

Line Graphs using matplotlib

- Your scientist has asked you to plot the following two functions:

$$y_1 = 2x - 5$$
$$y_2 = -0.3x^2 + 15$$

- The domain for both functions is $-10 \leq x \leq 10$
- You should plot both curves on the same graph

matplotlib

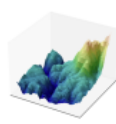
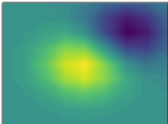
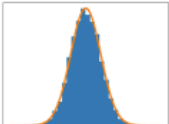
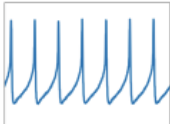
Version 3.3.3

InstallationDocumentationExamplesTutorialsContributing

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

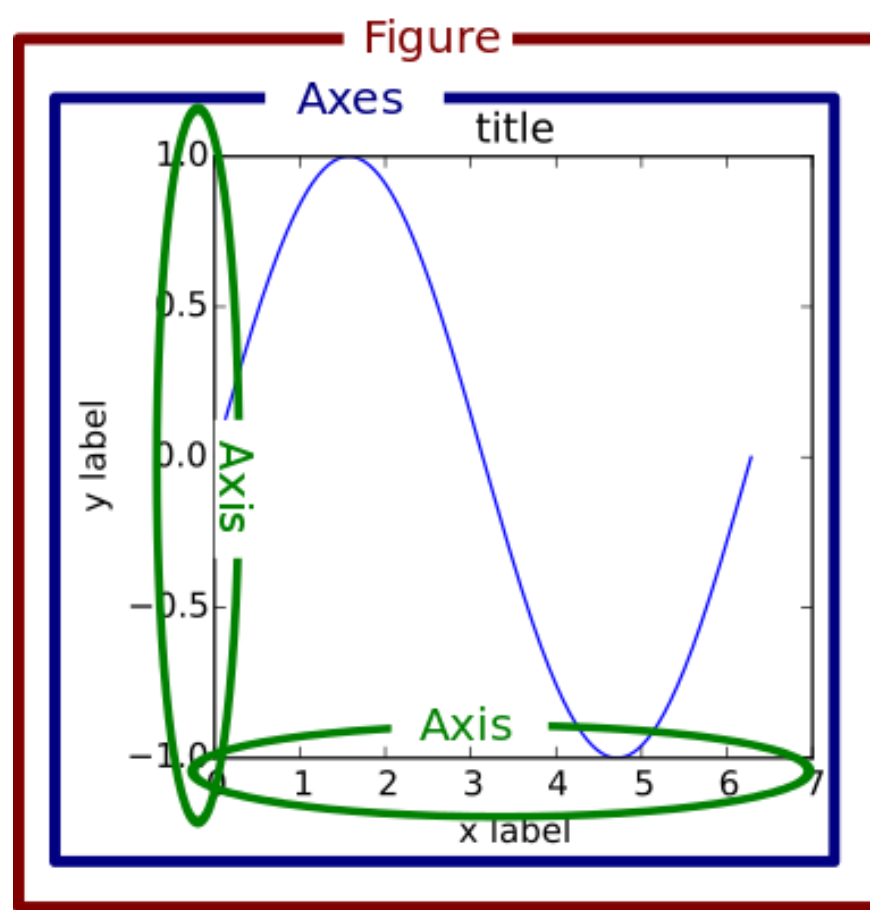
- Explore tailored functionality provided by **third party packages**
- Learn more about Matplotlib through the many **external learning resources**

Documentation

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

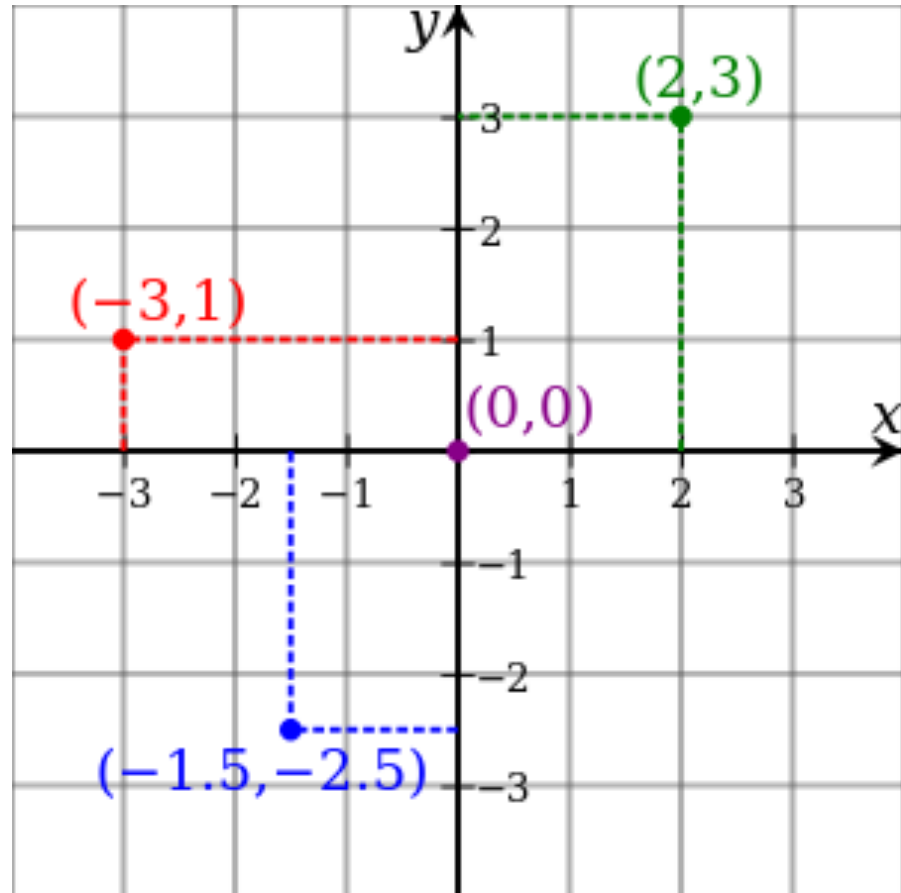
Trying to learn how to do a particular kind of plot? Check out the **examples gallery** or the **list of plotting commands**.

Matplotlib Container Hierarchy



Cartesian Coordinates

Created by
René Descartes
in 1637



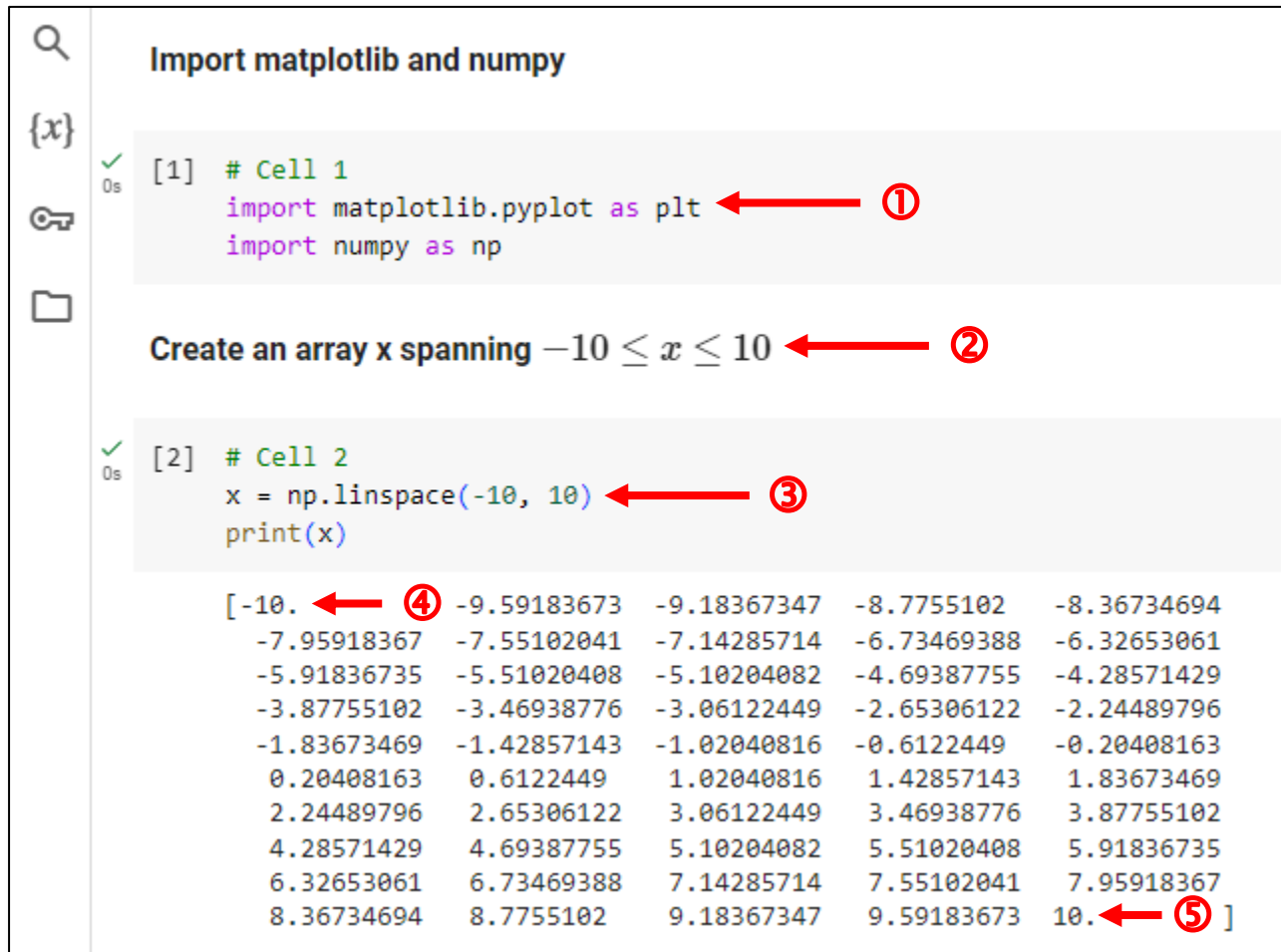
Line Graphs using matplotlib

- Your scientist has asked you to plot the following two functions:

$$y_1 = 2x - 5$$
$$y_2 = -0.3x^2 + 15$$

- The domain for both functions is $-10 \leq x \leq 10$
- You should plot both curves on the same graph

Edit line_graphs.ipynb – Cells 1..2



Import matplotlib and numpy

```
[1] # Cell 1
import matplotlib.pyplot as plt ← ①
import numpy as np
```

Create an array x spanning $-10 \leq x \leq 10$ ← ②

```
[2] # Cell 2
x = np.linspace(-10, 10) ← ③
print(x)
```

```
[-10. ← ④ -9.59183673 -9.18367347 -8.7755102 -8.36734694
 -7.95918367 -7.55102041 -7.14285714 -6.73469388 -6.32653061
 -5.91836735 -5.51020408 -5.10204082 -4.69387755 -4.28571429
 -3.87755102 -3.46938776 -3.06122449 -2.65306122 -2.24489796
 -1.83673469 -1.42857143 -1.02040816 -0.6122449 -0.20408163
 0.20408163 0.6122449 1.02040816 1.42857143 1.83673469
 2.24489796 2.65306122 3.06122449 3.46938776 3.87755102
 4.28571429 4.69387755 5.10204082 5.51020408 5.91836735
 6.32653061 6.73469388 7.14285714 7.55102041 7.95918367
 8.36734694 8.7755102 9.18367347 9.59183673 10. ← ⑤ ]
```

Edit line_graphs.ipynb – Cells 3..4

```
Set  $y_1 = 2x - 5$  ← ①

[3] # Cell 3
y1 = 2 * x - 5 ← ②
print(y1)

[-25.      -24.18367347 -23.36734694 -22.55102041 -21.73469388
 -20.91836735 -20.10204082 -19.28571429 -18.46938776 -17.65306122
 -16.83673469 -16.02040816 -15.20408163 -14.3877551  -13.57142857
 -12.75510204 -11.93877551 -11.12244898 -10.30612245  -9.48979592
  -8.67346939  -7.85714286  -7.04081633  -6.2244898   -5.40816327
  -4.59183673  -3.7755102  -2.95918367  -2.14285714  -1.32653061
  -0.51020408  0.30612245  1.12244898  1.93877551  2.75510204
   3.57142857  4.3877551   5.20408163  6.02040816  6.83673469
   7.65306122  8.46938776  9.28571429 10.10204082 10.91836735
  11.73469388 12.55102041 13.36734694 14.18367347 15. ← ③]

Set  $y_2 = -0.3x^2 + 15$  ← ④

[4] # Cell 4
y2 = -0.3 * x**2 + 15 ← ⑤
print(y2)

[-15.      -12.60099958 -10.30195752  -8.1028738  -6.00374844
 -4.00458142 -2.10537276  -0.30612245   1.39316951   2.99250312
  4.49187838  5.89129529  7.19075385   8.39025406   9.48979592
 10.48937943 11.38900458 12.18867139 12.88837984 13.48812995
 13.9879217 14.3877551 14.68763015 14.88754686 14.98750521
 14.98750521 14.88754686 14.68763015 14.3877551 13.9879217
 13.48812995 12.88837984 12.18867139 11.38900458 10.48937943
  9.48979592  8.39025406  7.19075385  5.89129529  4.49187838
  2.99250312  1.39316951 -0.30612245 -2.10537276 -4.00458142
 -6.00374844 -8.1028738 -10.30195752 -12.60099958 -15. ← ⑥]
```

$$= 2(10) - 5$$

$$= 20 - 5$$

$$= 15$$

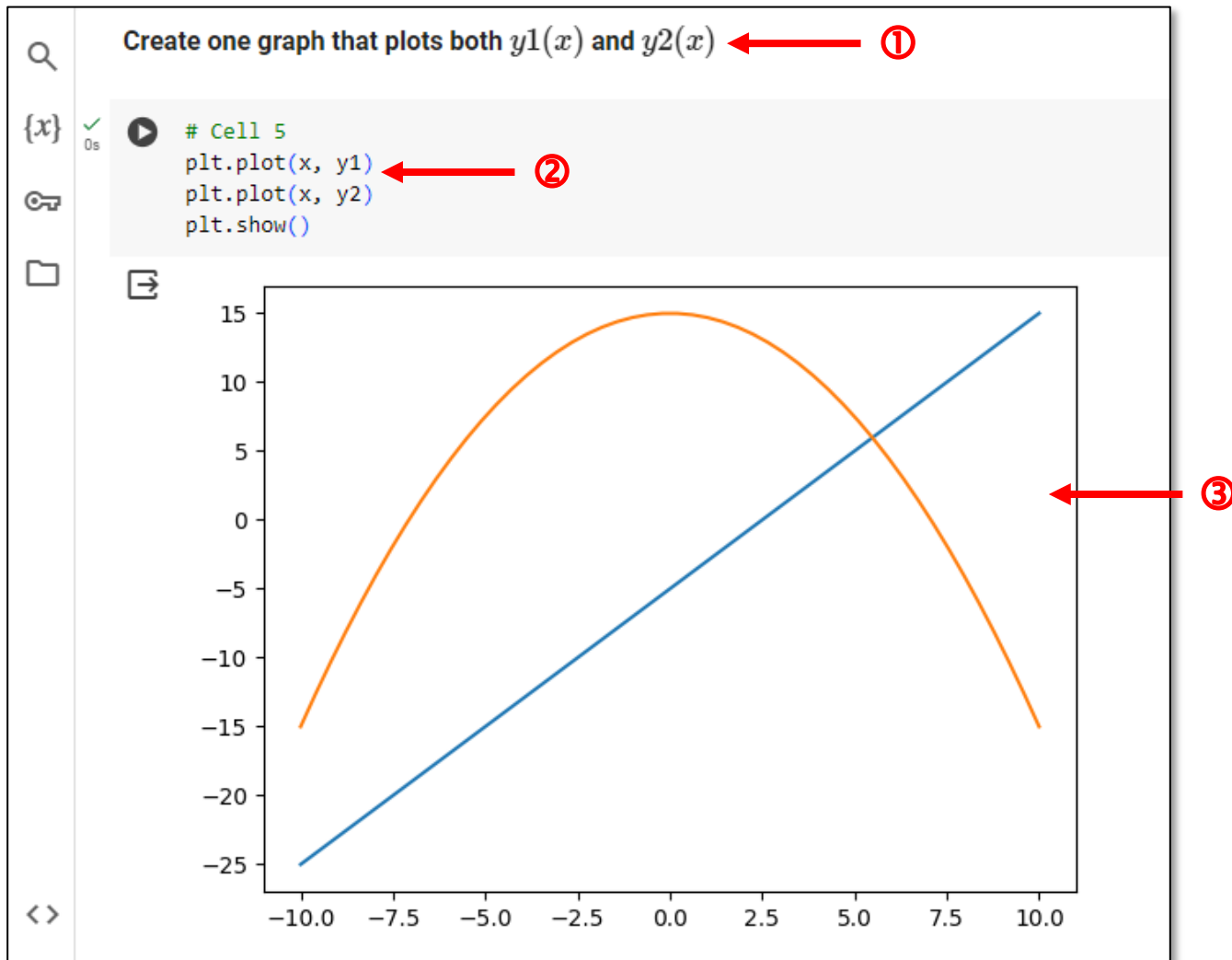
$$= -0.3(10^2) + 15$$

$$= -0.3(100) + 15$$

$$= -30 + 15$$

$$= -15$$

Edit line_graphs.ipynb – Cell 5



PEP 8 – Style Guide for Python Code

<https://peps.python.org/pep-0008>

Python Enhancement Proposals | [Python](#) » [PEP Index](#) » [PEP 8](#)

Contents

- Introduction
- A Foolish Consistency is the Hobgoblin of Little Minds
- Code Lay-out
 - Indentation
 - Tabs or Spaces?
 - Maximum Line Length
 - Should a Line Break Before or After a Binary Operator?
 - Blank Lines
 - Source File Encoding
 - Imports
 - Module Level Dunder Names
- String Quotes
- Whitespace in Expressions and Statements
 - Pet Peeves
 - Other Recommendations
- When to Use Trailing Commas
- Comments
 - Block Comments
 - Inline Comments
 - Documentation Strings
- Naming Conventions
 - Overriding Principle
 - Descriptive: Naming Styles

PEP 8 – Style Guide for Python Code

Author: Guido van Rossum <guido@python.org>, Barry Warsaw <barry@python.org>, Nick Coghlan <ncoghlan@gmail.com>

Status: *Active*

Type: *Process*

Created: 05-Jul-2001

Post-History: 05-Jul-2001, 01-Aug-2013

► **Table of Contents**

[Introduction](#)

This document gives coding conventions for the Python code comprising the standard library in the main Python distribution. Please see the companion informational PEP describing [style guidelines for the C code in the C implementation of Python](#).

This document and [PEP 257](#) (Docstring Conventions) were adapted from Guido's original Python Style Guide essay, with some additions from Barry's style guide [2].

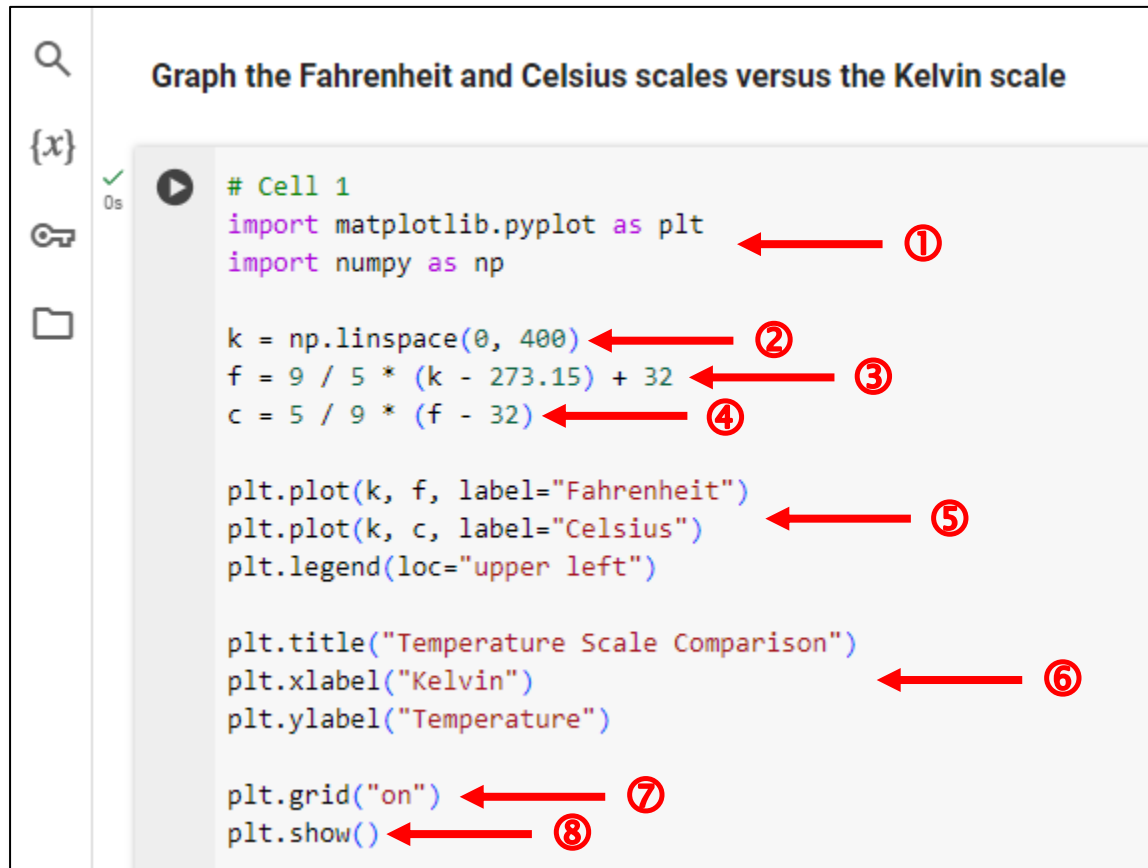
This style guide evolves over time as additional conventions are identified and past conventions are rendered obsolete by changes in the language itself.

Temperature Conversion

- Your scientist has asked you to plot the Fahrenheit and Celsius temperature equivalents for temperatures in Kelvin that span from 0K to 400K
- Your plot should label each temperature scale line graph so a legend can be added to the plot
- As with all professional graphs, each axis should be labeled with the appropriate units
- The graph should have a title and a grid for easier reading of the values
- The research question is, "**What is the one temperature that is the same in both Fahrenheit and Celsius?**"

Open fahrenheit_to_celsius.ipynb – Cell 1

Note: You should not edit this file!



```
# Cell 1
import matplotlib.pyplot as plt
import numpy as np

k = np.linspace(0, 400)
f = 9 / 5 * (k - 273.15) + 32
c = 5 / 9 * (f - 32)

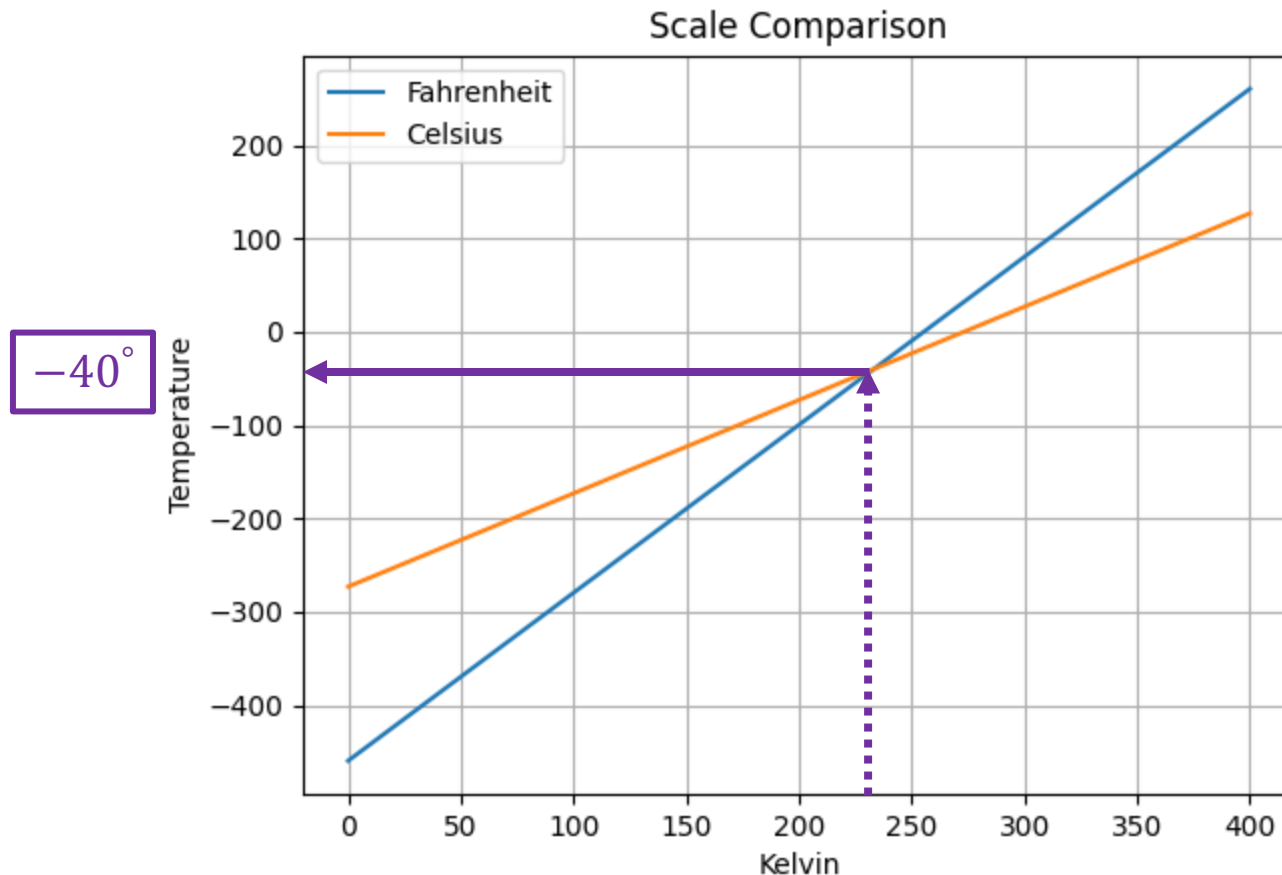
plt.plot(k, f, label="Fahrenheit")
plt.plot(k, c, label="Celsius")
plt.legend(loc="upper left")

plt.title("Temperature Scale Comparison")
plt.xlabel("Kelvin")
plt.ylabel("Temperature")

plt.grid("on")
plt.show()
```

The image shows a Jupyter Notebook interface with a sidebar on the left containing icons for search, variables, keys, and files. The main area displays a code cell titled "Graph the Fahrenheit and Celsius scales versus the Kelvin scale". The code cell is labeled "# Cell 1" and contains Python code for plotting. Red arrows and circled numbers 1 through 8 point to specific lines of code: 1 points to the first import statement, 2 points to the linspace function call, 3 points to the first conversion formula, 4 points to the second conversion formula, 5 points to the plot function calls, 6 points to the title and axis labels, 7 points to the grid function call, and 8 points to the show function call.

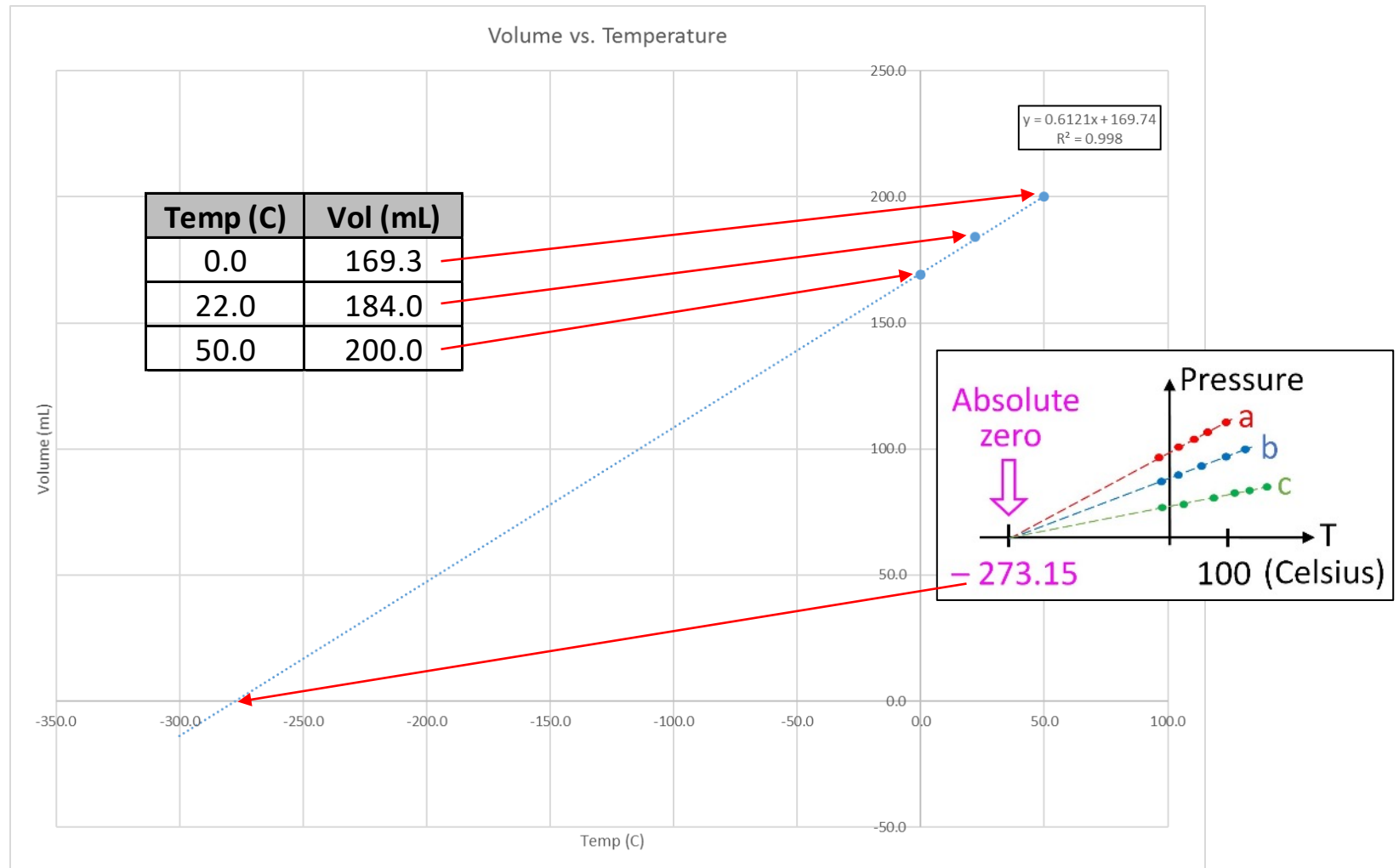
Run fahrenheit_to_celsius.ipynb – Cell 1



What is the one temperature that is the same
in Fahrenheit and Celsius?

How did we calculate absolute zero in **1779**?

($PV = nRT$)



Task 03

- Update the code in **plot_quintic.ipynb** to graph this polynomial:

$$y = x^5 - 2x^4 - 120x^3 + 22x^2 + 2119x + 1980$$

- The domain should be $-10 \leq x \leq 12$
- If you prefer, you can use this equivalent expression for y :
$$y = (x + 9)(x + 4)(x + 1)(x - 5)(x - 11)$$
- What does the **Fundamental Theorem of Algebra** tell us about the **maximum** number of places $y(x)$ *might* cross the **x-axis** in the domain of real (\mathbb{R}) numbers?