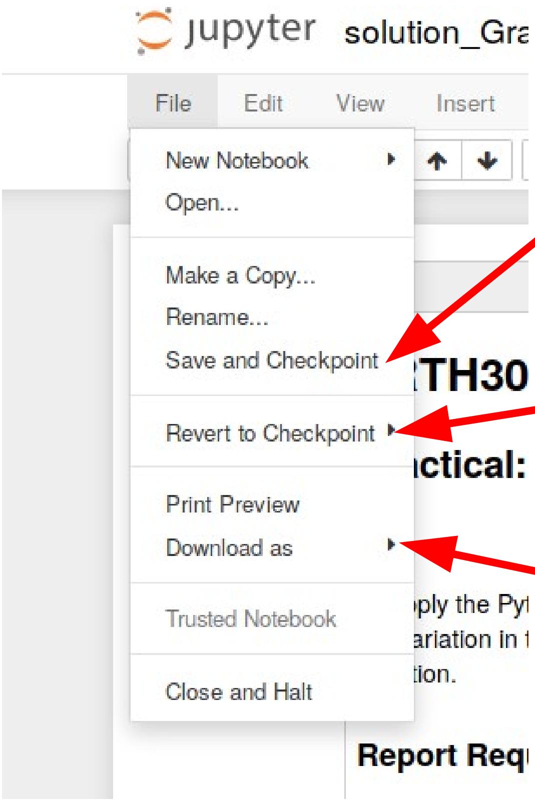


Matplotlib

graphical output is shown in the notebook after run:

```
In [15]:
import numpy as np
import matplotlib.pyplot as plt
# create 100 points from 0 to 2 (including 2)
z = np.linspace(0, 2, 100)
plt.plot(z, z, label='linear')
plt.plot(z, z**2, label='quadratic')
plt.plot(z, z**3, label='cubic')
plt.xlabel('z label')
plt.ylabel('y label')
plt.title("Simple Plot")
plt.legend()
plt.show()
```

File Menu

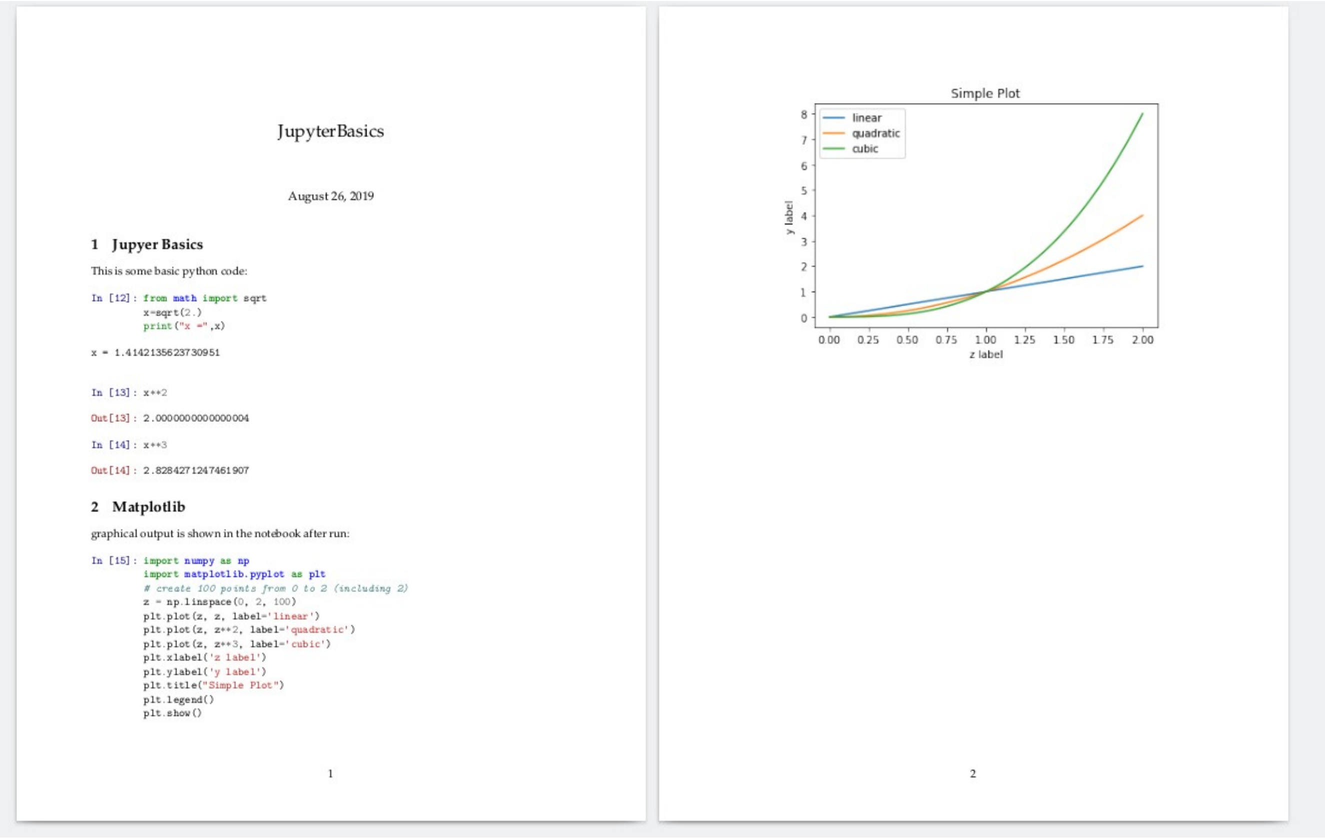


Save your work

Return to last version

Create PDF file

Example report



JupyterBasics

August 26, 2019

1 Jupyer Basics

This is some basic python code:

```
In [12]: from math import sqrt
         x=sqrt(2.)
         print("x =",x)

x = 1.4142135623730951


In [13]: x**2

Out[13]: 2.0000000000000004

In [14]: x**3

Out[14]: 2.8284271247461907
```

2 Matplotlib

graphical output is shown in the notebook after run:

```
In [15]: import numpy as np
         import matplotlib.pyplot as plt
         # create 100 points from 0 to 2 (including 2)
         z = np.linspace(0, 2, 100)
         plt.plot(z, z, label='linear')
         plt.plot(z, z**2, label='quadratic')
         plt.plot(z, z**3, label='cubic')
         plt.xlabel('z label')
         plt.ylabel('y label')
         plt.title("Simple Plot")
         plt.legend()
         plt.show()
```

1

2

And more

- ‘Edit’ → copy & past & split & merge
- ‘Insert’ → add new cell
- ‘Run’ → run all cells, run cells above/below, …
- For markdown text see for instance:
  - https://www.datacamp.com/community/tutorials/markdown-in-jupyter-notebook