

Python Packages

Jupyter, pandas, and matplotlib

Python

- Interpreted
- Common developing environments
 - What is an IDE?
 - What IDE did you use before?
- Installing packages
 - pip
 - <https://pypi.org/>

Jupyter Notebooks

- Integrate markdown, code, and visualizations into one document
- Must live on top of a python kernel to execute cells
- A great format for presentation of results with accountability of how the results were created.
- Not typically used for autonomous processes, meant to be consumed by a human

Python Virtual Environments

- We need everyone in the class to be using the *exact same* python environment for grading consistency
- We'll use a cloud product called *Google Colab* that can manage your python distribution and your installed packages
- This enables group work, grading environment consistency, and gives you “real world” practice for environment hygiene

Create a notebook

- <https://colab.research.google.com/notebooks/intro.ipynb>

The screenshot displays the Google Colaboratory (Colab) web interface. At the top, the browser address bar shows the URL `colab.research.google.com/notebooks/intro.ipynb?utm_source=scs-index#`. Below the browser bar, the Colab logo and the text "Welcome To Colaboratory" are visible, along with a menu bar containing "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help". On the right side of the header, there are links for "Share", a settings gear, and a user profile icon.

On the left side, a "Table of contents" sidebar is open, listing the following sections: "Getting started", "Data science", "Machine learning", "More Resources", "Machine Learning Examples", and a "Section" button at the bottom.

The main content area is titled "What is Colaboratory?" and features the Colab logo. The text explains that Colaboratory, or "Colab" for short, allows users to write and execute Python in their browser. It lists three key features:

- Zero configuration required
- Free access to GPUs
- Easy sharing

Below this, a paragraph states: "Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) to learn more, or just get started below!"

A section titled "Getting started" follows. It explains that the document is an interactive environment called a "Colab notebook" for writing and executing code. It then provides an example of a "code cell" containing a short Python script that calculates the number of seconds in a day:

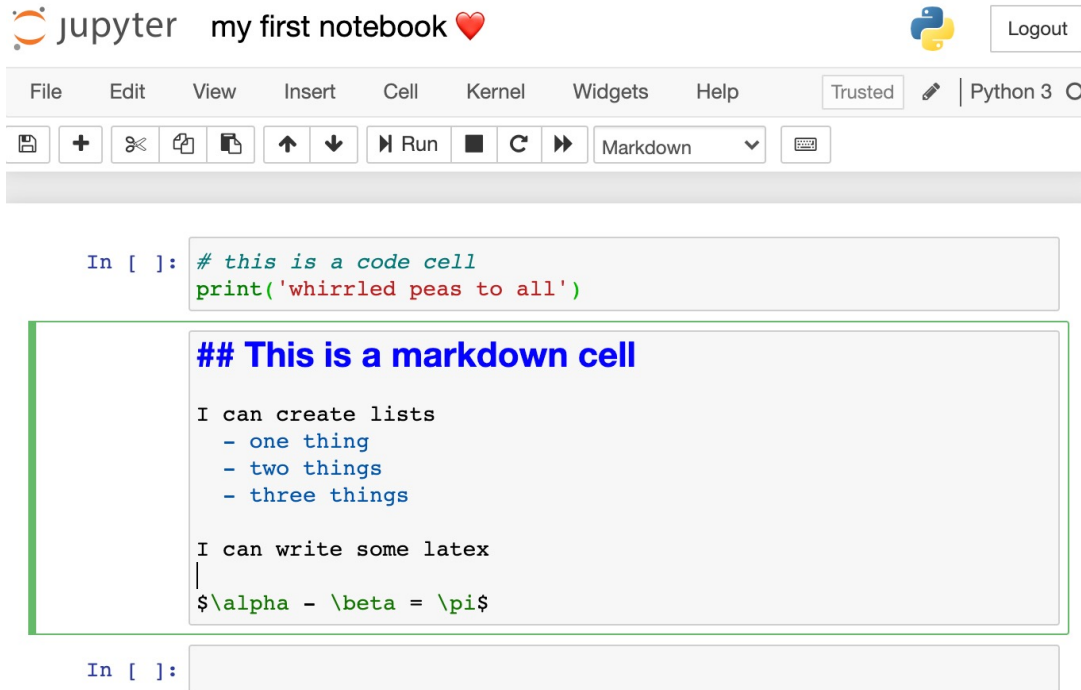
```
[ ] seconds_in_a_day = 24 * 60 * 60
    seconds_in_a_day
```

The output of the code is displayed as `86400`.

At the bottom of the screenshot, the text reads: "To execute the code in the above cell, select it with a click and then either press the play button to the left of the code, or use the"

Your first notebook ❤️

Before eval



The Jupyter Notebook interface shows a menu bar with File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. Below the menu is a toolbar with icons for saving, adding, deleting, copying, pasting, undo, redo, and running. The notebook has a title bar that says "jupyter my first notebook ❤️" and a "Logout" button. The first cell is a code cell with the following text:

```
In [ ]: # this is a code cell
print('whirled peas to all')
```

The second cell is a markdown cell with the following text:

This is a markdown cell

I can create lists

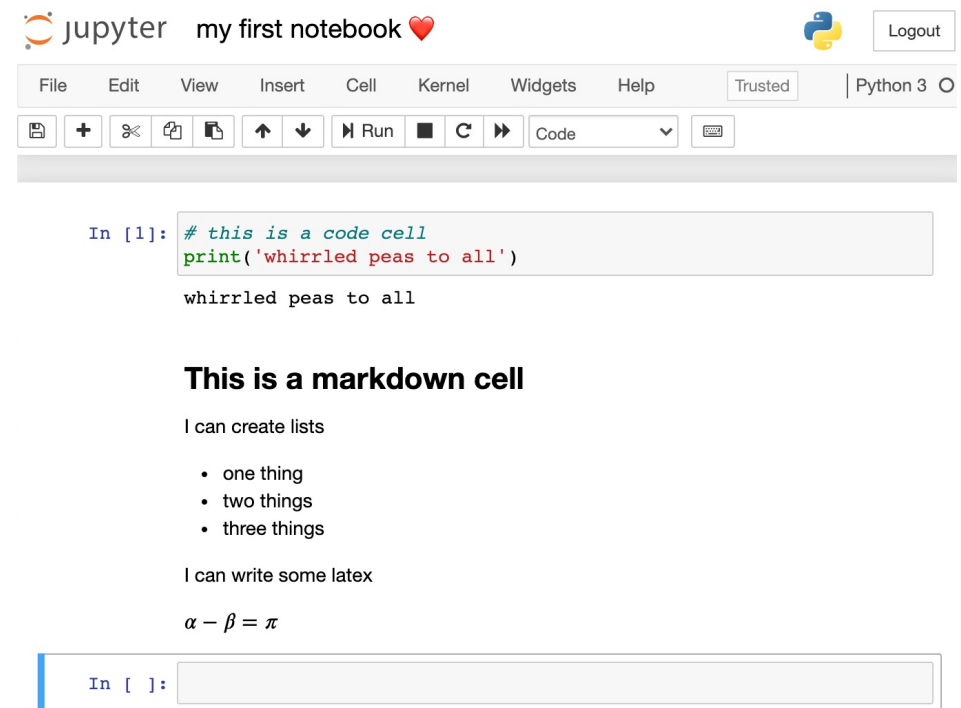
- one thing
- two things
- three things

I can write some latex

$$\alpha - \beta = \pi$$

The third cell is an empty code cell with the text "In []:".

After eval



The Jupyter Notebook interface shows the same menu bar and toolbar as before. The notebook title bar is the same. The first cell is a code cell with the following text:

```
In [1]: # this is a code cell
print('whirled peas to all')
```

The output of the first cell is displayed below the code cell:

```
whirled peas to all
```

The second cell is a markdown cell with the following text:

This is a markdown cell

I can create lists

- one thing
- two things
- three things

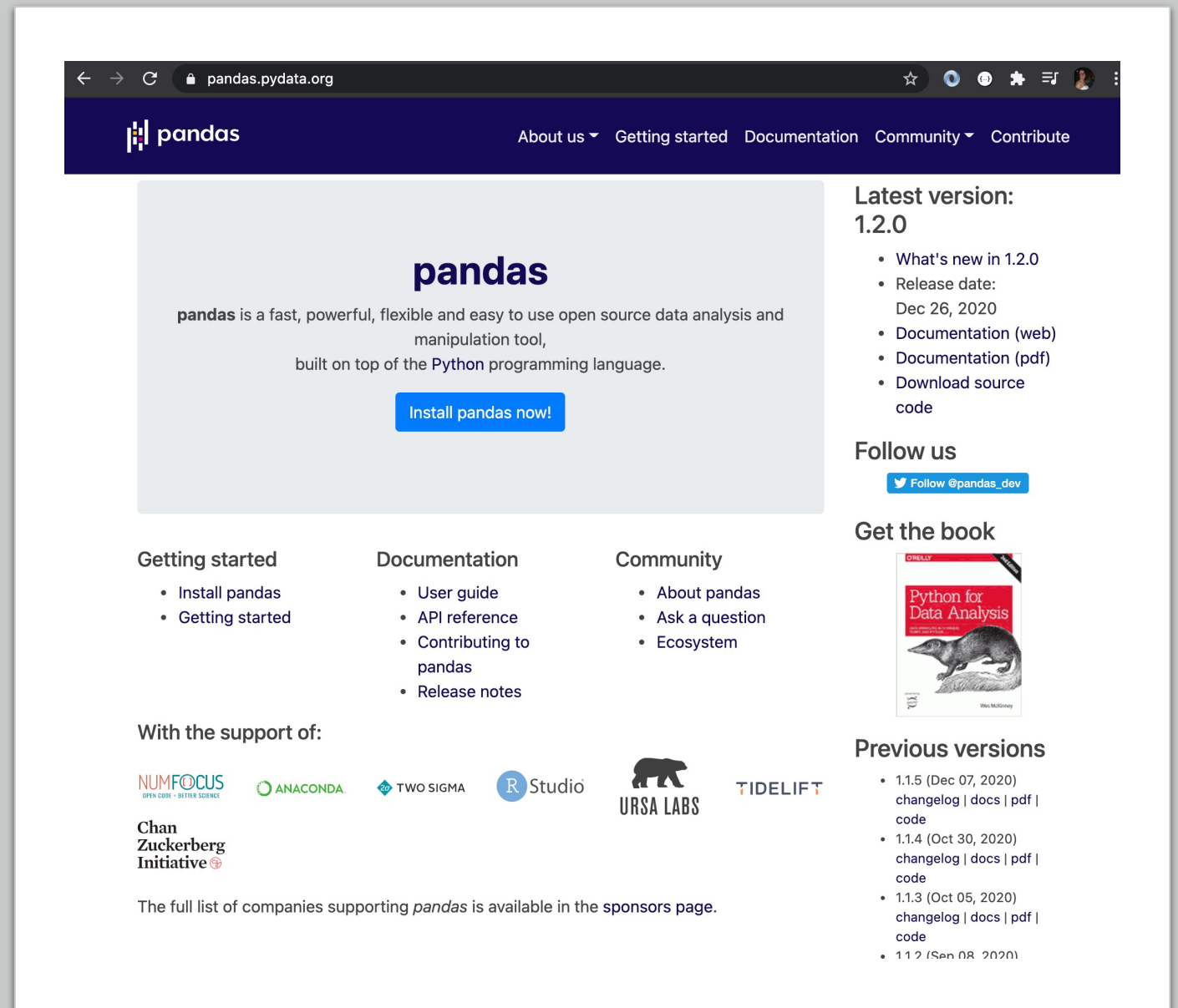
I can write some latex

$$\alpha - \beta = \pi$$

The third cell is an empty code cell with the text "In []:".

pandas – your first package

- <https://pandas.pydata.org/>
- To install a package into a python environment there are many ways, it can be confusing to know what to use!
- The “pip” command is the most widely used



The screenshot shows the pandas website at pandas.pydata.org. The header is dark blue with the pandas logo and navigation links: About us, Getting started, Documentation, Community, and Contribute. The main content area has a light blue background with the pandas logo and a description: "pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language." Below this is a blue button that says "Install pandas now!". To the right, under "Latest version: 1.2.0", there is a list of links: "What's new in 1.2.0", "Release date: Dec 26, 2020", "Documentation (web)", "Documentation (pdf)", and "Download source code". Below this is a "Follow us" section with a Twitter button for @pandas_dev. Further down is a "Get the book" section featuring the cover of "Python for Data Analysis" by Wes McKinney. At the bottom, there is a "With the support of:" section with logos for NUMFOCUS, ANACONDA, TWO SIGMA, R Studio, URSA LABS, and TIDELIFT. Below these logos is the Chan Zuckerberg Initiative logo and a link to the sponsors page.

pandas

pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.

[Install pandas now!](#)

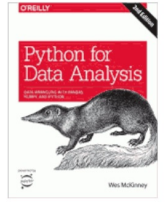
Latest version: 1.2.0

- What's new in 1.2.0
- Release date: Dec 26, 2020
- Documentation (web)
- Documentation (pdf)
- Download source code

Follow us

[Follow @pandas_dev](#)







Get the book



Previous versions

- 1.1.5 (Dec 07, 2020) [changelog](#) | [docs](#) | [pdf](#) | [code](#)
- 1.1.4 (Oct 30, 2020) [changelog](#) | [docs](#) | [pdf](#) | [code](#)
- 1.1.3 (Oct 05, 2020) [changelog](#) | [docs](#) | [pdf](#) | [code](#)
- 1.1.2 (Sep 08, 2020)

With the support of:

Chan Zuckerberg Initiative

The full list of companies supporting *pandas* is available in the [sponsors page](#).

Installing Pandas with pip via Jupyter Notebook

$$\alpha - \beta = \pi$$

```
In [2]: !pip install pandas
```

```
Requirement already satisfied: pandas in /Users/chaney/opt/anaconda3/lib/python3.8/site-packages (1.0.5)  
Requirement already satisfied: pytz>=2017.2 in /Users/chaney/opt/anaconda3/lib/python3.8/site-packages (from pandas) (2020.1)  
Requirement already satisfied: python-dateutil>=2.6.1 in /Users/chaney/opt/anaconda3/lib/python3.8/site-packages (from pandas) (2.8.1)  
Requirement already satisfied: numpy>=1.13.3 in /Users/chaney/opt/anaconda3/lib/python3.8/site-packages (from pandas) (1.18.5)  
Requirement already satisfied: six>=1.5 in /Users/chaney/opt/anaconda3/lib/python3.8/site-packages (from python-dateutil>=2.6.1->pandas) (1.15.0)
```

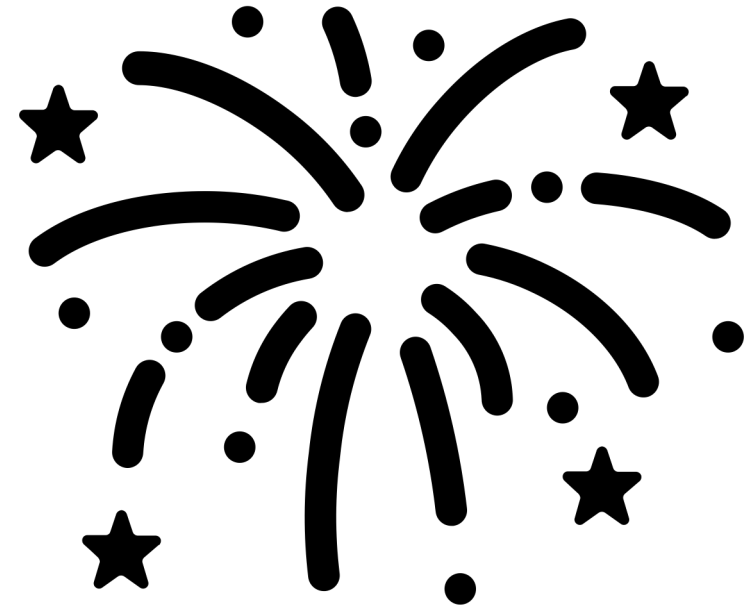
```
In [3]: import pandas as pd
```

```
In [5]: df = pd.DataFrame( {'a':[1,2,3], 'b':['a', 'b', 'c']})
```

```
In [6]: df
```

```
Out[6]:
```

	a	b
0	1	a
1	2	b
2	3	c

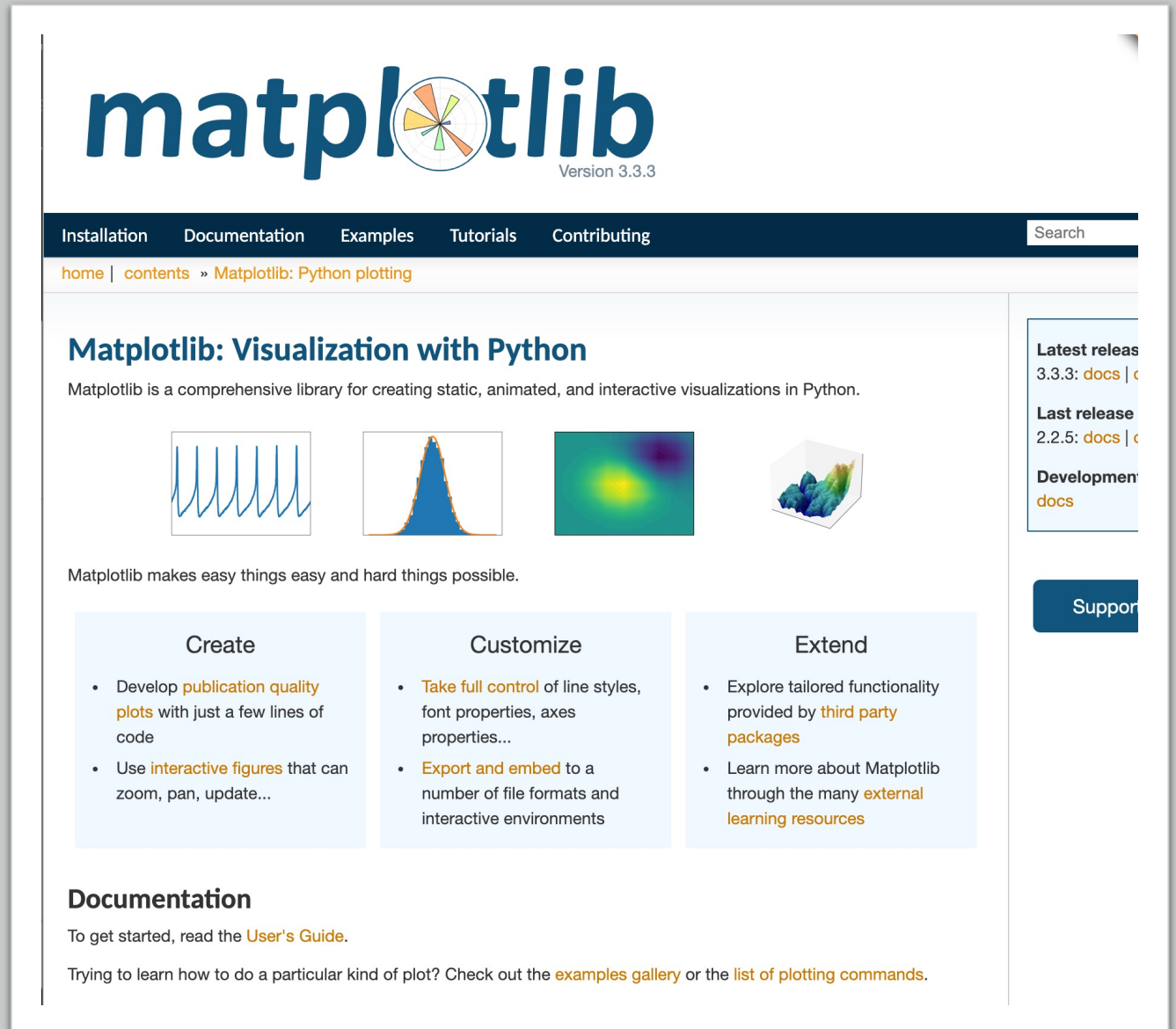


matplotlib

<https://matplotlib.org/>

The most common
visualization tool in
scientific python

Your second package



The screenshot shows the Matplotlib website homepage. At the top is the Matplotlib logo with the text "matplotlib" in a large, blue, sans-serif font, followed by "Version 3.3.3" in a smaller font. Below the logo is a dark blue navigation bar with white text links: "Installation", "Documentation", "Examples", "Tutorials", and "Contributing". To the right of these links is a search bar with the placeholder text "Search". Below the navigation bar is a breadcrumb trail: "home | contents » Matplotlib: Python plotting". The main content area has a heading "Matplotlib: Visualization with Python" and a subheading "Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python." Below this are four small images: a line plot with multiple peaks, a histogram with a normal distribution curve, a 2D heatmap, and a 3D surface plot. Below the images is the text "Matplotlib makes easy things easy and hard things possible." Below this are three columns of text, each with a heading and a list of bullet points. The first column is titled "Create" and lists two bullet points: "Develop publication quality plots with just a few lines of code" and "Use interactive figures that can zoom, pan, update...". The second column is titled "Customize" and lists two bullet points: "Take full control of line styles, font properties, axes properties..." and "Export and embed to a number of file formats and interactive environments". The third column is titled "Extend" and lists two bullet points: "Explore tailored functionality provided by third party packages" and "Learn more about Matplotlib through the many external learning resources". To the right of the main content area is a sidebar with a "Support" button and a section titled "Latest releases" with links to "3.3.3: docs" and "2.2.5: docs". Below this is a section titled "Development" with a link to "docs".


matplotlib
Version 3.3.3

Installation Documentation Examples Tutorials Contributing

home | contents » Matplotlib: Python plotting

Matplotlib: Visualization with Python

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.



Matplotlib makes easy things easy and hard things possible.

Create

- Develop **publication quality plots** with just a few lines of code
- Use **interactive figures** that can zoom, pan, update...

Customize

- **Take full control** of line styles, font properties, axes properties...
- **Export and embed** to a number of file formats and interactive environments

Extend

- Explore tailored functionality provided by **third party packages**
- Learn more about Matplotlib through the many **external learning resources**

Documentation

To get started, read the **User's Guide**.

Trying to learn how to do a particular kind of plot? Check out the **examples gallery** or the **list of plotting commands**.

Support

Latest releases
3.3.3: [docs](#) | [changelog](#)

Last release
2.2.5: [docs](#) | [changelog](#)

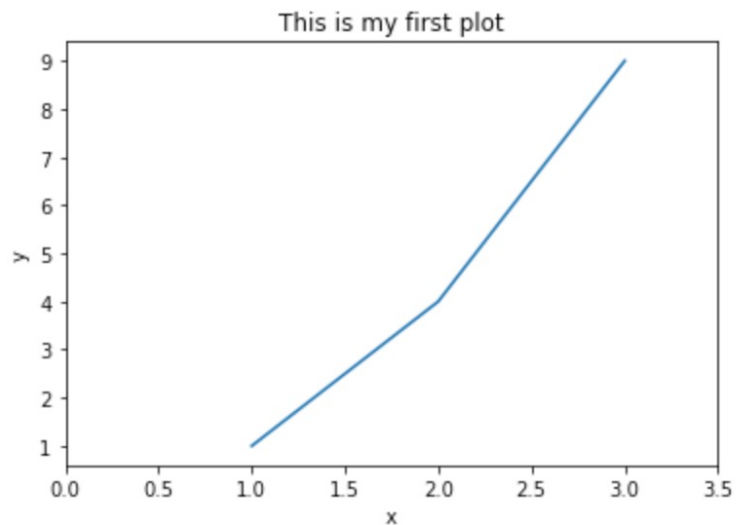
Development
[docs](#)

```
In [ ]: !pip install matplotlib
```

```
In [12]: import matplotlib.pyplot as plt
```

```
In [16]: plt.plot( [1,2,3],[1,4,9])  
plt.xlabel('x')  
plt.ylabel('y')  
plt.title('This is my first plot')  
plt.xlim([0,3.5])
```

```
Out[16]: (0.0, 3.5)
```



Install
matplotlib
and make a
plot