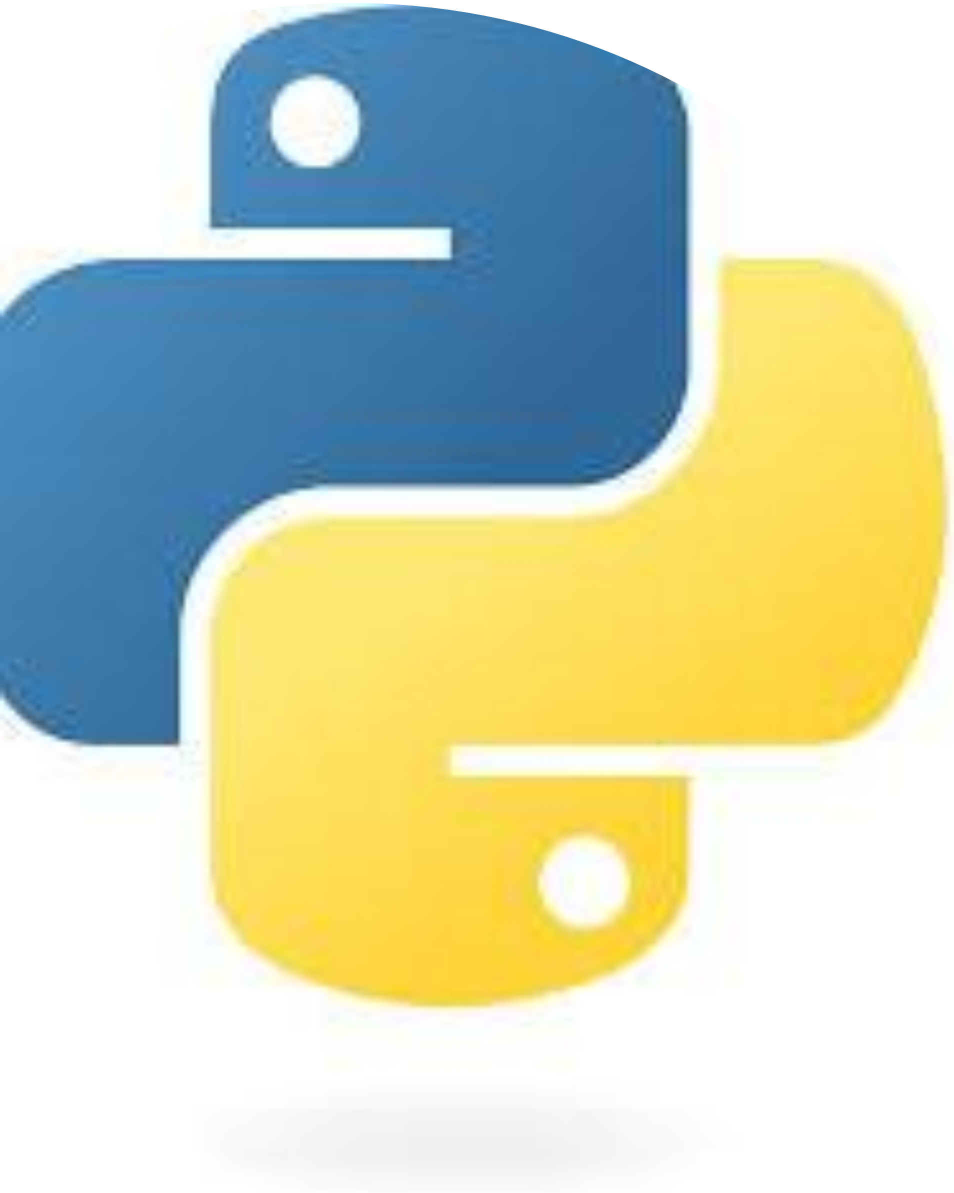


- MZ & JK

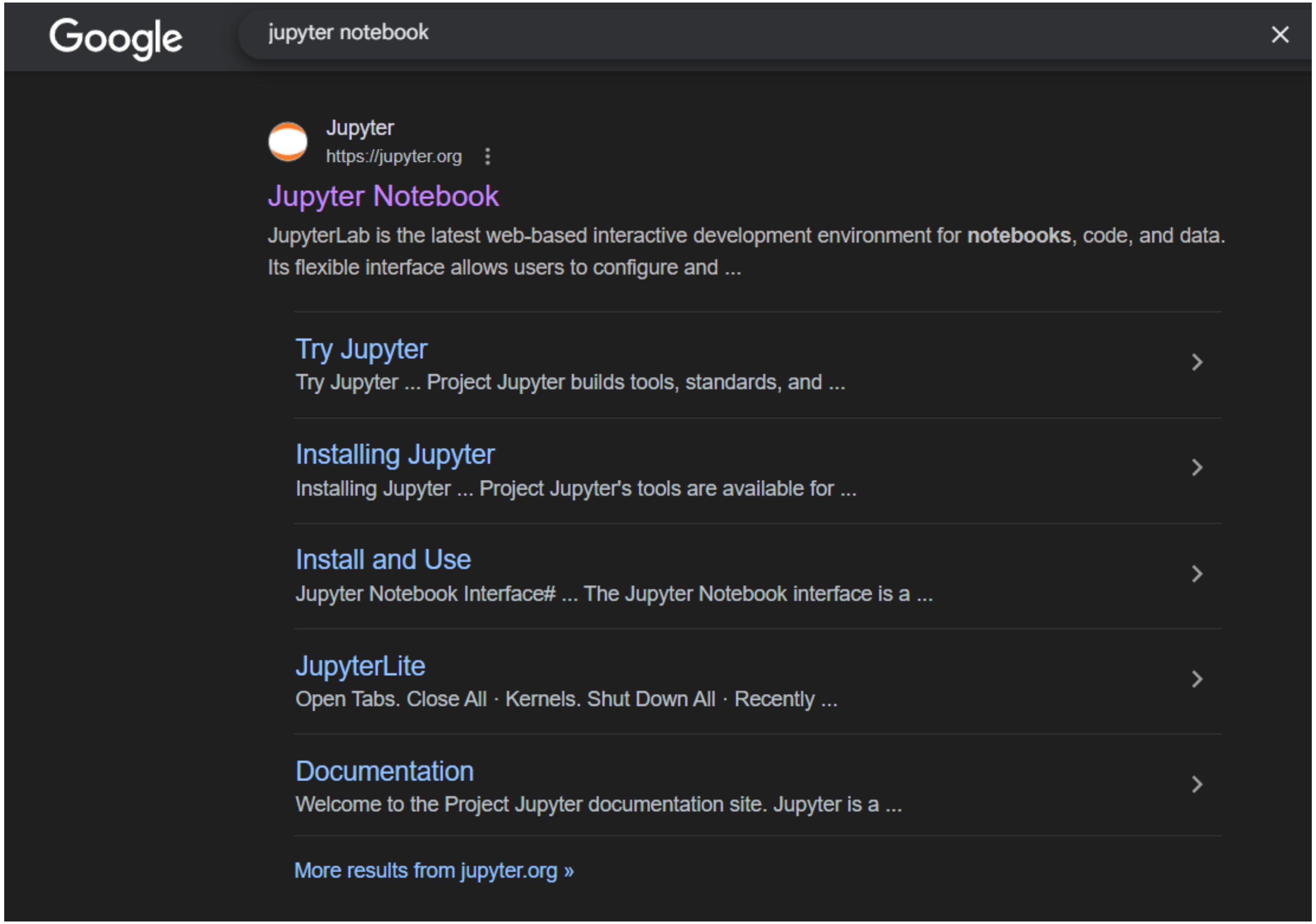
Introduction to Python programming

Introduction

- What is a Python?
- Why to learn it?
- Why Jupyter Notebook?

Jupyter Notebook

https://jupyter.org/try-jupyter/notebooks/?path=notebooks/Intro.ipynb



Jupyter
https://jupyter.org

Jupyter Notebook

JupyterLab is the latest web-based interactive development environment for **notebooks**, code, and data. Its flexible interface allows users to configure and ...

Try Jupyter
Try Jupyter ... Project Jupyter builds tools, standards, and ...

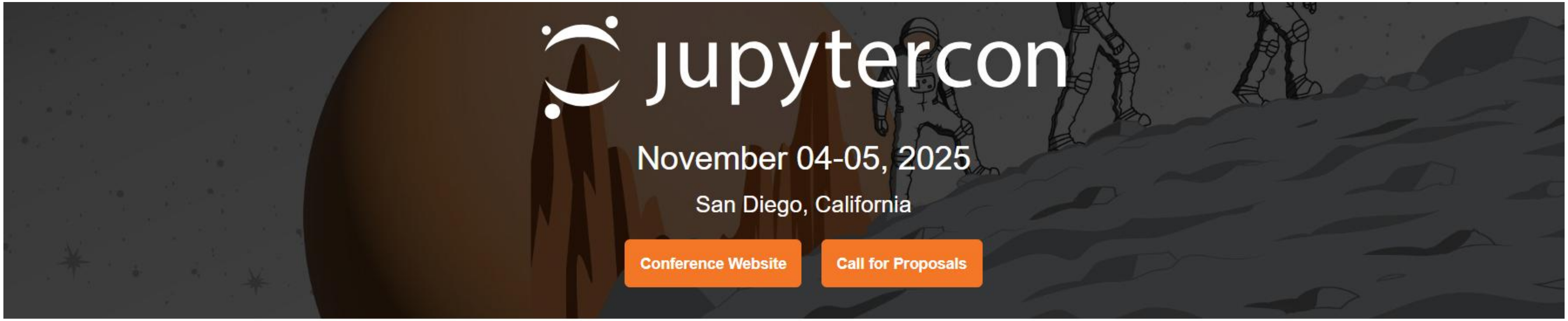
Installing Jupyter
Installing Jupyter ... Project Jupyter's tools are available for ...

Install and Use
Jupyter Notebook Interface# ... The Jupyter Notebook interface is a ...

JupyterLite
Open Tabs. Close All · Kernels. Shut Down All · Recently ...

Documentation
Welcome to the Project Jupyter documentation site. Jupyter is a ...

More results from jupyter.org »

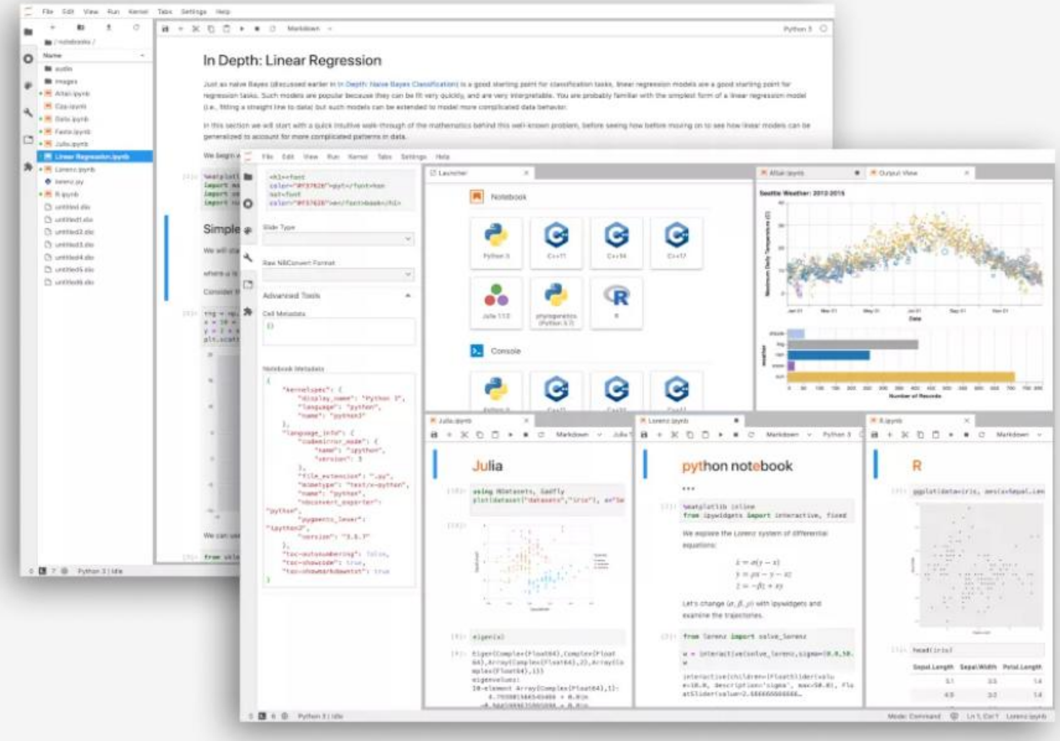


jupytercon

November 04-05, 2025

San Diego, California

s://events.linuxfoundation.org/jupytercon/

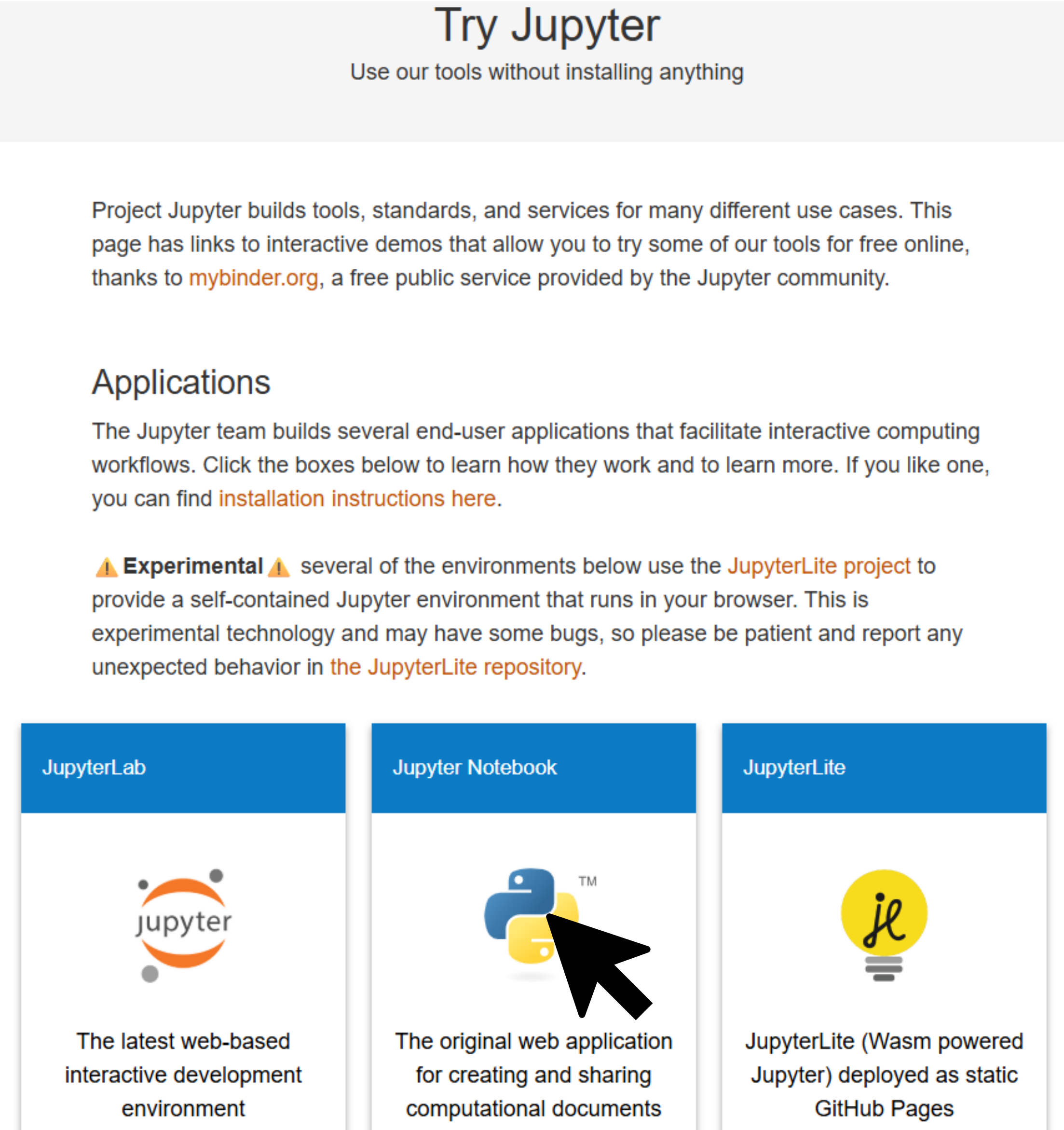


JupyterLab: A Next-Generation Notebook Interface

JupyterLab is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality.

Try it in your browser

Install JupyterLab



Try Jupyter

Use our tools without installing anything

Project Jupyter builds tools, standards, and services for many different use cases. This page has links to interactive demos that allow you to try some of our tools for free online, thanks to mybinder.org, a free public service provided by the Jupyter community.

Applications

The Jupyter team builds several end-user applications that facilitate interactive computing workflows. Click the boxes below to learn how they work and to learn more. If you like one, you can find installation instructions here.

⚠ **Experimental** ⚠ several of the environments below use the JupyterLite project to provide a self-contained Jupyter environment that runs in your browser. This is experimental technology and may have some bugs, so please be patient and report any unexpected behavior in the JupyterLite repository.

JupyterLab

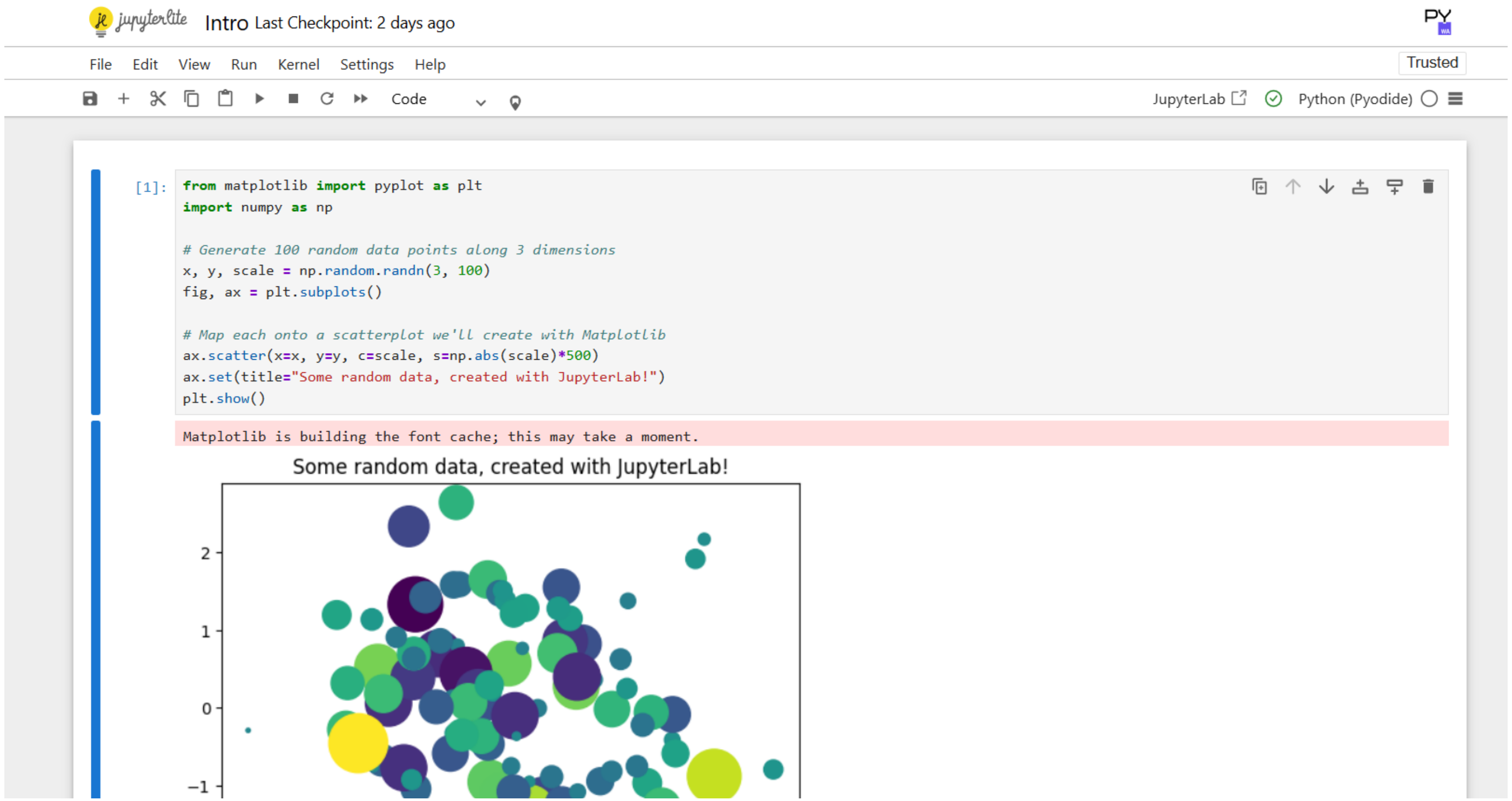
The latest web-based interactive development environment

Jupyter Notebook

The original web application for creating and sharing computational documents

JupyterLite

JupyterLite (Wasm powered Jupyter) deployed as static GitHub Pages



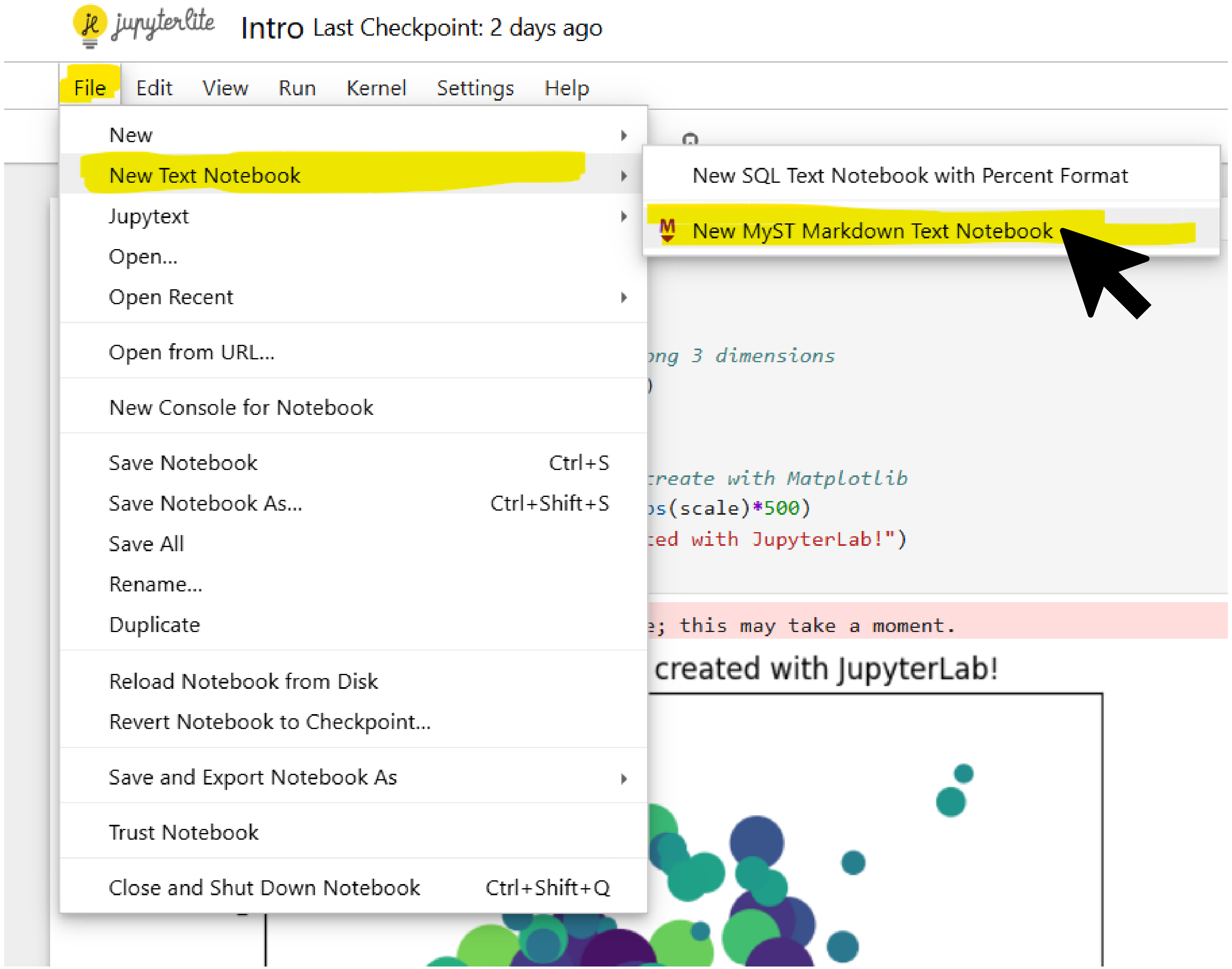
```
from matplotlib import pyplot as plt
import numpy as np

# Generate 100 random data points along 3 dimensions
x, y, scale = np.random.randn(3, 100)
fig, ax = plt.subplots()

# Map each onto a scatterplot we'll create with Matplotlib
ax.scatter(x=x, y=y, c=scale, s=np.abs(scale)*500)
ax.set(title="Some random data, created with JupyterLab!")
plt.show()
```

```
Matplotlib is building the font cache; this may take a moment.
```

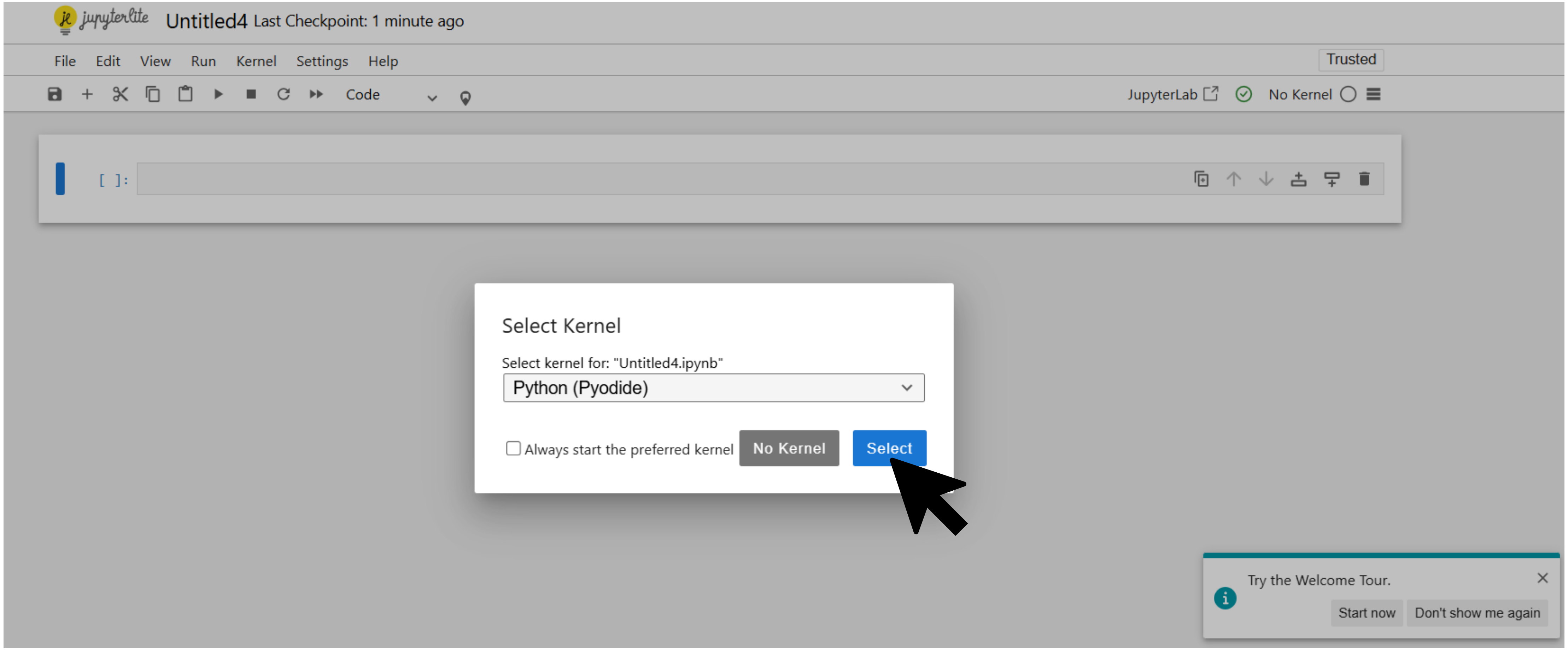
Some random data, created with JupyterLab!


jupyterlite
Intro Last Checkpoint: 2 days ago
File
Edit
View
Run
Kernel
Settings
Help
New
New Text Notebook
Jupytext
Open...
Open Recent
Open from URL...
New Console for Notebook
Save Notebook Ctrl+S
Save Notebook As... Ctrl+Shift+S
Save All
Rename...
Duplicate
Reload Notebook from Disk
Revert Notebook to Checkpoint...
Save and Export Notebook As
Trust Notebook
Close and Shut Down Notebook Ctrl+Shift+Q
New SQL Text Notebook with Percent Format
New MyST Markdown Text Notebook

```
ong 3 dimensions
)

create with Matplotlib
os(scale)*500)
ted with JupyterLab!")
```

```
e; this may take a moment.
```



Select Kernel

Select kernel for: "Untitled4.ipynb"

Python (Pyodide)

□ Always start the preferred kernel

The workshop includes a little bit of theory followed by hands-on work

- Your first python programe
- Print function – outputting text and calculations
- Understanding variables and data types
- Conditions and loops

Basics
1. Functions

```
print("Hello, world!")
```

- So when we write:

```
print("Hello, world!")
```

- The computer will respond by showing:
- *Hello, world!*
- It's like saying: “Hey computer, say hello to the world!”

2. Variables

- x = 5
- Height = 170.5
- name = "Minja„

- So if we say:

print(name)

- The computer will show:
- *Minja*
- Because it goes and looks indise the box called *name*.

- You can use variable in a sentences, like:
- *print(„My name is „ , name, „ and I am“ , age, „ years old.“)*

Summary

- print("Hello, world!")
- x = 5
- name = "Minja„
- Height = 170.5

Line	What it does	Type of data
print("Hello, world!")	Shows a message on screen	Text (string)
x = 5	Stores a number in a variable	Integer (whole number)
name = "Minja„	Stores a name in a variable	String (text)
Height = 170.5	Stores a decimal number in a variable	Double(Decimal)

Task: Introduce Yourself Using Variables

1. Modify the message in the print() function to say:
Hello, Summer School students!
2. Create a string variable called name and assign it your name.
3. Create another string variable called city and assign it the name of the city where you live.
4. Create a numeric variable called year and assign it the year you are attending the Summer School.
5. Use the print() function to display the following sentence, using the variables:

Hi, my name is [name]. I come from [city]. I attended Summer School in [year].
(Replace [name], [city], and [year] with your variables.)

If conditions and for loops

Conditions

```
if x > 3:
        print("x is greater than 3")
```

Challenge

- Try changing the value of x and observe the results:
- x = 7
- x = 2

Task: check if sequence Start Codon appears *anywhere* in a sequence

- DNA = "TAAGTCCAAAGGGAAATTGCTTATGAAAACTGTCATTTTTACTTCTCTG"
- if „ATG“ in DNA:
-     print("Start codon found!")

Loop

```
for i in range(5):
    print(i)
```

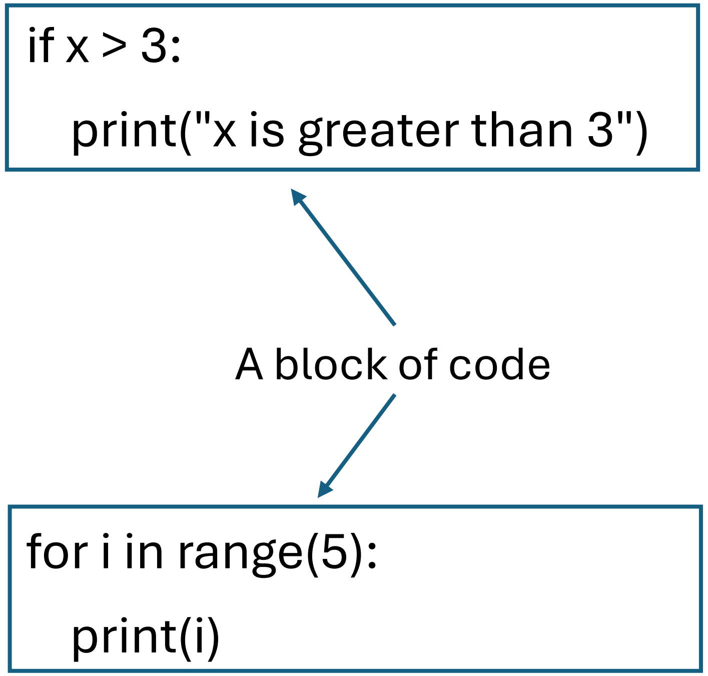
Task

- for i in range(1, 6):
-     print("*" * i)

Challenge

- Try changing the height of the tree to have 10 rows.
- Draw another three with 2 rows.

Summary



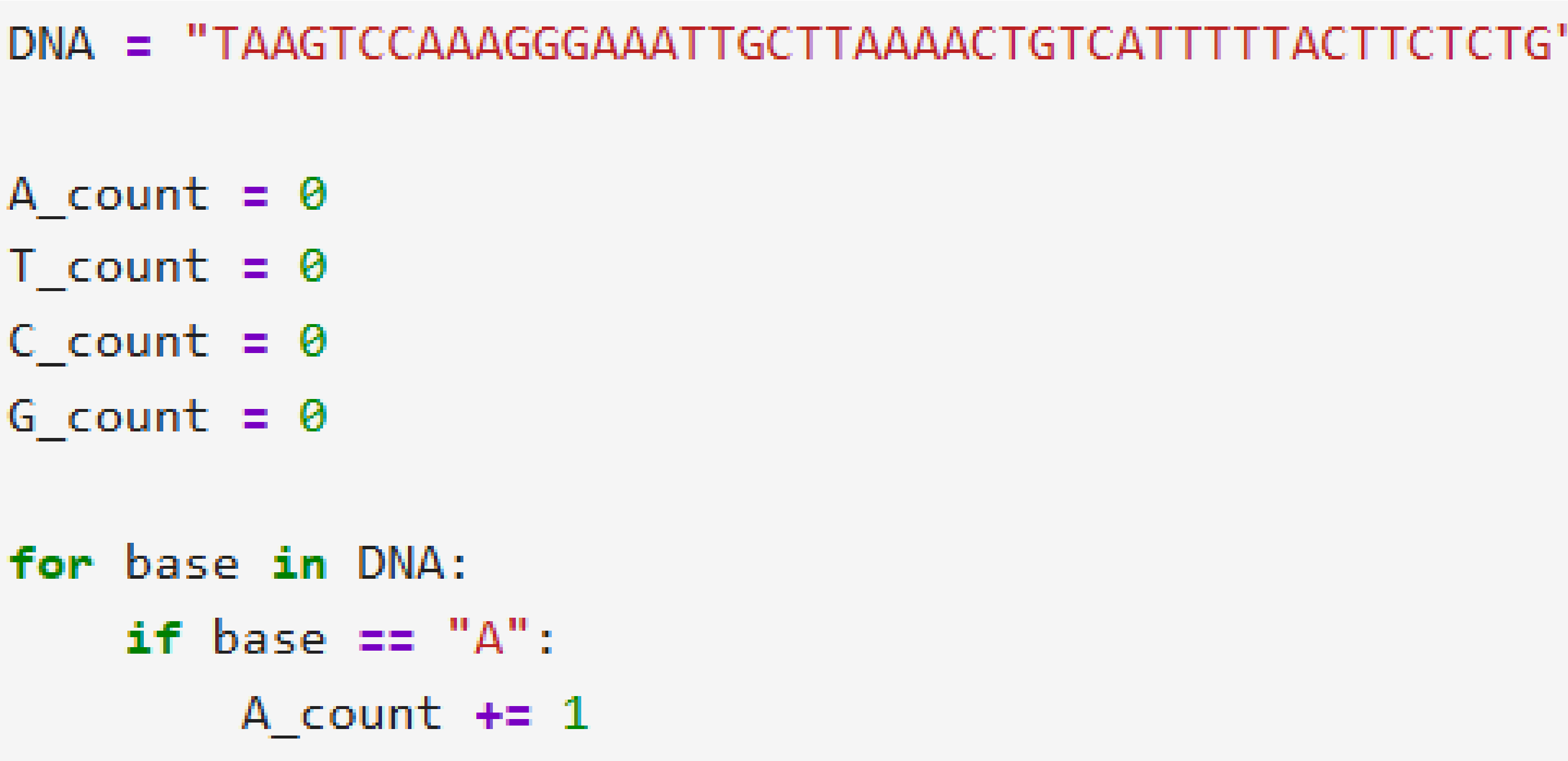
```
if x > 3:
    print("x is greater than 3")
```

A block of code

```
for i in range(5):
    print(i)
```

Code	What it does
if x > 3:	Checks a condition (is x greater than 3?)
print(...) (inside if)	Runs only if the condition is true
for i in range(5):	Loops from 0 to 4 (5 times total)
print(i) (inside for)	Prints the value of i each time through the loop

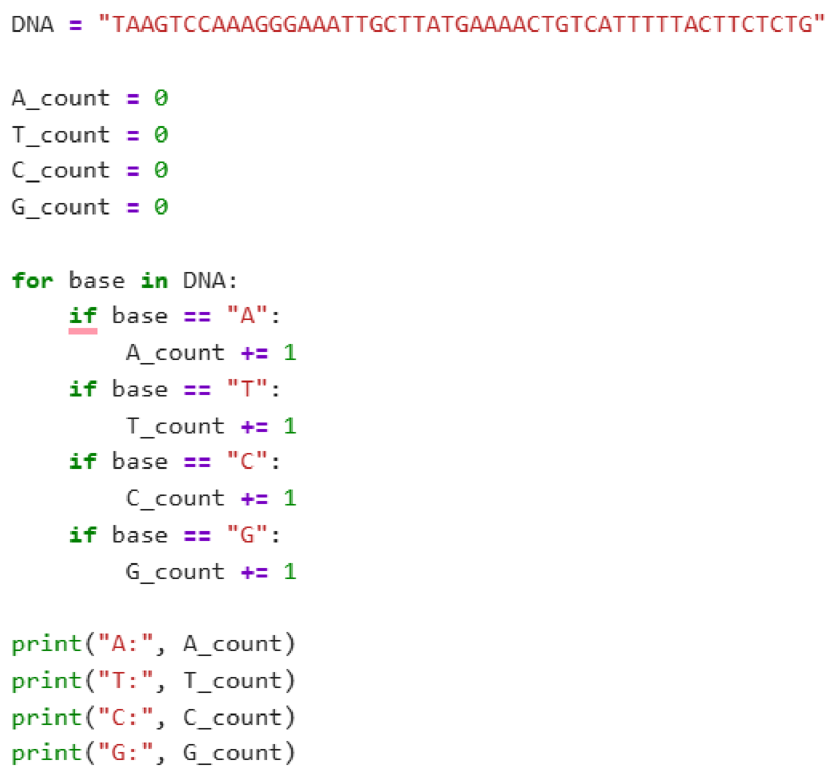
Challenge: Count how many times each base (A, T, C, G) appears in a DNA sequence



```
DNA = "TAAGTCCAAAGGGAAATTGCTTAAAACTGTCATTTTTACTTCTCTG"

A_count = 0
T_count = 0
C_count = 0
G_count = 0

for base in DNA:
    if base == "A":
        A_count += 1
```



```python
DNA = "TAAGTCCAAAGGGAAATTGCTTATGAAAACTGTCATTTTTACTTCTCTG"

A_count = 0
T_count = 0
C_count = 0
G_count = 0

for base in DNA:
    if base == "A":
        A_count += 1
    if base == "T":
        T_count += 1
    if base == "C":
        C_count += 1
    if base == "G":
        G_count += 1

print("A:", A_count)
print("T:", T_count)
print("C:", C_count)
print("G:", G_count)
```