Introduction to Python

Vishal Patel

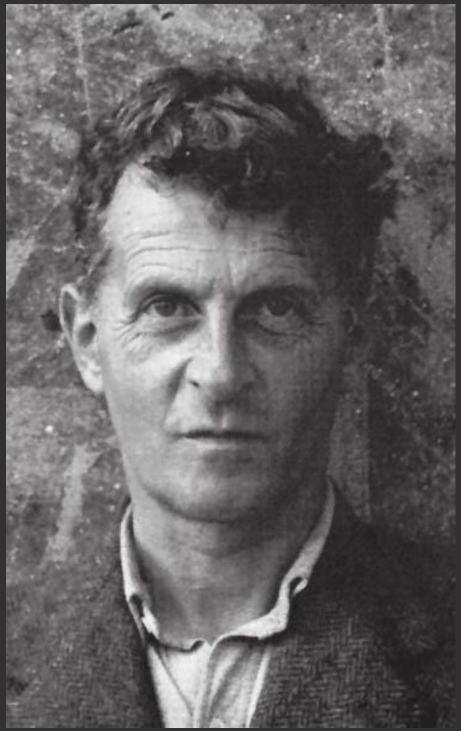
Spring 2026

ARRIVAL

The language you speak (or use) affects the way you think.

Linguistic Relativity / Sapir–Whorf hypothesis

> The limits of my language
>
> mean the limits of my world.
>
> – Ludwig Wittgenstein



Pythonic

Readable

Uncluttered

The Hitchhiker's Guide to Python

Explicit

Simple

Concise

The Zen of Python

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Sparse is better than dense.

Readability counts.

Special cases aren’t special enough to break the rules.

Although practicality beats purity.

Errors should never pass silently.

Unless explicitly silenced. (contd.)

Pearl Programming Motto

There is more than one way to do it.

TIMTOWTDI, pronounced Tim Toady

The Zen of Python

“There should be one

— and preferably only one —

obvious way to do it."

Python is powerful... and fast;
plays well with others;
runs everywhere;
is friendly & easy to learn;
is Open.

“Python is an interpreted, object-oriented,
high-level programming language with dynamic semantics.”

IN DECEMBER 1989

Guido van Rossum,

a Dutch computer scientist,

set himself a Christmas project.

Irked by shortcomings

in other programming languages,

he wanted to build his own.

Guido van Rossum

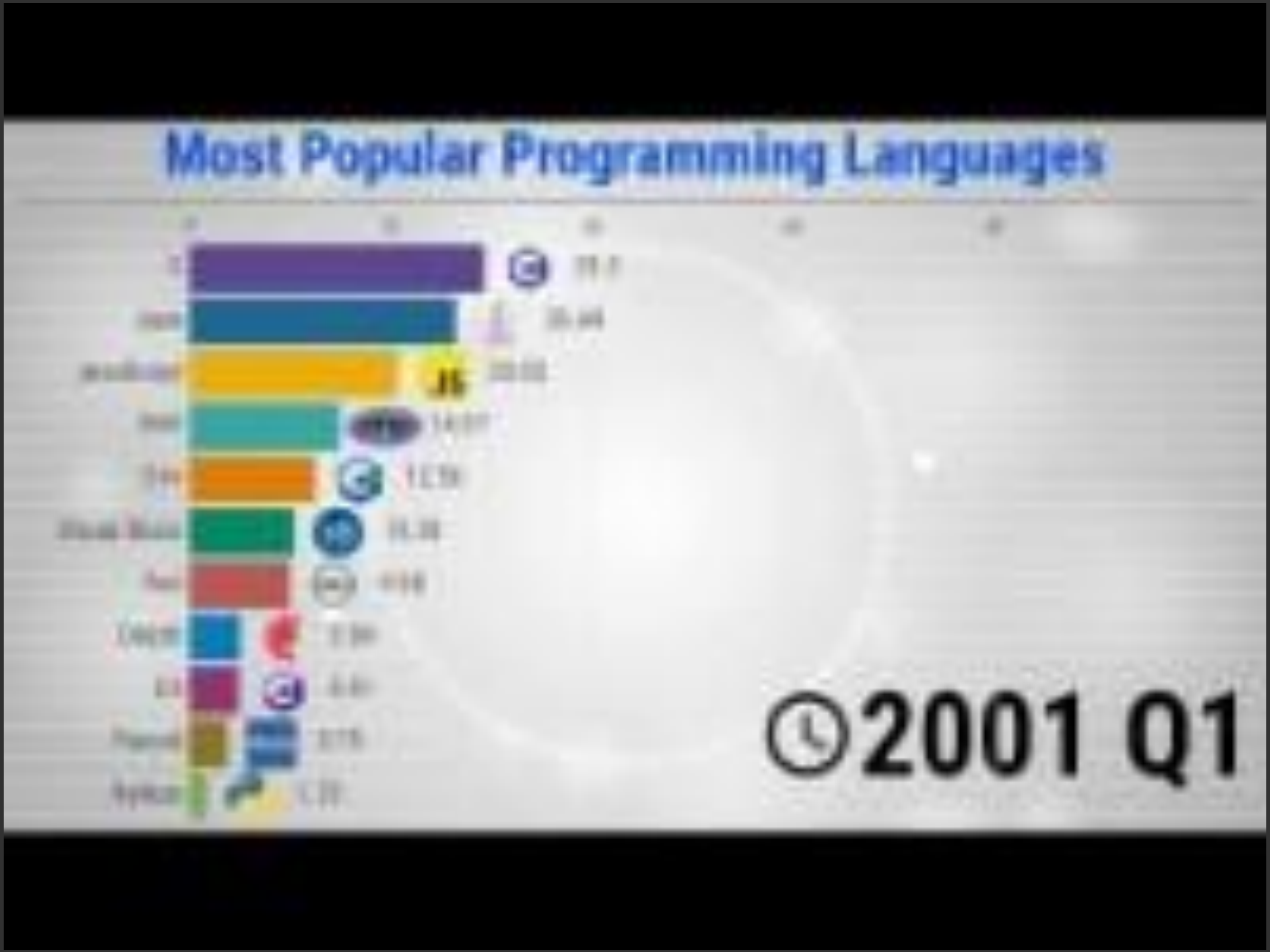
Python's creator and

emeritus BDFL (Benevolent Dictator For Life)

@gvanrossum

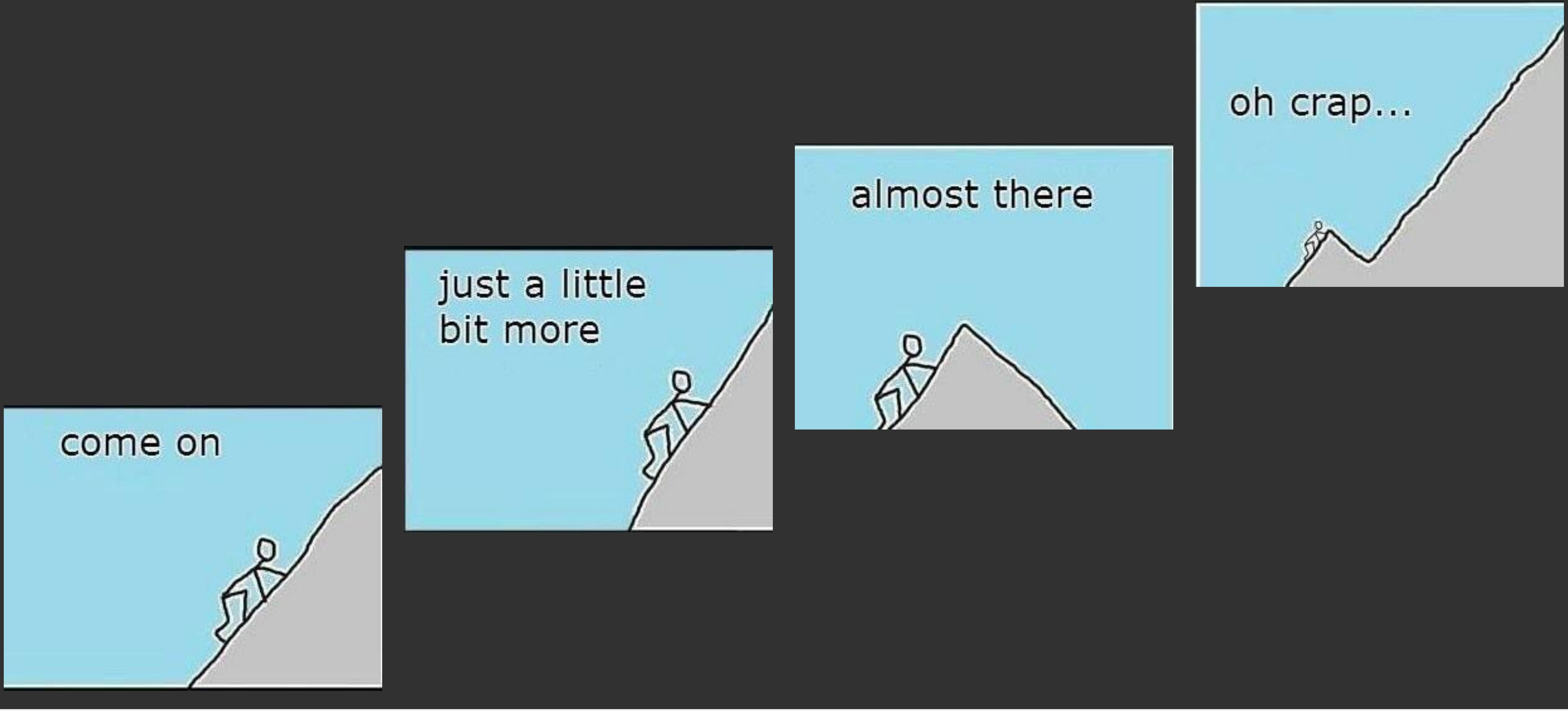
https://www.economist.com/science-and-technology/2018/07/21/python-has-brought-computer-programming-to-a-vast-new-audience

“short, unique and slightly mysterious” name

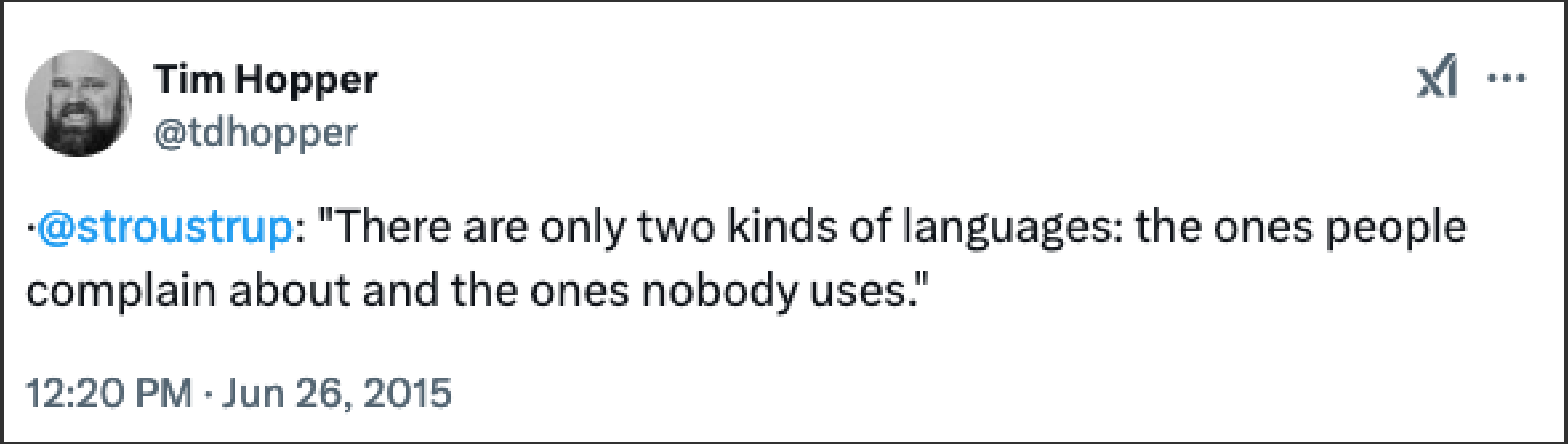


https://youtu.be/UNSoPa-XQN0?t=125

Trying to Learn Any Programming Language 100%



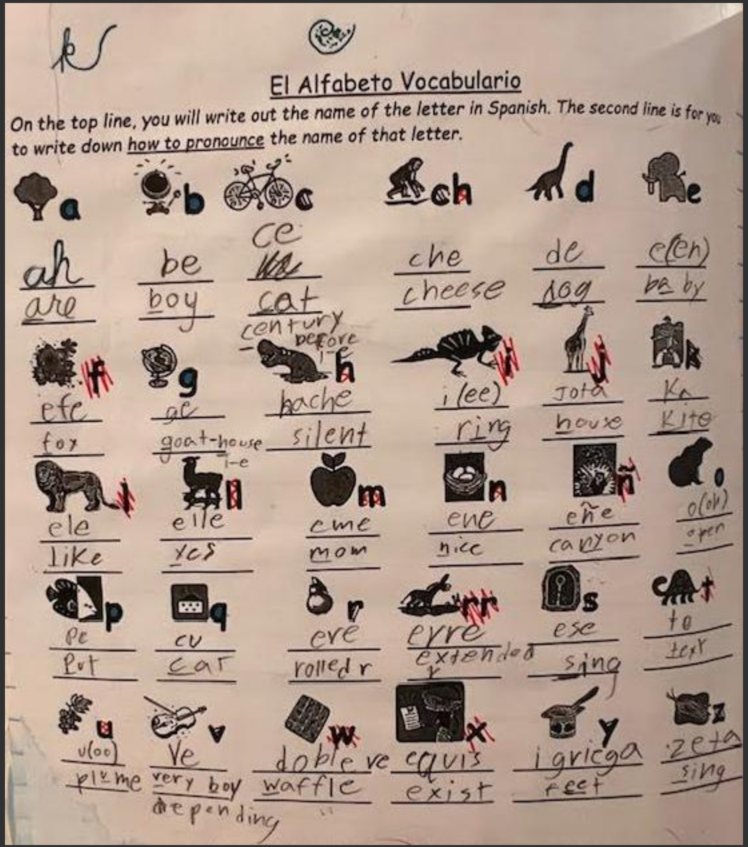
It’s not Perfect.



Tim Hopper
@tdhopper

·@stroustrup: "There are only two kinds of languages: the ones people complain about and the ones nobody uses."

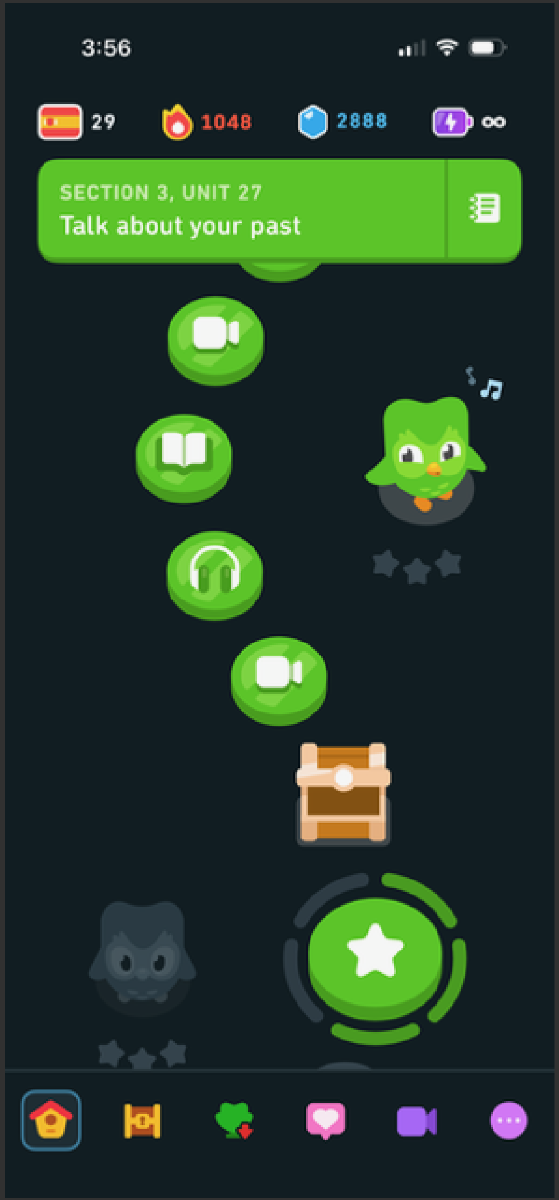
12:20 PM · Jun 26, 2015



El Alfabeto Vocabulario

On the top line, you will write out the name of the letter in Spanish. The second line is for you to write down how to pronounce the name of that letter.

a	b	c	ch	d	e
ah	be	ce ~~ka~~	che	de	e(en)
are	boy	cat century before i-e	cheese	dog	ba by
f	g	h	i	j	k
efe	ge	hache	i (ee)	Jota	Ka
fox	goat-house i-e	silent	ring	house	Kite
l	ll	m	n	ñ	o
ele	elle	eme	ene	eñe	o(oh)
like	yes	mom	nice	canyon	open
p	q	r	rr	s	t
pe	cu	ere	erre	ese	te
put	car	rolled r	extended r	sing	text
u	v	w	x	y	z
u(oo)	ve	doble ve	equis	i griega	zeta
plume	very boy depending	waffle	exist	feet	sing



3:56

29 1048 2888 ∞

SECTION 3, UNIT 27

Talk about your past

The Rise of Coding AI Agents

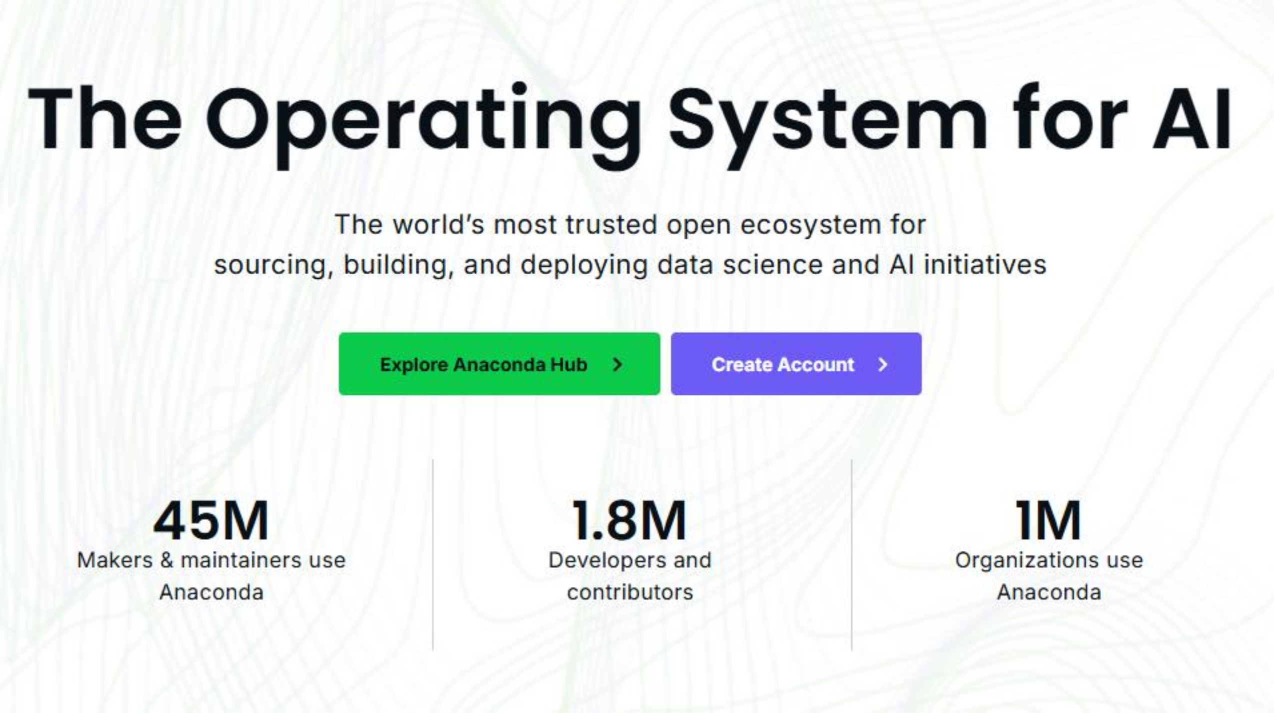
"I think the job of being paid money to type code into a computer will go the same way as punching punch cards [...] in six years time, I do not think anyone will be paid to just to do the thing where you type the code.

I think software engineering will still be an enormous career. I just think the software engineers won't be spending multiple hours of their day in a text editor typing out syntax."

– Simon Willison

https://open.substack.com/pub/simonw/p/llm-predictions-for-2026-shared-with

ANACONDA®



The Operating System for AI

The world's most trusted open ecosystem for
sourcing, building, and deploying data science and AI initiatives

45M
Makers & maintainers use Anaconda

1.8M
Developers and contributors

1M
Organizations use Anaconda

pandas

$y_i t = \beta' x_{it} + \mu_i + \epsilon_{it}$

Provides high-performance, easy-to-use data structures and data analysis tools

scikit learn

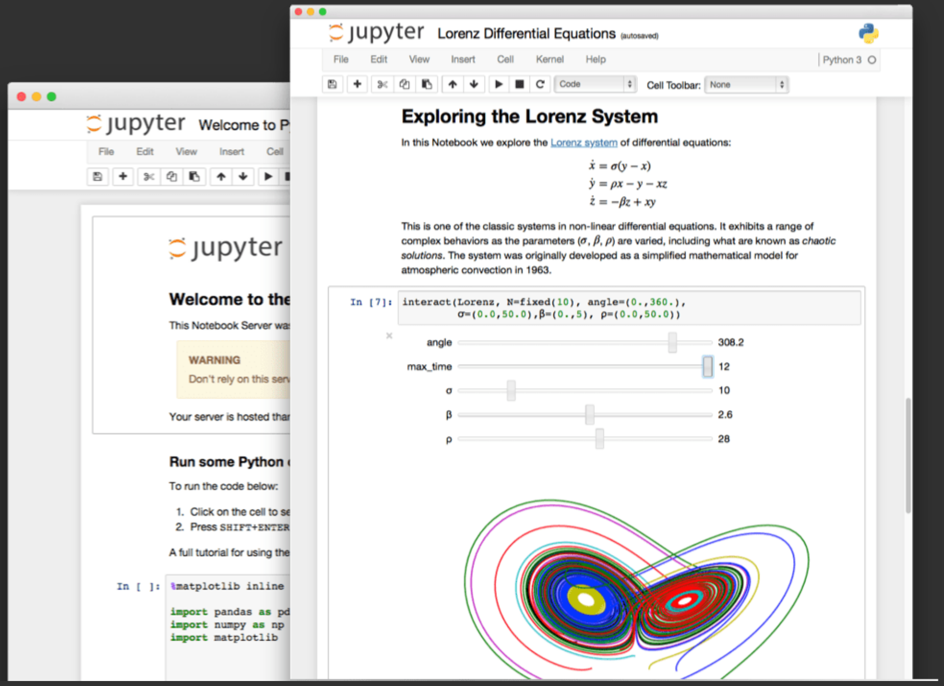
Simple and efficient tools for data mining and data analysis

seaborn

Provides a high-level interface for drawing attractive and informative statistical graphics.

Project Jupyter exists to develop open-source software, open-standards, and services for interactive computing across dozens of programming languages.

Julia, **Pyt**hon and **R**



Welcome to the

This Notebook Server wa

WARNING

Don't rely on this serv

Your server is hosted thar

Run some Python (

To run the code below:

1. Click on the cell to se
2. Press SHIFT+ENTER

A full tutorial for using the

```
In [ ]: %matplotlib inline

        import pandas as pd
        import numpy as np
        import matplotlib
```

Exploring the Lorenz System

In this Notebook we explore the Lorenz system of differential equations:

$$\dot{x} = \sigma(y - x)$$
$$\dot{y} = \rho x - y - xz$$
$$\dot{z} = -\beta z + xy$$

This is one of the classic systems in non-linear differential equations. It exhibits a range of complex behaviors as the parameters (σ, β, ρ) are varied, including what are known as *chaotic solutions*. The system was originally developed as a simplified mathematical model for atmospheric convection in 1963.

```
In [7]: interact(Lorenz, N=fixed(10), angle=(0.,360.),
                 σ=(0.0,50.0),β=(0.,5), ρ=(0.0,50.0))
```

angle 308.2

max_time 12

σ 10

β 2.6

ρ 28

The **Jupyter Notebook** is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text.

Language of choice

The Notebook has support for over 40 programming languages, including Python, R, Julia, and Scala.

Share notebooks

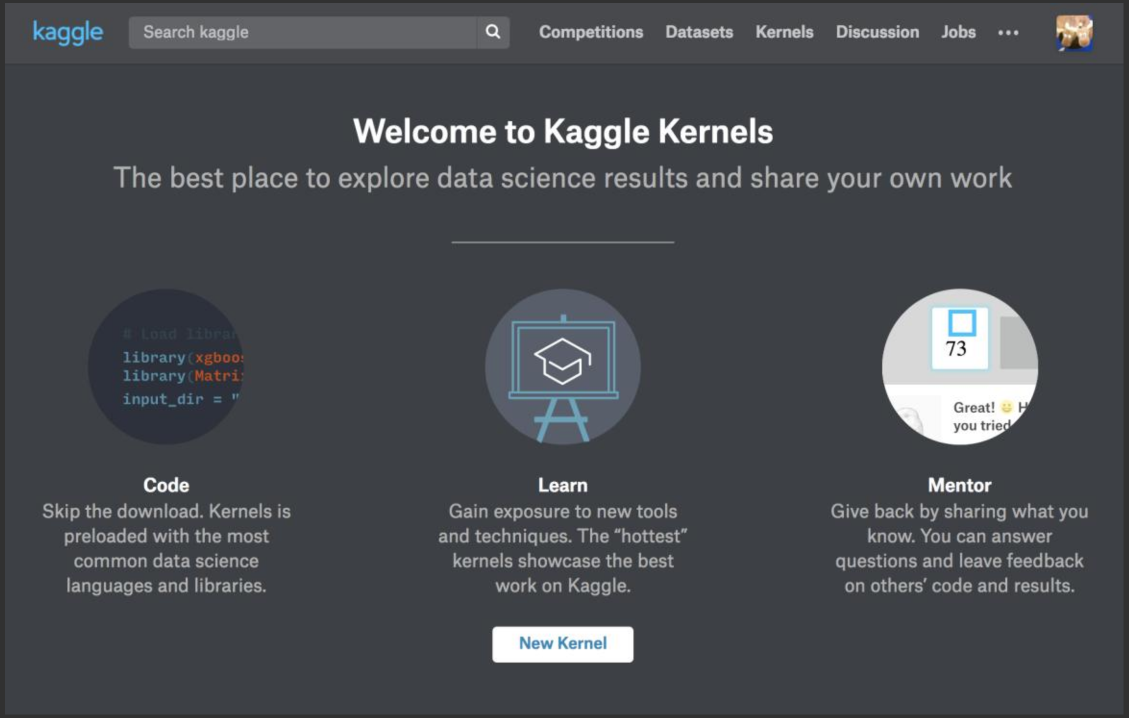
Notebooks can be shared with others using email, Dropbox, GitHub and the Jupyter Notebook Viewer.

Interactive output

Your code can produce rich, interactive output: HTML, images, videos, LaTeX, and custom MIME types.

Big data integration

Leverage big data tools, such as Apache Spark, from Python, R and Scala. Explore that same data with pandas, scikit-learn, ggplot2, TensorFlow.



Welcome to Kaggle Kernels

The best place to explore data science results and share your own work

Code
Skip the download. Kernels is preloaded with the most common data science languages and libraries.

Learn
Gain exposure to new tools and techniques. The "hottest" kernels showcase the best work on Kaggle.

Mentor
Give back by sharing what you know. You can answer questions and leave feedback on others' code and results.

colab

Amazon SageMaker

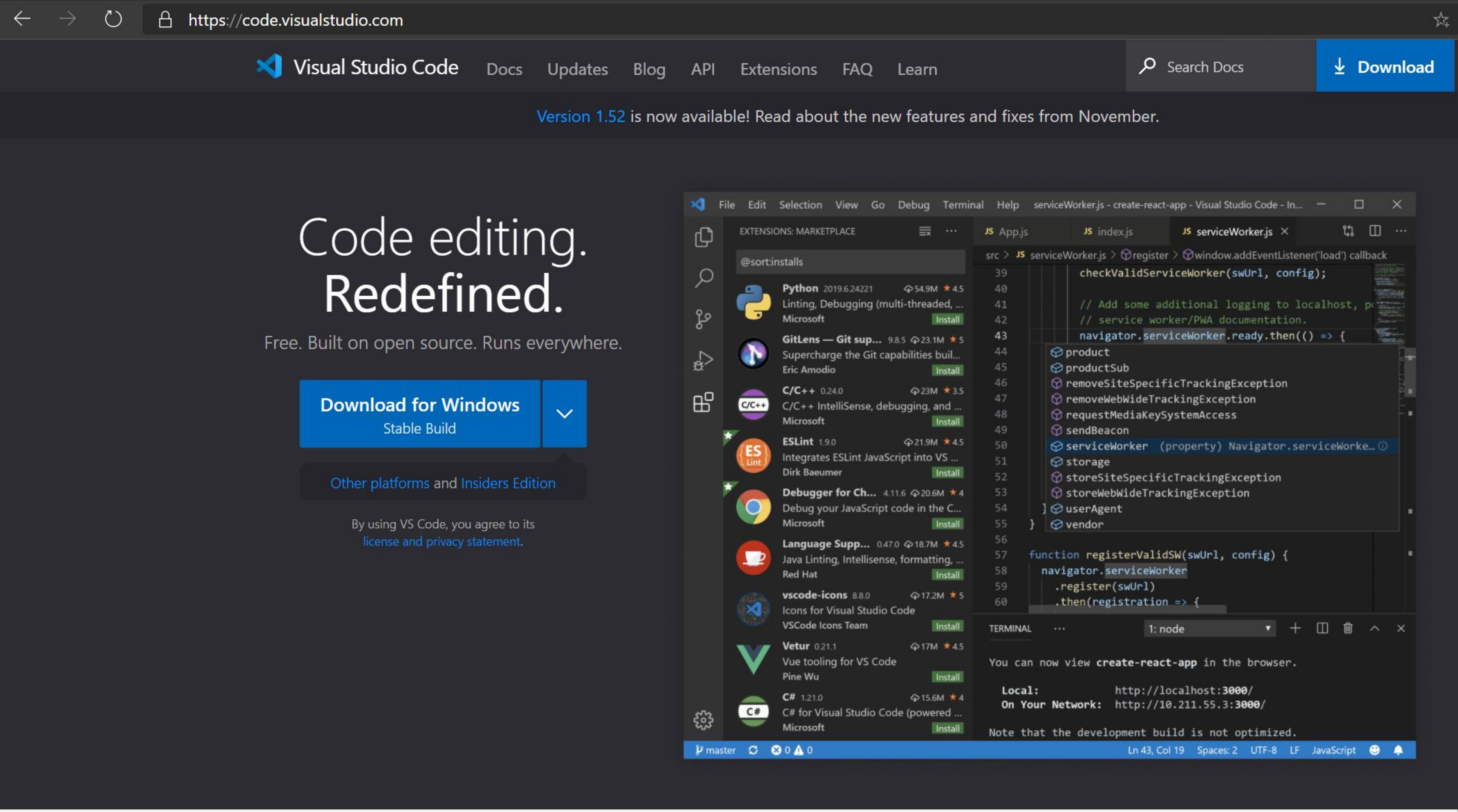
Machine learning for every data scientist and developer

A gallery of interesting Jupyter Notebooks

Jupyter Notebook

1. The **Jupyter Notebook App** (formerly IPython Notebook) is a server-client application that allows editing and running notebook documents via a web browser.
2. A notebook **kernel** is a “computational engine” that executes the code contained in a Notebook document.
3. The Jupyter Notebook App gets installed automatically when you install Anaconda.
4. The Jupyter Notebook App can be launched by clicking on the Jupyter Notebook icon installed by Anaconda in the start menu.

https://jupyter-notebook-beginner-guide.readthedocs.io



Version 1.52 is now available! Read about the new features and fixes from November.

Code editing. Redefined.

Free. Built on open source. Runs everywhere.

Other platforms and Insiders Edition

By using VS Code, you agree to its license and privacy statement.

File
Edit
Selection
View
Go
Debug
Terminal
Help
serviceWorker.js - create-react-app - Visual Studio Code - In...
EXTENSIONS: MARKETPLACE
@sort:installs
App.js
index.js
serviceWorker.js
src > serviceWorker.js > register > window.addEventListener('load') callback

Python 2019.6.24221 54.9M ★4.5
Linting, Debugging (multi-threaded, ...
Microsoft Install

GitLens — Git sup... 9.8.5 23.1M ★5
Supercharge the Git capabilities buil...
Eric Amodio Install

C/C++ 0.24.0 23M ★3.5
C/C++ IntelliSense, debugging, and ...
Microsoft Install

ESLint 1.9.0 21.9M ★4.5
Integrates ESLint JavaScript into VS ...
Dirk Baeumer Install

Debugger for Ch... 4.11.6 20.6M ★4
Debug your JavaScript code in the C...
Microsoft Install

Language Supp... 0.47.0 18.7M ★4.5
Java Linting, Intellisense, formatting, ...
Red Hat Install

vscode-icons 8.8.0 17.2M ★5
Icons for Visual Studio Code
VSCode Icons Team Install

Vetur 0.21.1 17M ★4.5
Vue tooling for VS Code
Pine Wu Install

C# 1.21.0 15.6M ★4
C# for Visual Studio Code (powered ...
Microsoft Install

```
        checkValidServiceWorker(swUrl, config);

        // Add some additional logging to localhost, p
        // service worker/PWA documentation.
        navigator.serviceWorker.ready.then(() => {
```

product
productSub
removeSiteSpecificTrackingException
removeWebWideTrackingException
requestMediaKeySystemAccess
sendBeacon
serviceWorker (property) Navigator.serviceWorke...
storage
storeSiteSpecificTrackingException
storeWebWideTrackingException
userAgent
vendor

```
      ]
    }

function registerValidSW(swUrl, config) {
  navigator.serviceWorker
    .register(swUrl)
    .then(registration => {
```

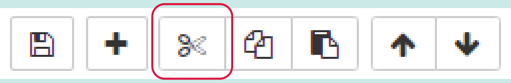
```
You can now view create-react-app in the browser.

  Local:            http://localhost:3000/
  On Your Network:  http://10.211.55.3:3000/

Note that the development build is not optimized.
```

A quick intro to Jupyter Notebook/Lab

Closing the Jupyter Notebook App

1. Closing the browser (or the tab) **will not close** the Jupyter Notebook App. To completely shut it down you need to **close the associated terminal**.
2. To shut it down, go to the associated notebook and click on menu **File → Close** and **Halt**.
    1. Alternatively, the Notebook Dashboard has a tab named *Running* that shows all the running notebooks (i.e. kernels) and allows shutting them down (by clicking on a *Shutdown* button).

A Few Useful Commands

Task	Command/Short-cut
Run selected cell	Ctrl + Enter
Run selected cell and insert a new cell below	Alt + Enter
Run all cells	Menu: *Cell -> Run All*.
Restart the kernel	Menu: *Kernel -> Restart*
Delete a cell	

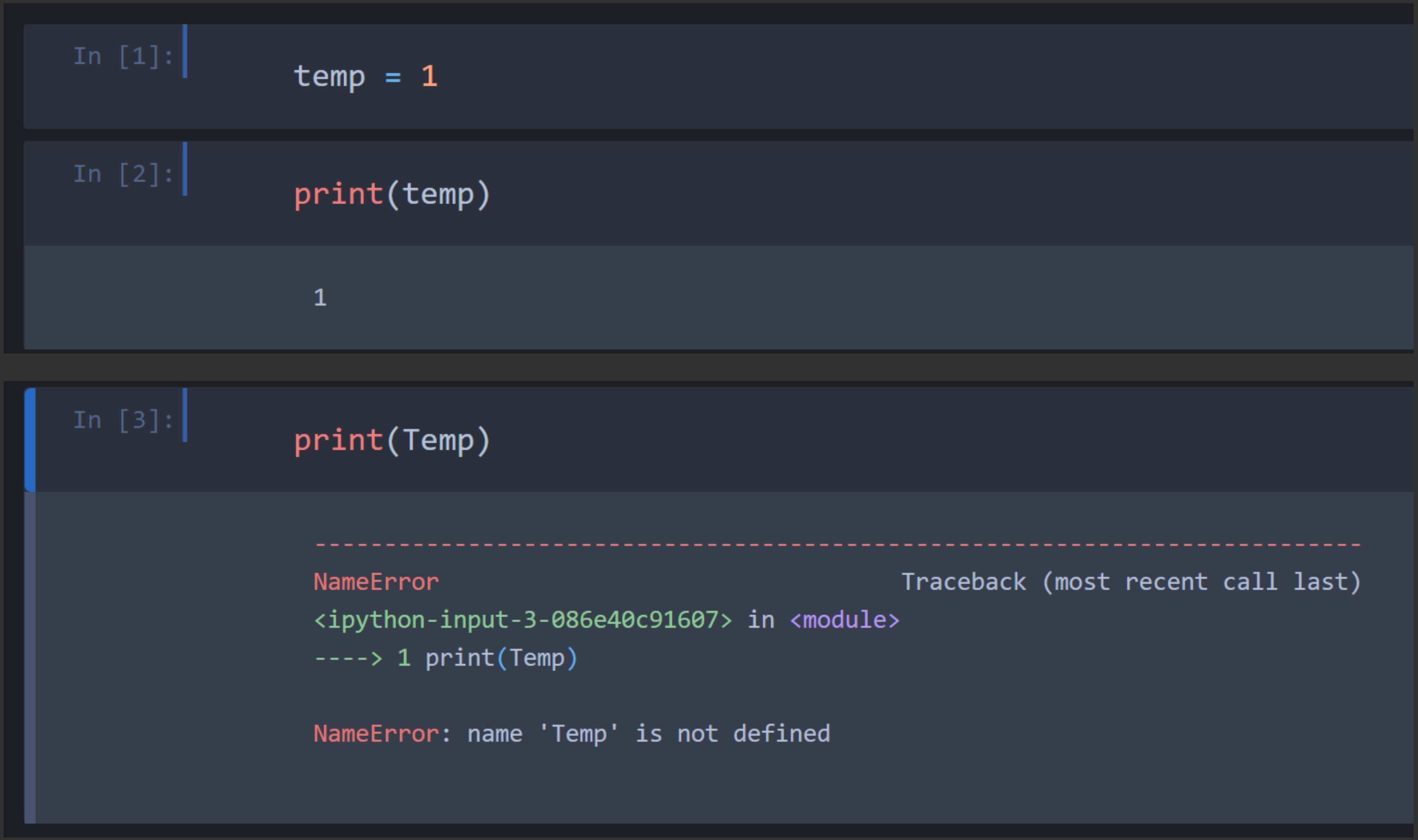
How much of coding experience do you have?

- ○ None / Less than 1 year
- ○ 1-5 years
- ○ 5+ years

Python Intro Tutorial

`00_intro_python_tutorial.ipynb`

Case Sensitive



```
In [1]:  temp = 1
```

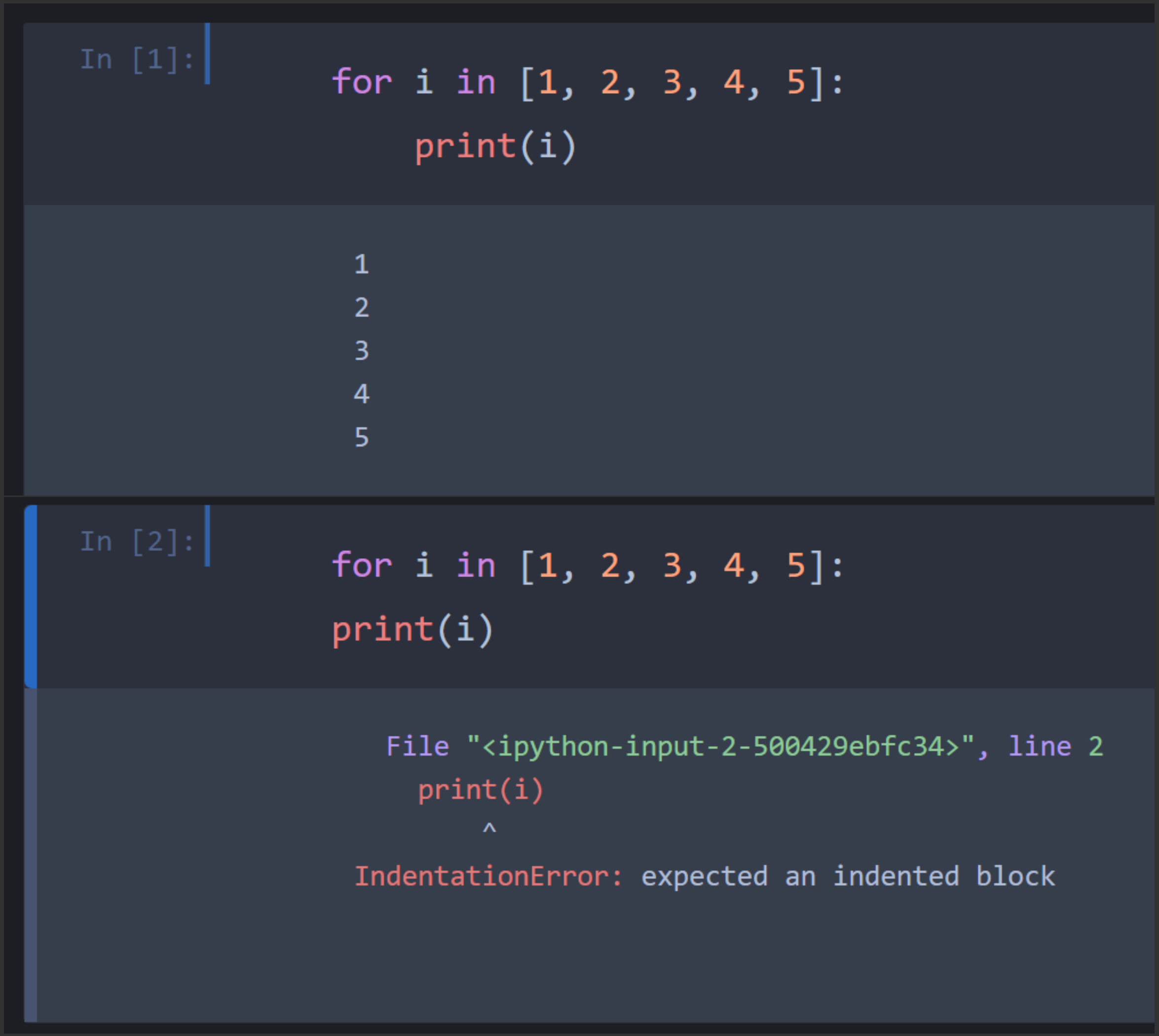
```
In [2]:  print(temp)
```

```
1
```

```
In [3]:  print(Temp)
```

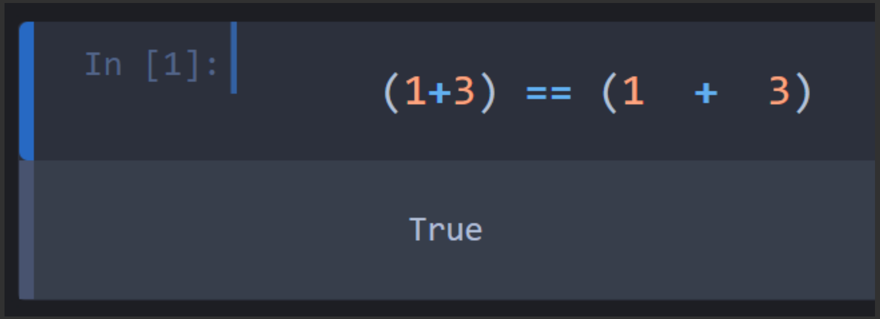
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-3-086e40c91607> in <module>
----> 1 print(Temp)

NameError: name 'Temp' is not defined
```

Whitespace / Indentation



```
In [1]:  for i in [1, 2, 3, 4, 5]:
             print(i)

         1
         2
         3
         4
         5
```

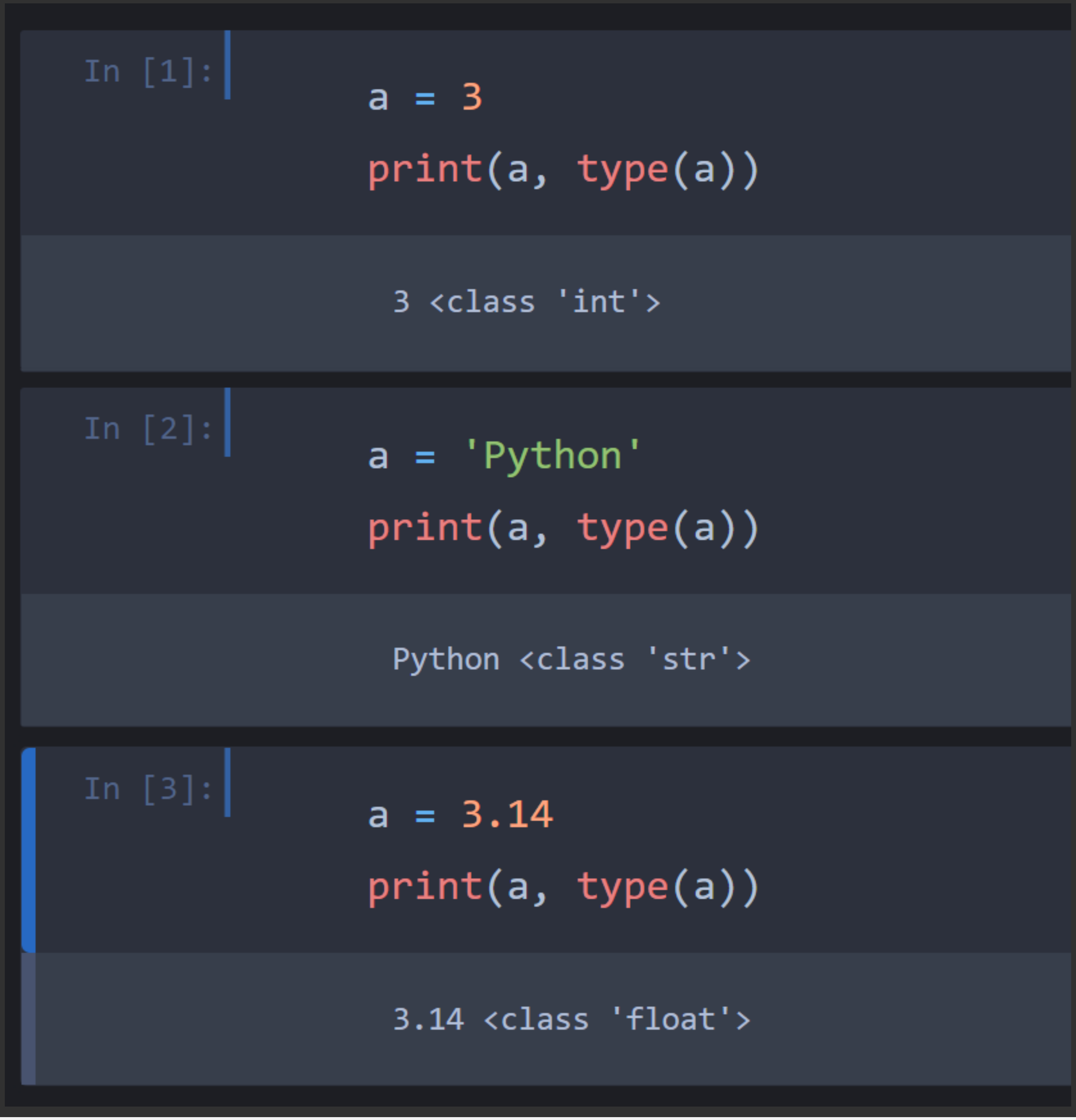
```
In [2]:  for i in [1, 2, 3, 4, 5]:
         print(i)

           File "<ipython-input-2-500429ebfc34>", line 2
             print(i)
                 ^
         IndentationError: expected an indented block
```



```
In [1]:  (1+3) == (1  +  3)

         True
```

Whitespaces *within* lines do not matter.

Dynamic Typing



```
In [1]:
a = 3
print(a, type(a))
```

```
3 <class 'int'>
```

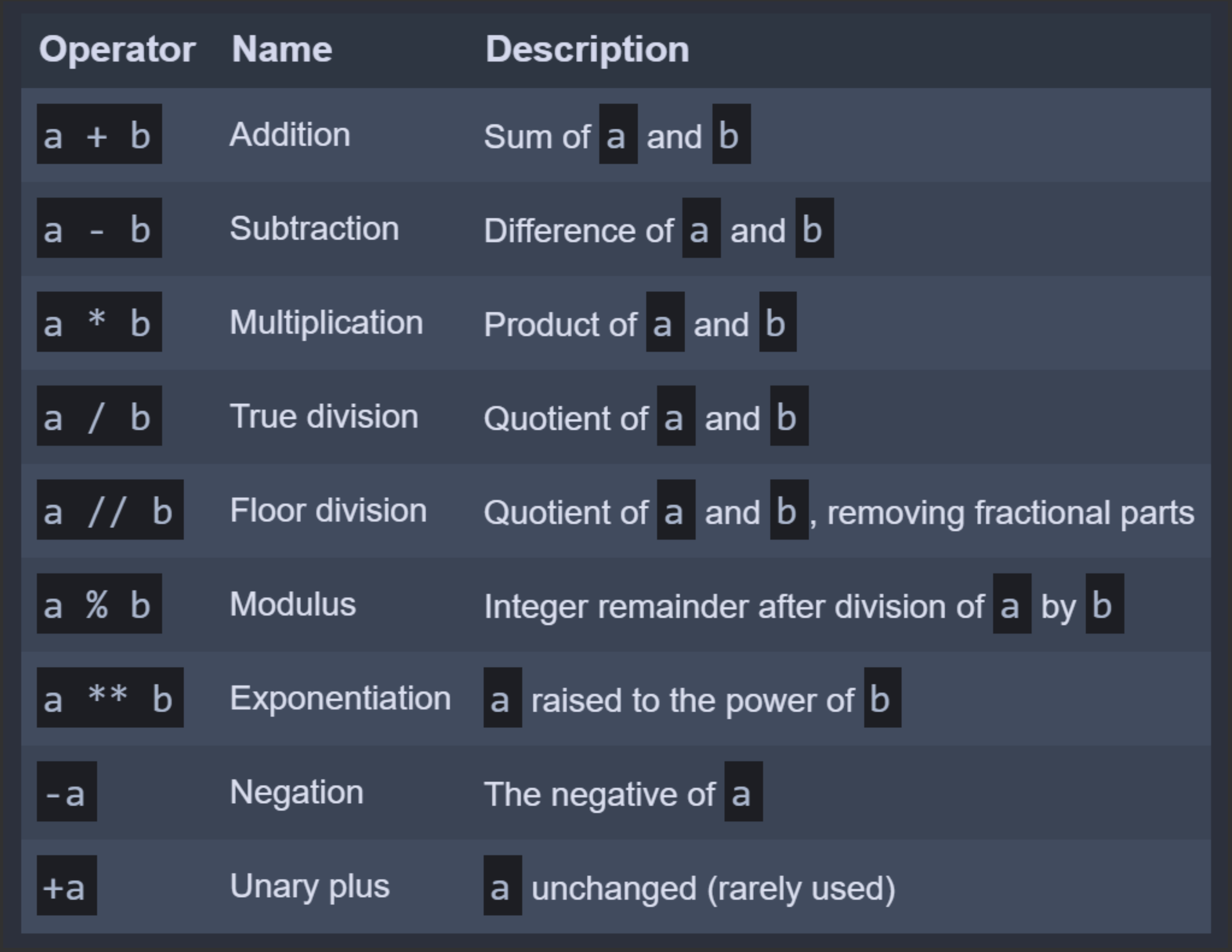
```
In [2]:
a = 'Python'
print(a, type(a))
```

```
Python <class 'str'>
```

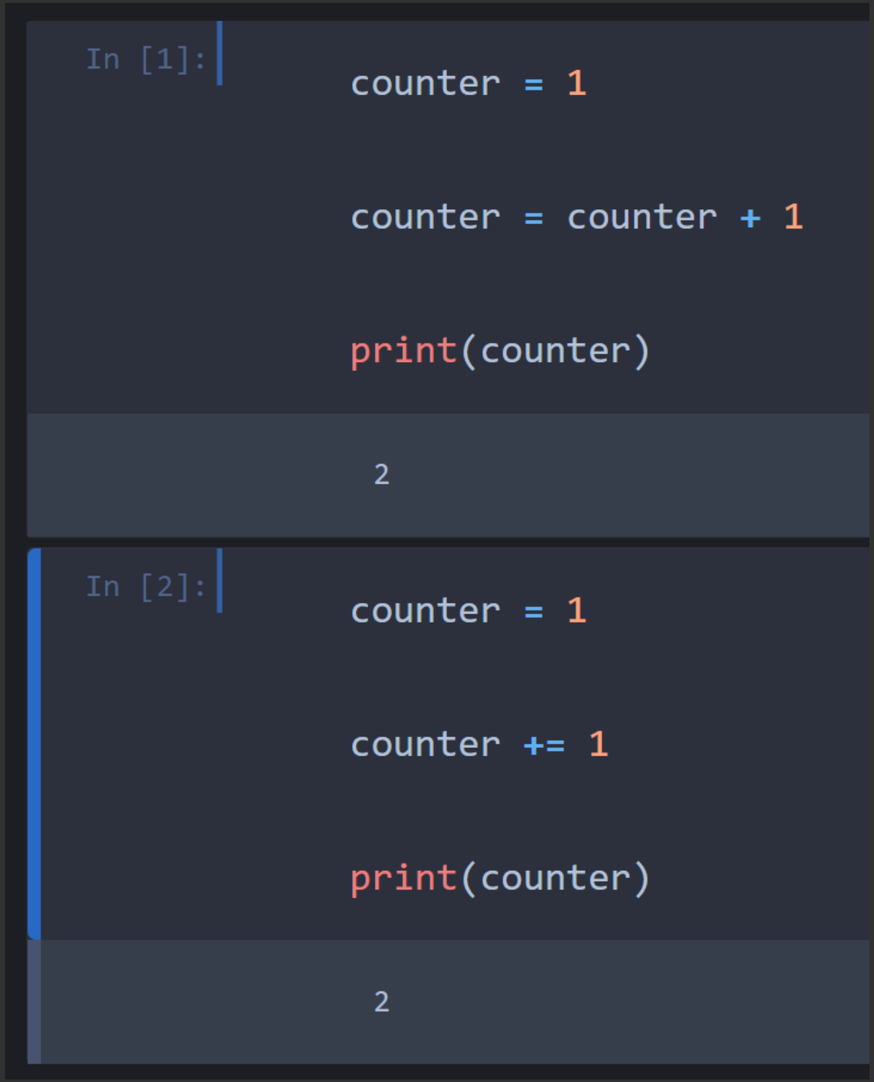
```
In [3]:
a = 3.14
print(a, type(a))
```

```
3.14 <class 'float'>
```

Arithmetic Operations



Operator	Name	Description
`a + b`	Addition	Sum of `a` and `b`
`a - b`	Subtraction	Difference of `a` and `b`
`a * b`	Multiplication	Product of `a` and `b`
`a / b`	True division	Quotient of `a` and `b`
`a // b`	Floor division	Quotient of `a` and `b`, removing fractional parts
`a % b`	Modulus	Integer remainder after division of `a` by `b`
`a ** b`	Exponentiation	`a` raised to the power of `b`
`-a`	Negation	The negative of `a`
`+a`	Unary plus	`a` unchanged (rarely used)



```
In [1]: counter = 1

        counter = counter + 1

        print(counter)
```

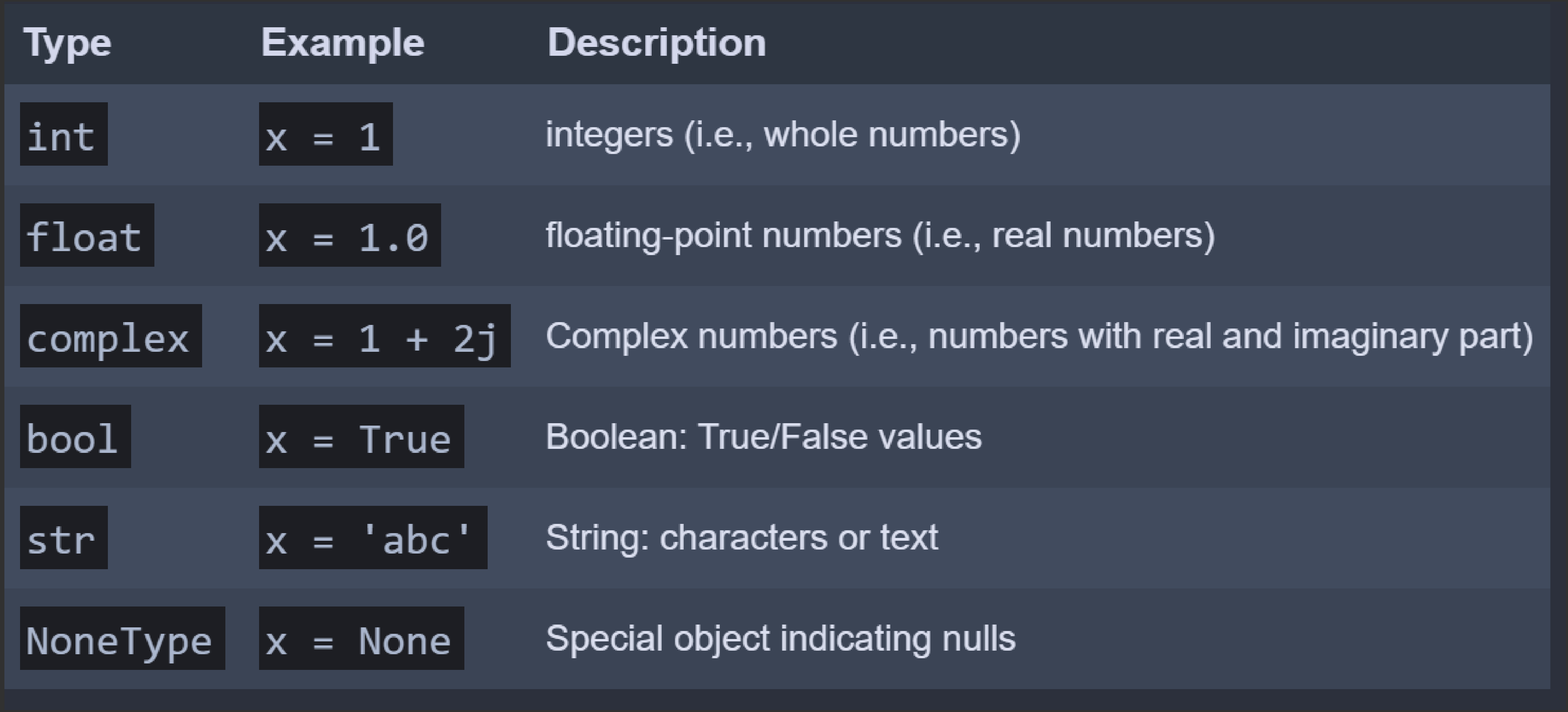
```
2
```

```
In [2]: counter = 1

        counter += 1

        print(counter)
```

```
2
```

http://nbviewer.jupyter.org/github/jakevdp/WhirlwindTourOfPython/blob/master/04-Semantics-Operators.ipynb

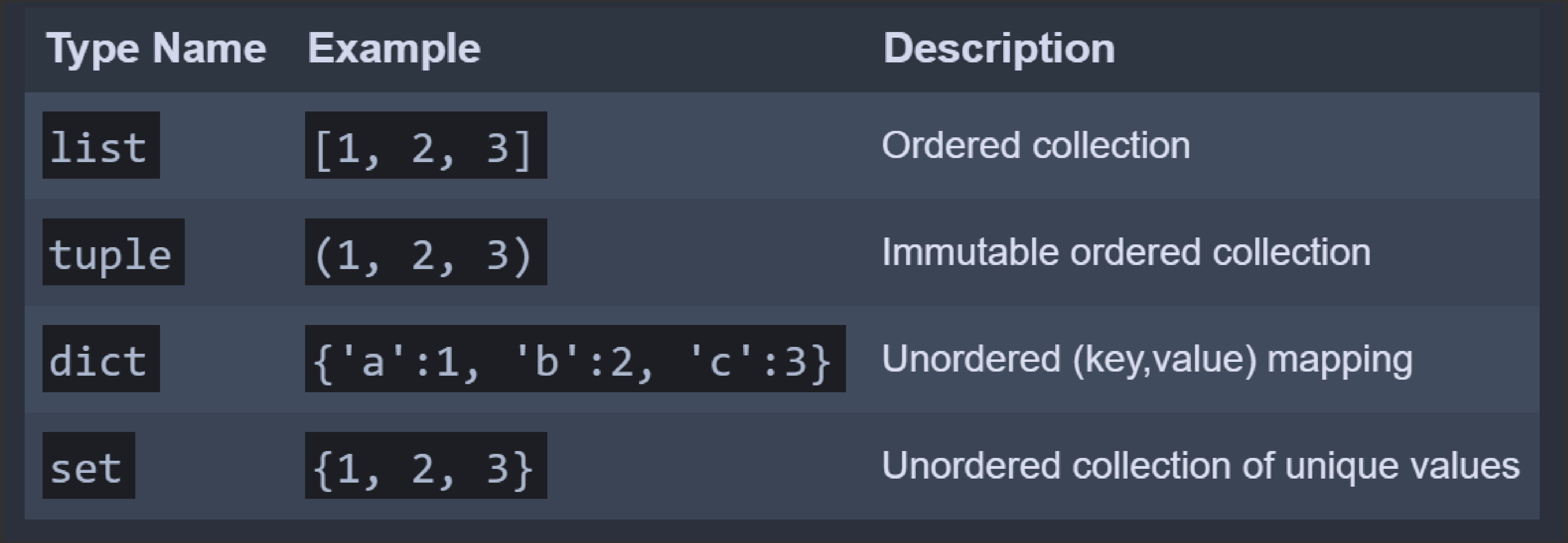
Scalar Types



Type	Example	Description
`int`	`x = 1`	integers (i.e., whole numbers)
`float`	`x = 1.0`	floating-point numbers (i.e., real numbers)
`complex`	`x = 1 + 2j`	Complex numbers (i.e., numbers with real and imaginary part)
`bool`	`x = True`	Boolean: True/False values
`str`	`x = 'abc'`	String: characters or text
`NoneType`	`x = None`	Special object indicating nulls

http://nbviewer.jupyter.org/github/jakevdp/WhirlwindTourOfPython/blob/master/05-Built-in-Scalar-Types.ipynb

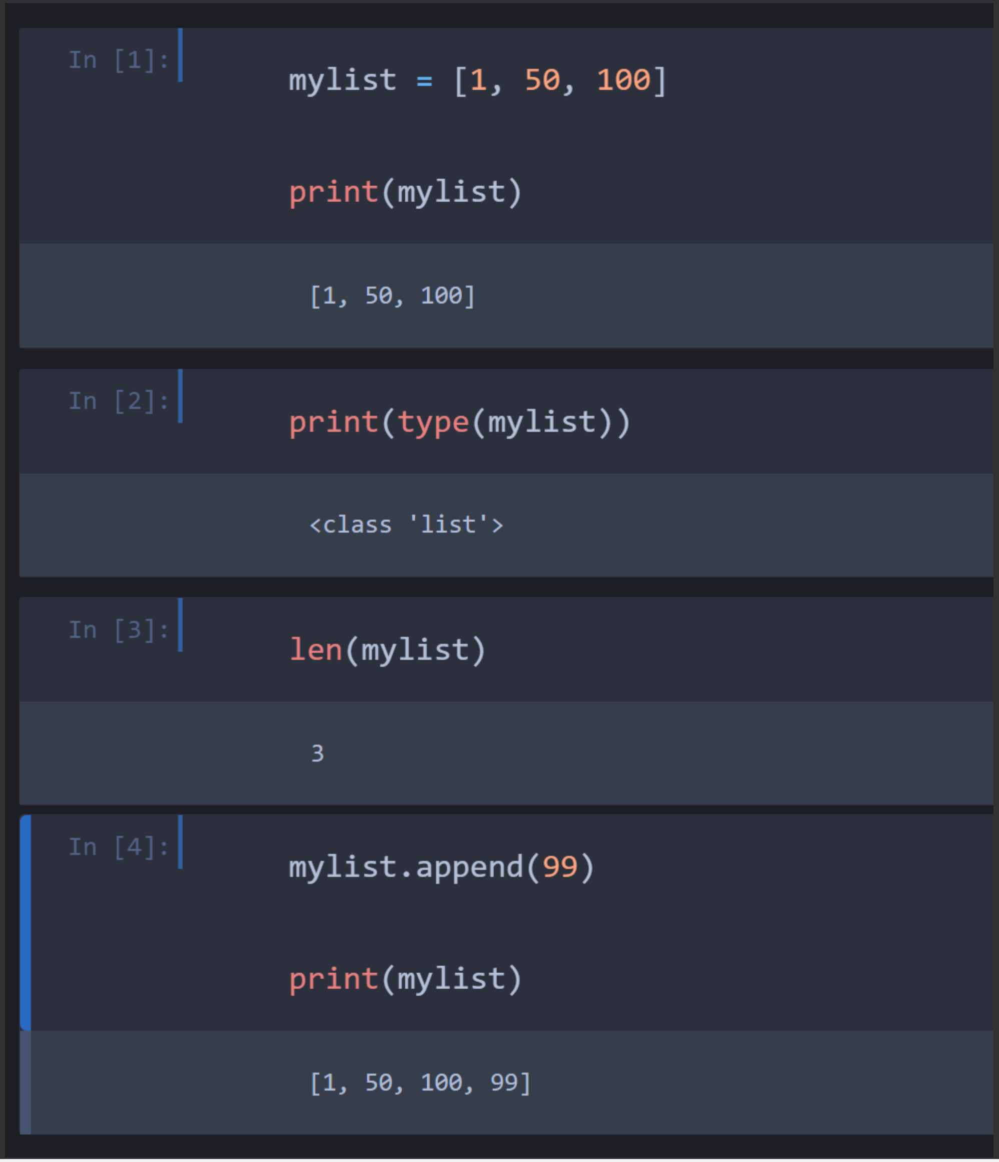
Data Structures



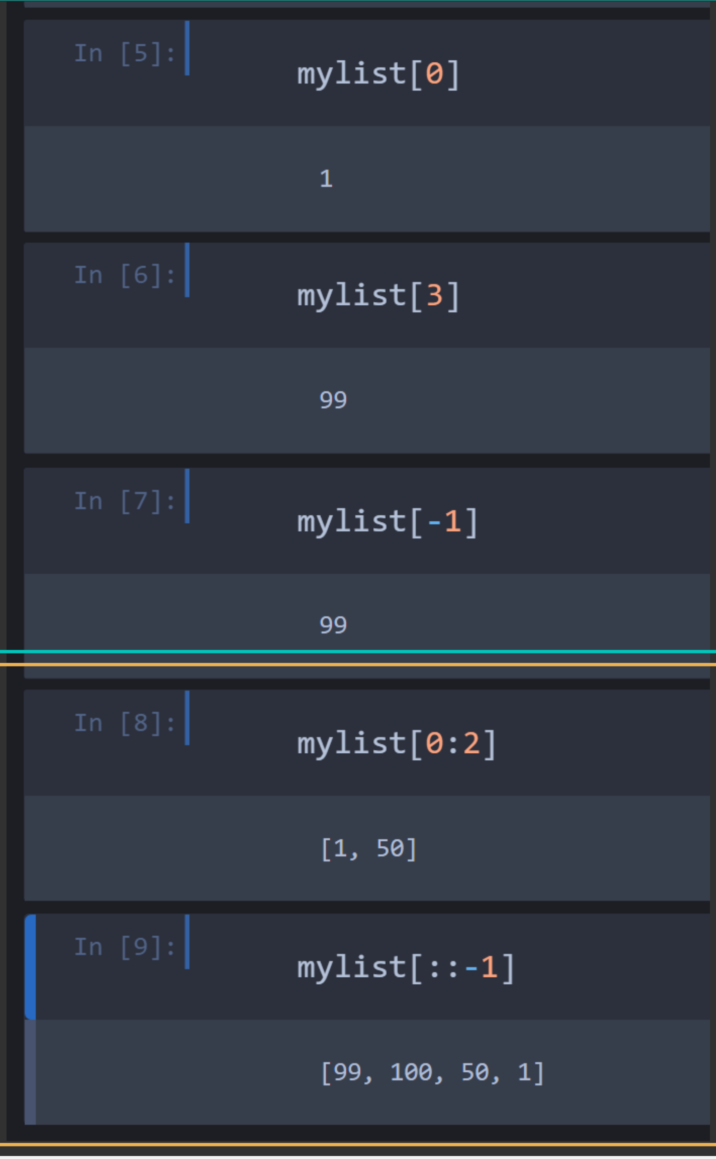
Type Name	Example	Description
`list`	`[1, 2, 3]`	Ordered collection
`tuple`	`(1, 2, 3)`	Immutable ordered collection
`dict`	`{'a':1, 'b':2, 'c':3}`	Unordered (key,value) mapping
`set`	`{1, 2, 3}`	Unordered collection of unique values

http://nbviewer.jupyter.org/github/jakevdp/WhirlwindTourOfPython/blob/master/06-Built-in-Data-Structures.ipynb

Indexing and Slicing



```
In [1]:  mylist = [1, 50, 100]

         print(mylist)

         [1, 50, 100]

In [2]:  print(type(mylist))

         <class 'list'>

In [3]:  len(mylist)

         3

In [4]:  mylist.append(99)

         print(mylist)

         [1, 50, 100, 99]
```



Indexing

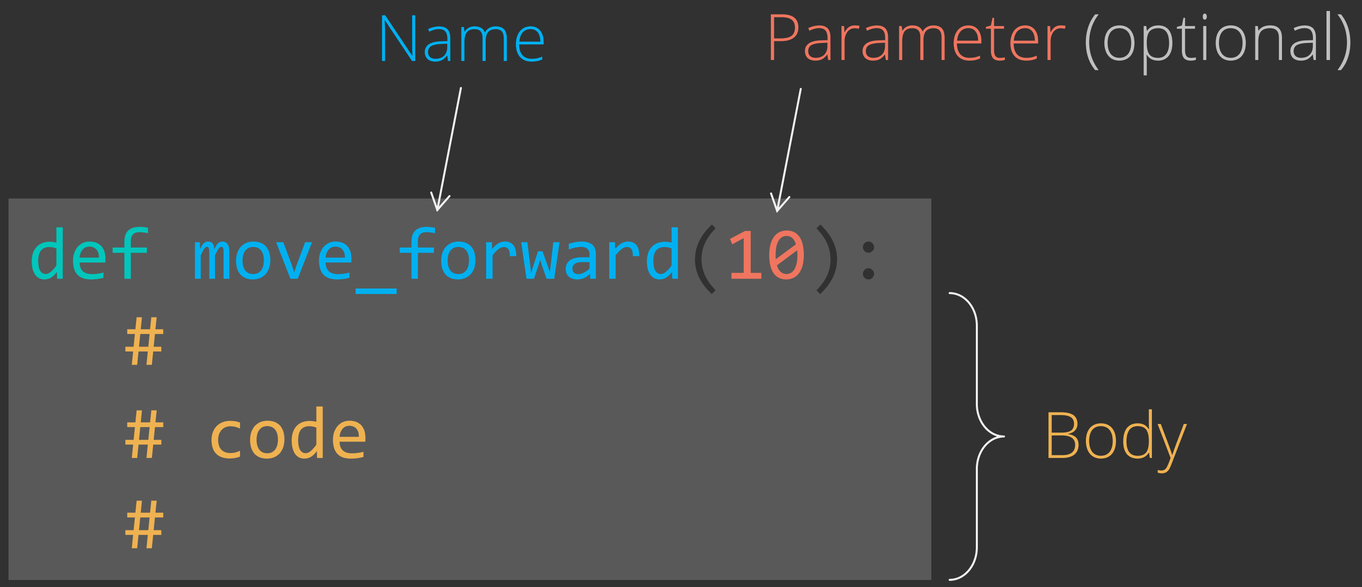
```
In [5]:  mylist[0]

         1

In [6]:  mylist[3]

         99

In [7]:  mylist[-1]

         99
```

Slicing

```
In [8]:  mylist[0:2]

         [1, 50]

In [9]:  mylist[::-1]

         [99, 100, 50, 1]
```

Anatomy of a Python Function

- Functions ≈ Actions, verbs, commands
- Parameters ≈ adverbs



Name

Parameter (optional)

```
def move_forward(10):
    #
    # code
    #
```

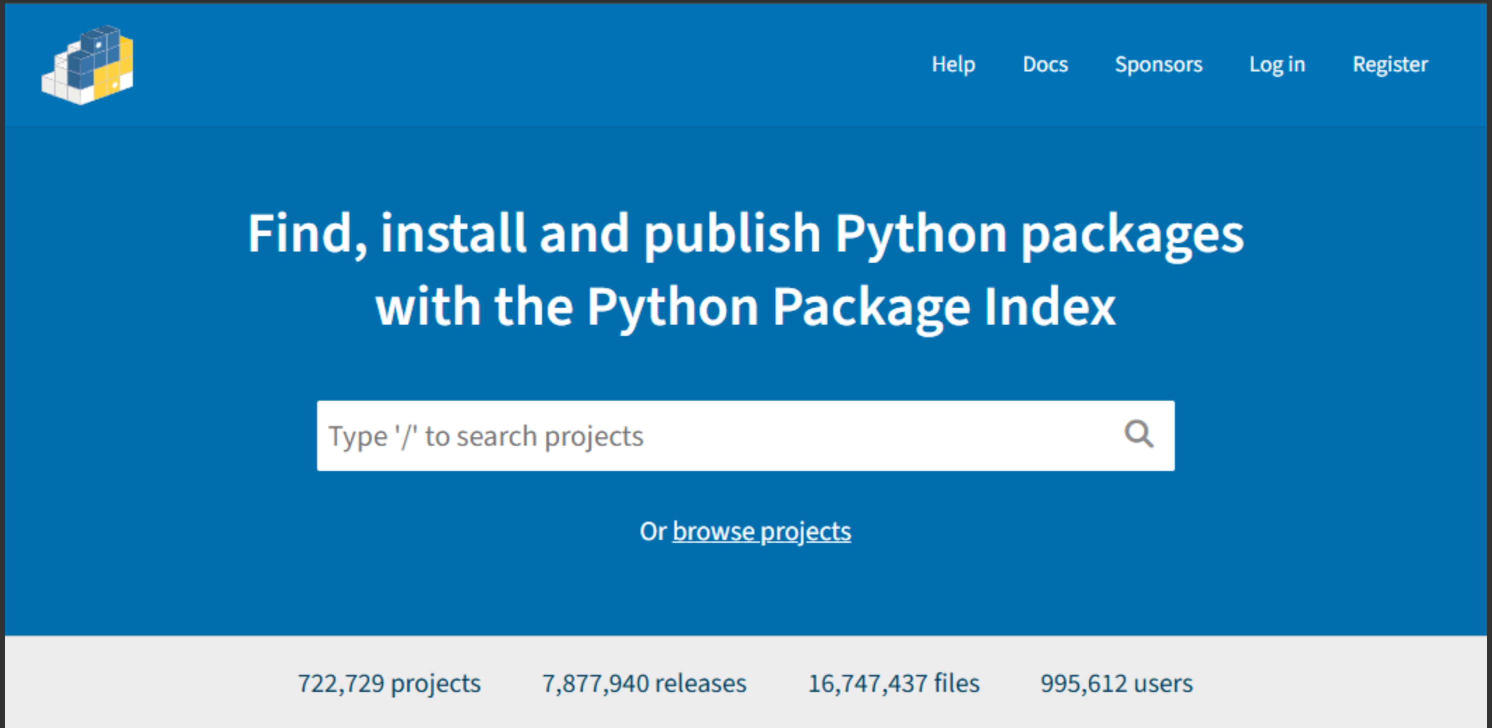
Body

Functions — Introduction to Programming with Python (python-intro.readthedocs.io)

Python Packages

1. **Installing** a package on your computer:
   1. Go to Anaconda command prompt (search for it on your computer using the search bar).
   2. Type ‘pip install <package-name>’ or ‘conda install <package-name>’ to install a specific package.
2. **Importing** a package into your program:
   1. Explicit module import: e.g., `import math`
   2. Explicit module import by alias: e.g., `import numpy as np`
   3. Explicit import of module contents: e.g., `from math import cos, pi`

“The Cheese Shop”



Find, install and publish Python packages with the Python Package Index

Or browse projects

722,729 projects · 7,877,940 releases · 16,747,437 files · 995,612 users

Jan 2025	598,203 projects	6,453,837 releases	12,986,350 files	889,931 users
Jan 2024	505,211 projects	5,245,624 releases	9,992,706 files	773,013 users
Jan 2023	426,421 projects	4,064,431 releases	7,333,845 files	655,788 users
Jan 2022	281,813 projects	2,297,688 releases	3,708,457 files	476,044 users