Python for Machine Learning

Dovan Rai

Outline

Python Universe

Python Basics

Data Structures

Numpy

Pandas

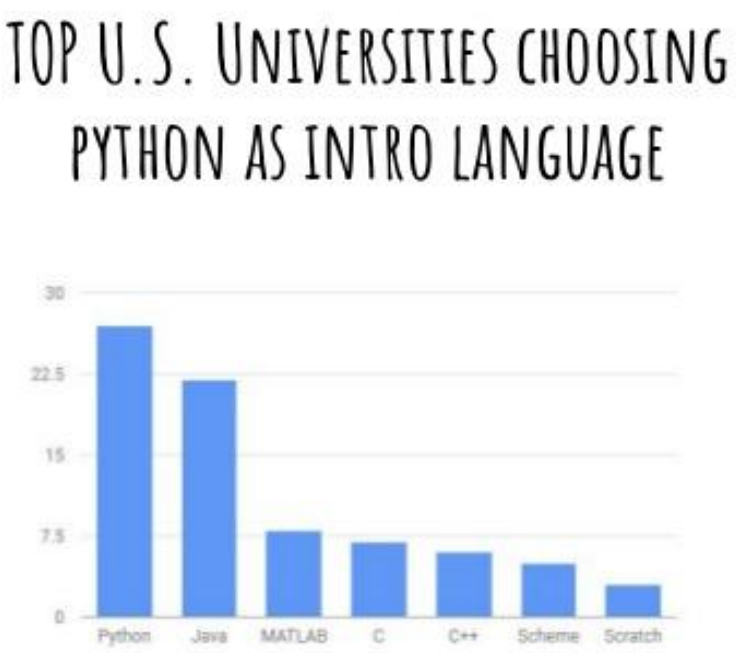
Python is not scary !

Monty Python

Funny ?

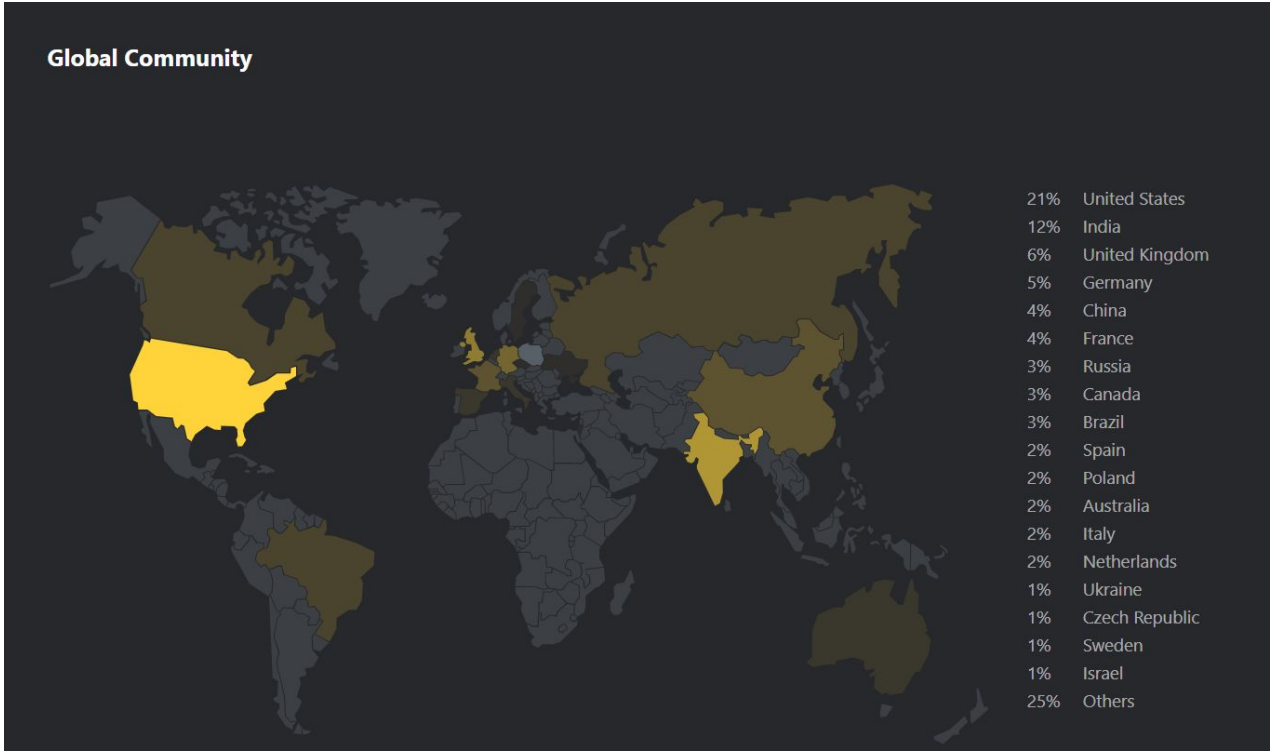
Python is accessible

Because it is designed to be.



Resource - http://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-u-s-universities/fulltext

It has a huge support community.



Python Developers Survey 2018

Python for Machine Learning

Python and Machine Learning not ‘one-to-one’ relationship.

It is rather a many-to-one and one-to-many relationship

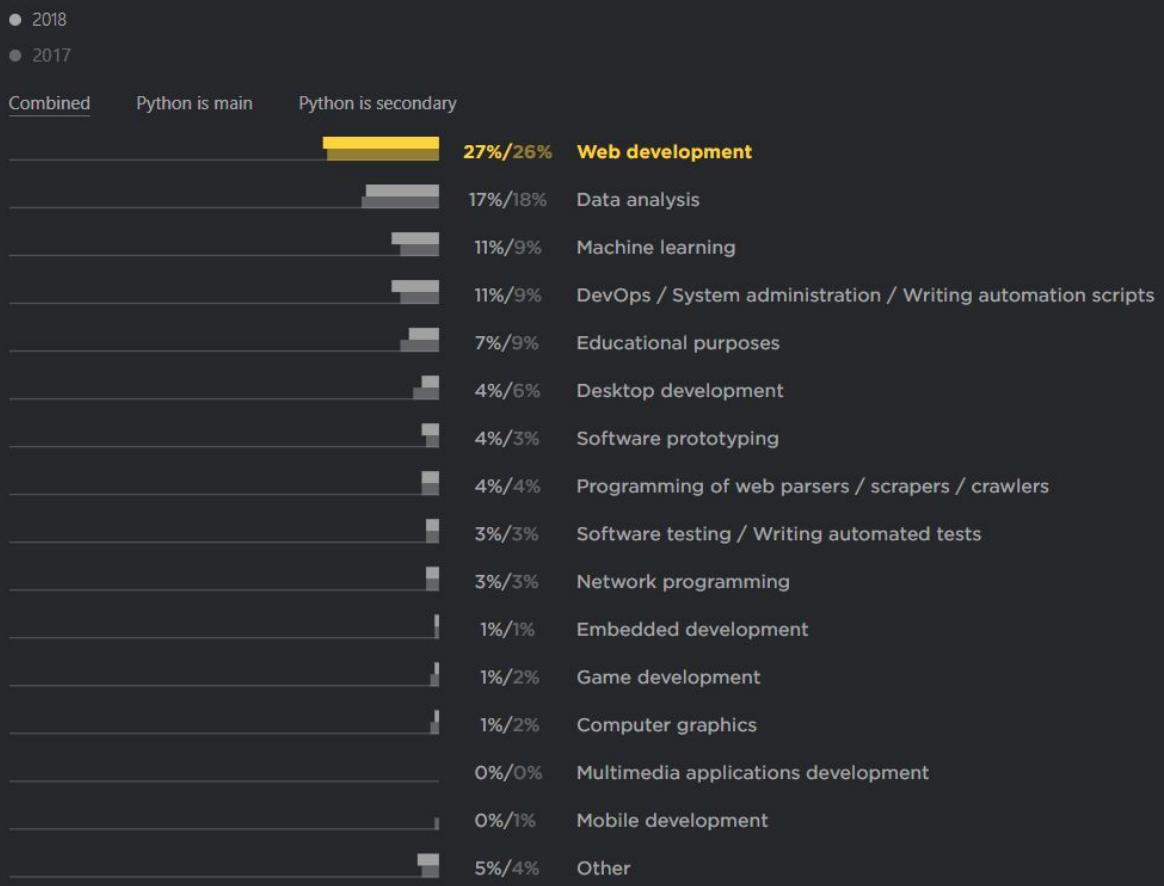
There are various languages for Machine Learning

MATLAB, C++, R, Java, Julia

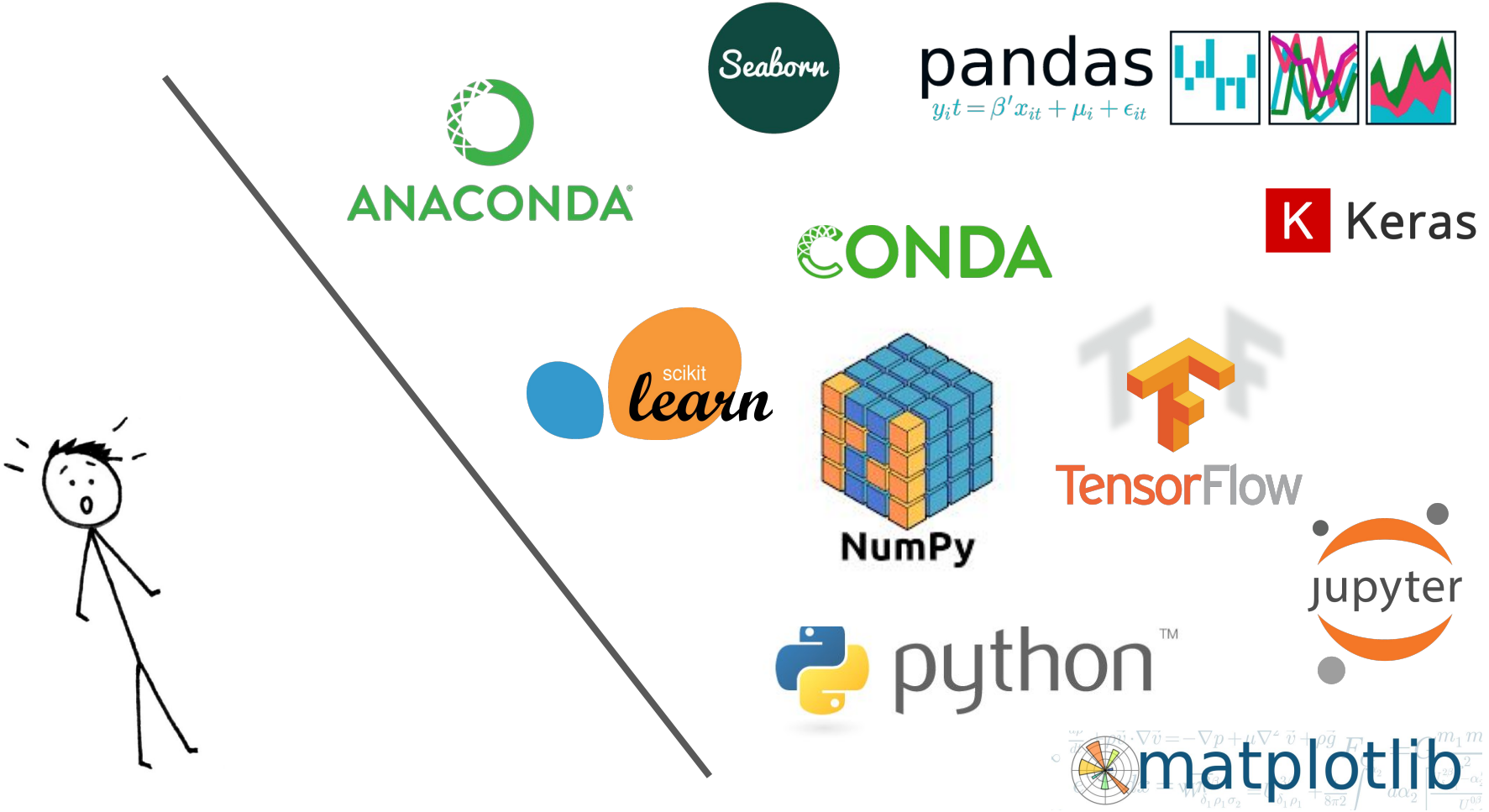
There are various things Python does besides Machine Learning

Scripting and automation, network programming, web development, game programming, desktop programming

What do you use Python for the most? (single answer)



Python for Machine Learning


Seaborn
pandas
$y_i t = \beta' x_{it} + \mu_i + \epsilon_{it}$
ANACONDA
CONDA
K Keras
scikit learn
NumPy
TensorFlow
jupyter
python™
matplotlib

Let’s sort it out !

Libraries/
Packages

Platform/ Environment

Libraries/ Packages

pandas

$y_it = \beta' x_{it} + \mu_i + \epsilon_{it}$

NumPy

scikit learn

ANACONDA

jupyter

Platform/ Environment

Libraries/ Packages

pandas

$y_it = \beta' x_{it} + \mu_i + \epsilon_{it}$

NumPy

scikit learn

Platform/ Environment

ANACONDA

jupyter

Libraries/ Packages

pandas

$y_it = \beta' x_{it} + \mu_i + \epsilon_{it}$

NumPy

scikit learn

my_notebook.ipnyb

Jupyter Notebook

[Python Code]

Files (.txt; .csv; .xlsx; ...)

import/ export

Platform/ Environment

ANACONDA

jupyter

.py vs .ipynb

.py is the standard python file extension that the python interpreter can read directly.

.ipynb stands for “IPython notebook” which was that Jupyter was originally called before supporting other language backends like R, Julia.

It cannot be directly be read by the python interpreter but rather requires the jupyter environment to run

Python and Jupyter Notebook is also many-to-many relationship.

There are various environments that support Python

Google Colab, Apache Spark , Apache Zeppelin

Jupyter Notebook does multiple things besides Python

Jupyter notebook can support many languages
(Julia, Python, R)

Libraries/ Packages

pandas

$y_it = \beta' x_{it} + \mu_i + \epsilon_{it}$

NumPy

scikit learn

my_notebook.ipnyb

Jupyter Notebook

[Python Code]

Files (.txt; .csv; .xlsx; ...)

import/ export

Platform/ Environment

ANACONDA

jupyter

SETUP

Platform/ Environment

ANACONDA®

Platform/ Environment

ANACONDA

jupyter

my_notebook.ipnyb

Jupyter Notebook

[Python Code]

Platform/ Environment

ANACONDA

jupyter

my_notebook.ipnyb

Desktop>

Jupyter Notebook

[Python Code]

Platform/ Environment

ANACONDA

Python Basics

nepalschool.naamii.com.np/resources

Second Nepal Winter School in AI

Home · Speakers · Program · Registration · Logistics · **Resources** · People ·

First Nepal Winter School in AI 2018: Lecture Videos and Slides

Links

Course Materials

AI & Society Group Assignment

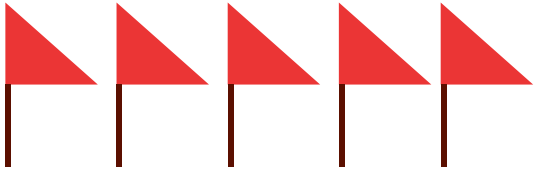
AI & Society Group Work Questionnaire

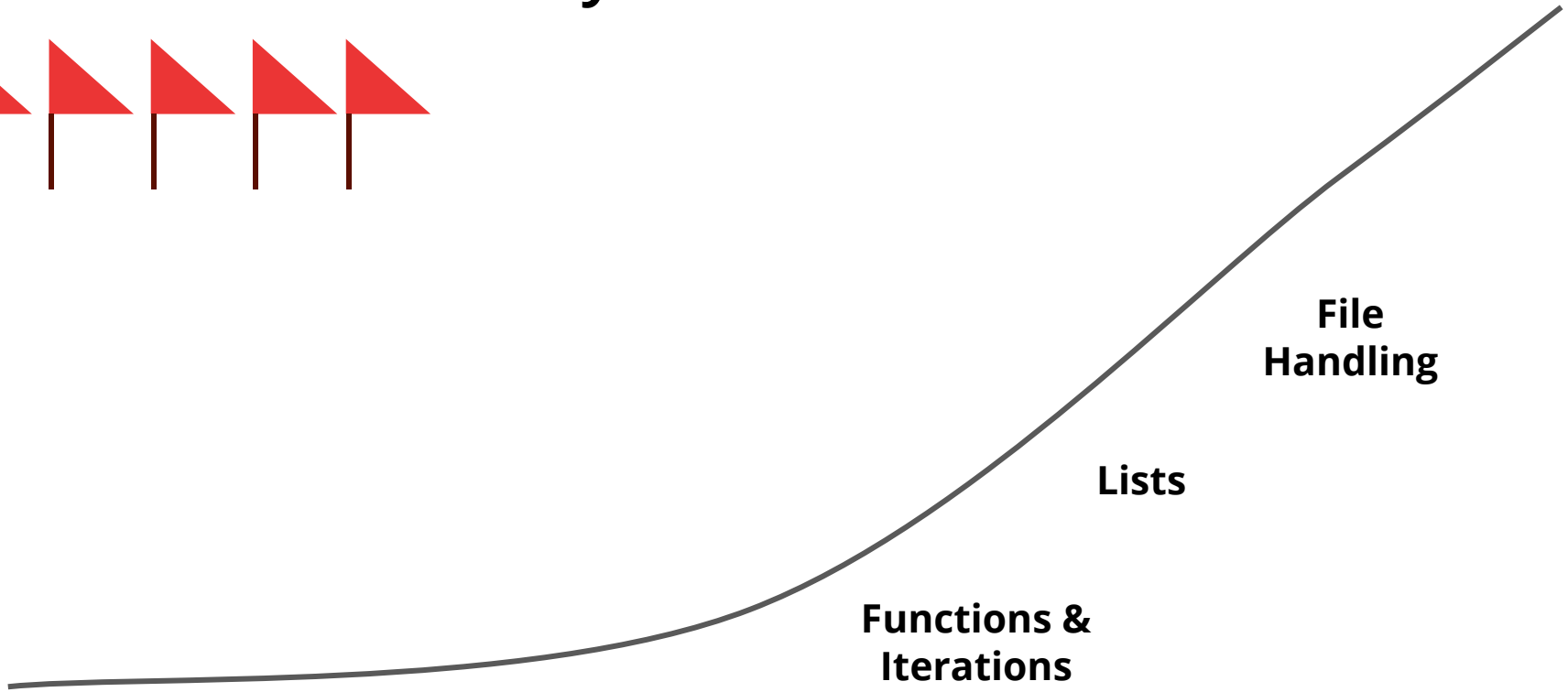
https://nepalschool.naamii.com.np/resources

Shared with me > ••• > CourseMaterialsForParticipants > Day1_Python

Name	Owner	Last opened by me	File size
Modeling	Joshi Sunnie	–	–
PythonforML_II	Ajad Chhatkuli	–	–
PythonNumpyPandas	me	1:31 PM	–

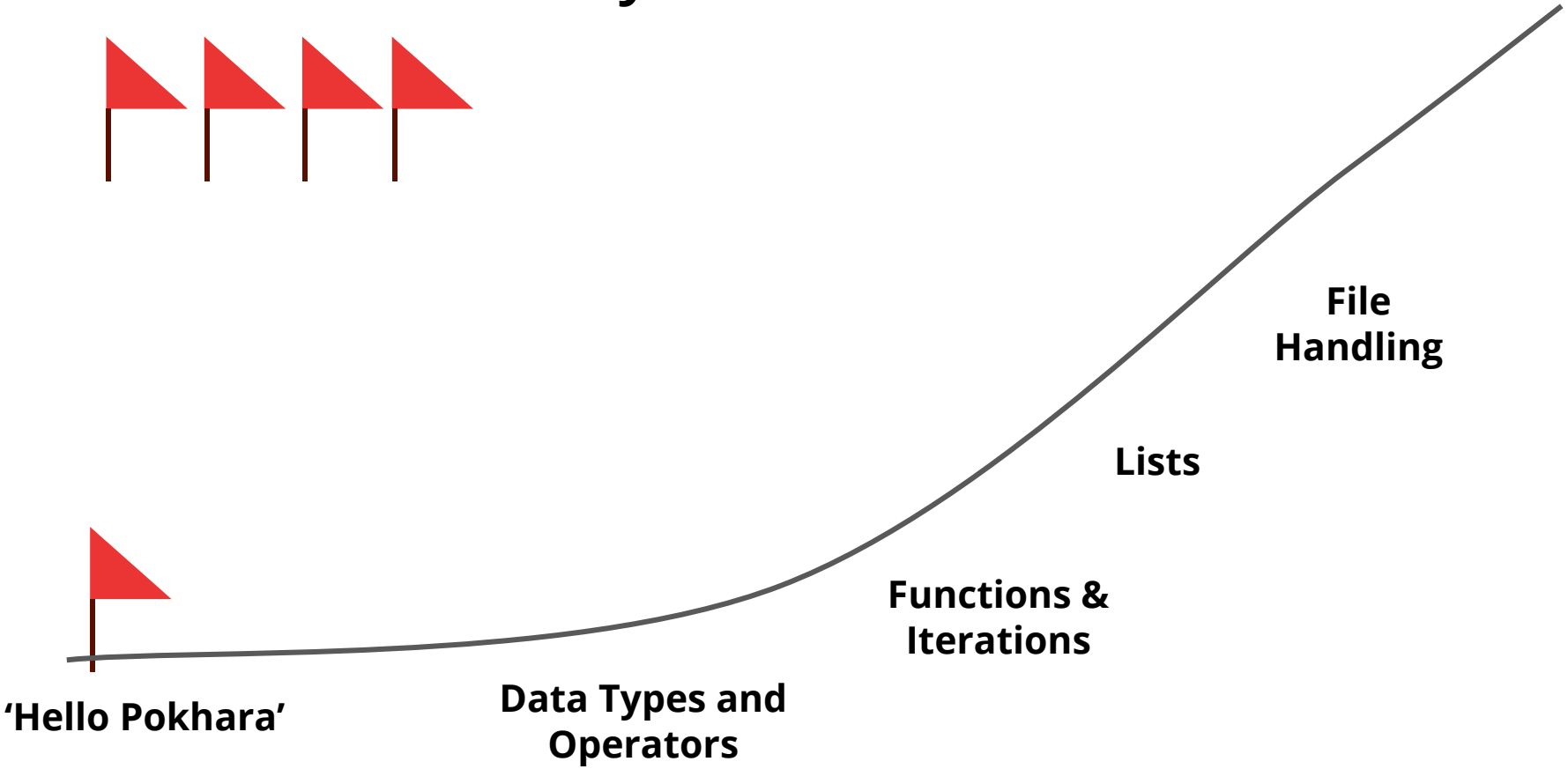
Python Basics





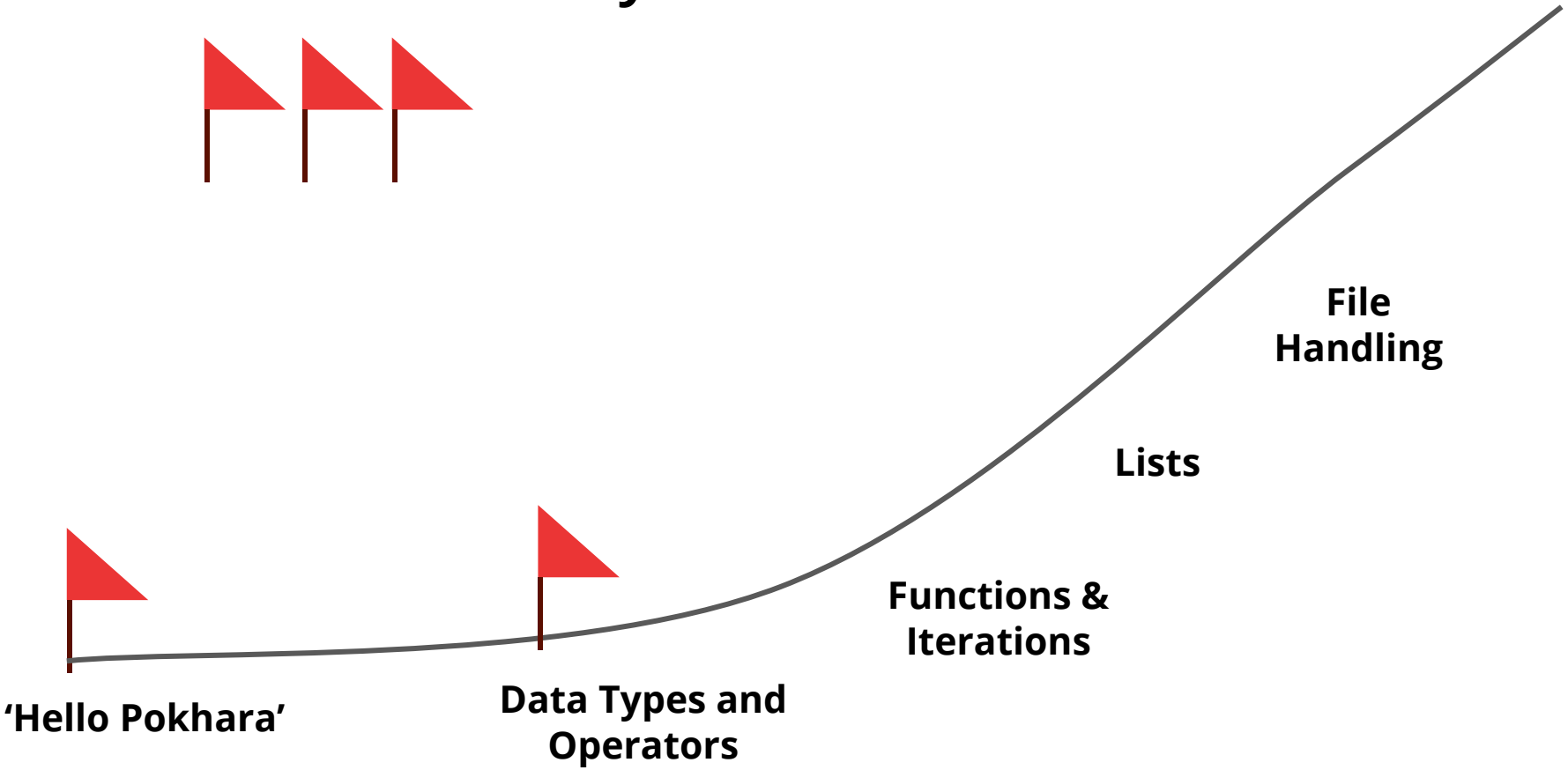
‘Hello Pokhara’

Python Basics



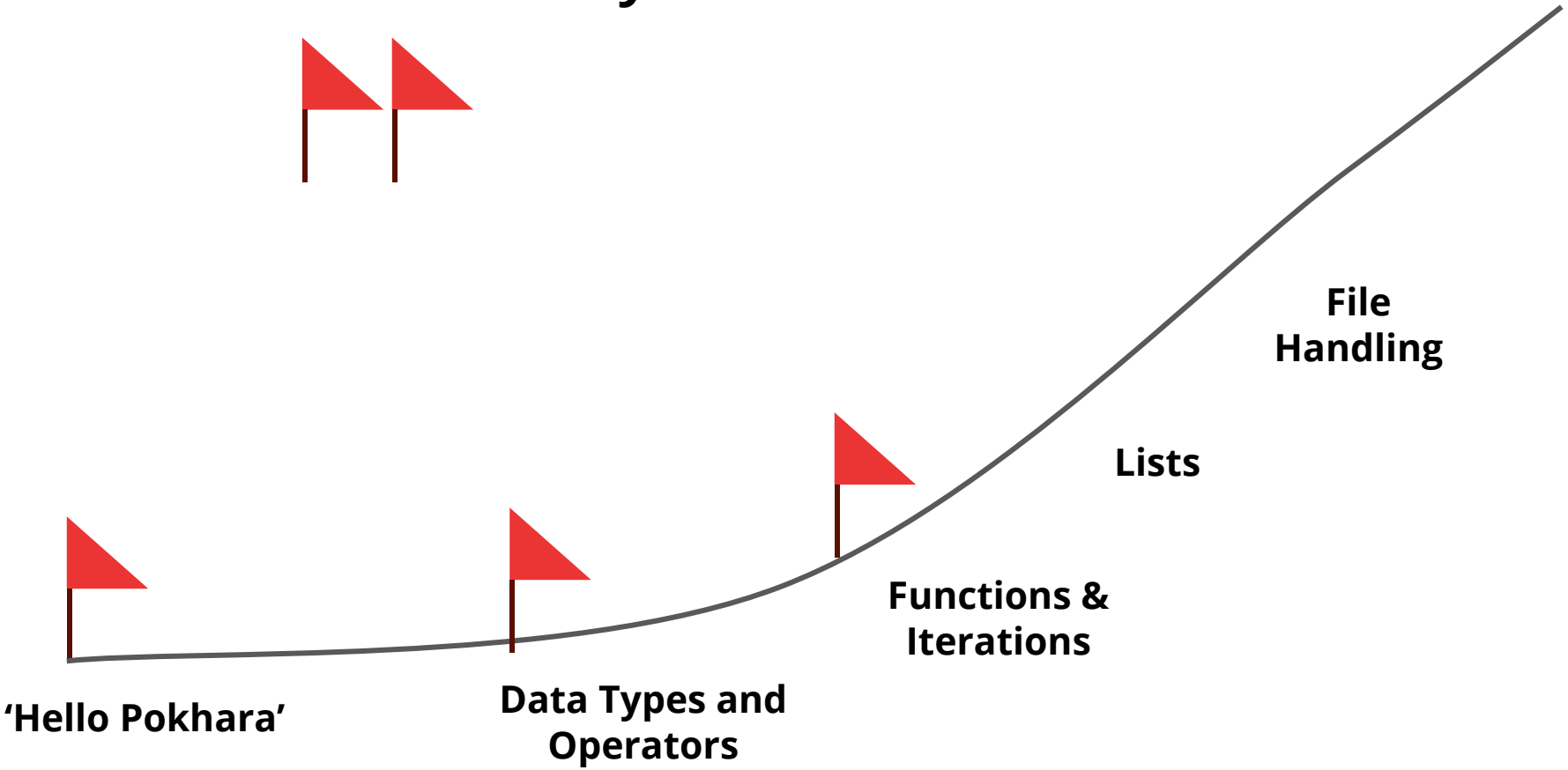
Python Data Types

Python Basics



Functions & Iterations

Python Basics



Lists & Dictionaries

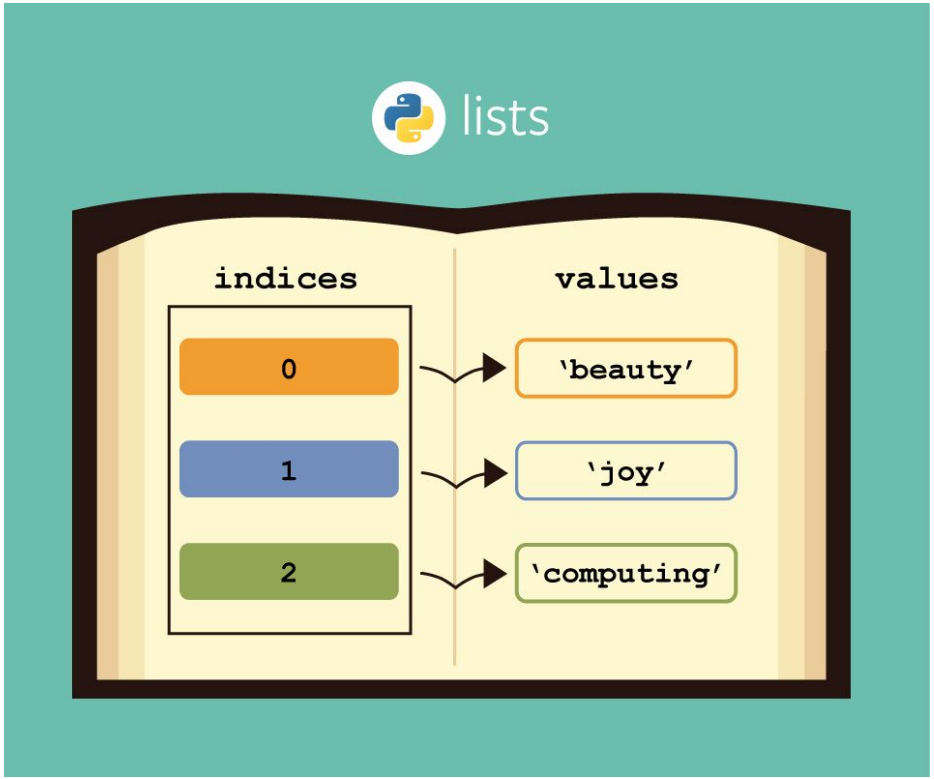
Python Dictionary

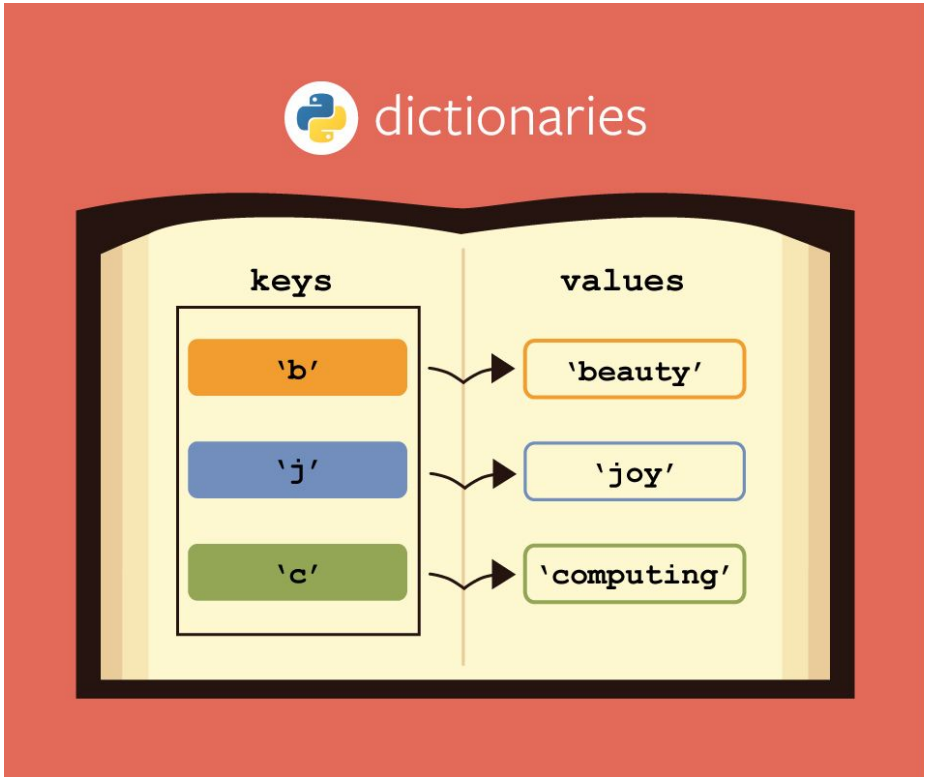
- py_dict = { 1: ‘Apple’, 2: ‘OnePlus’ }

key value key value

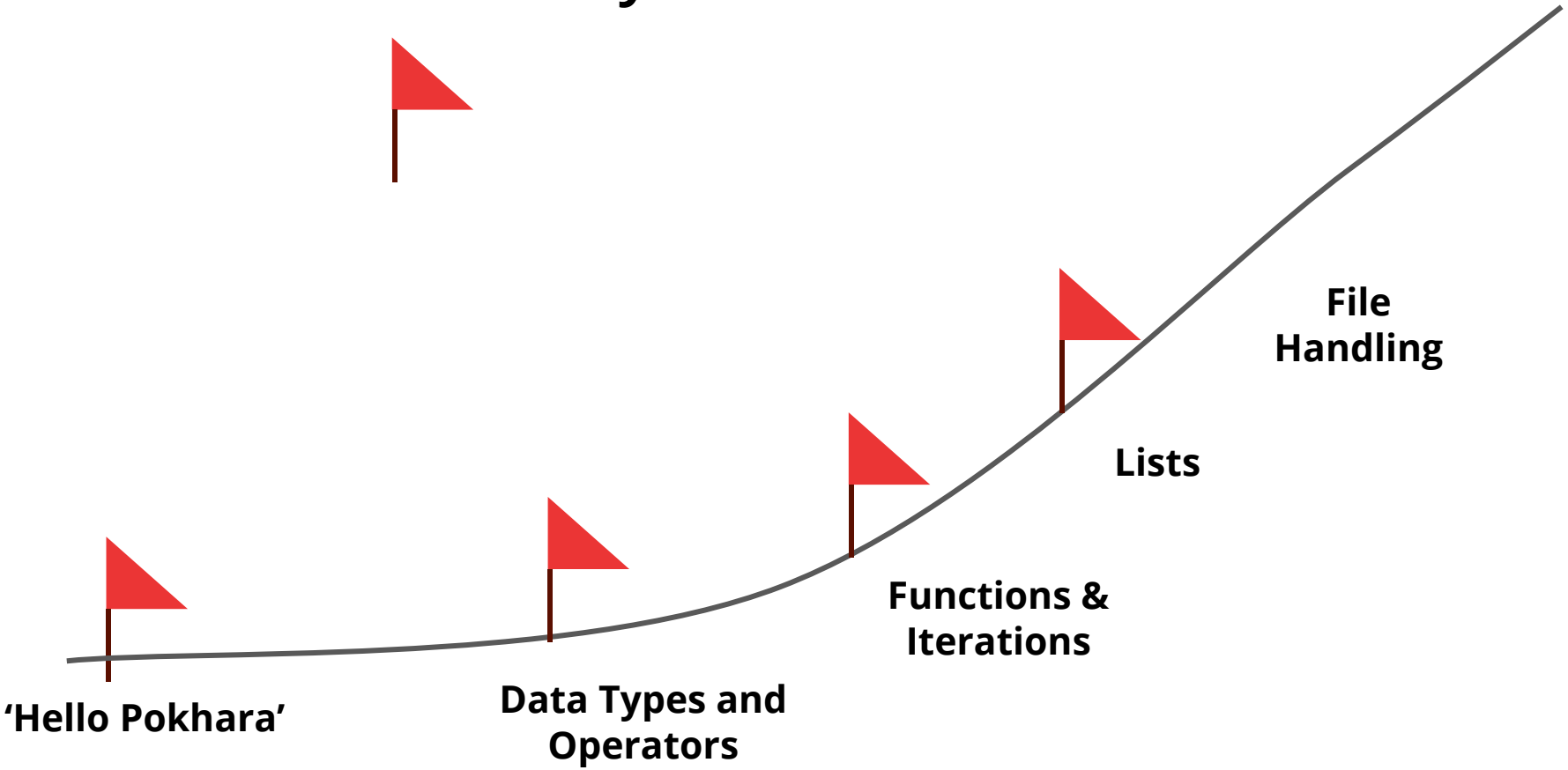
Item 1 Item 2

trytoprogram.com


lists
indices
values
0
1
2
'beauty'
'joy'
'computing'


dictionaries
keys
values
'b'
'j'
'c'
'beauty'
'joy'
'computing'

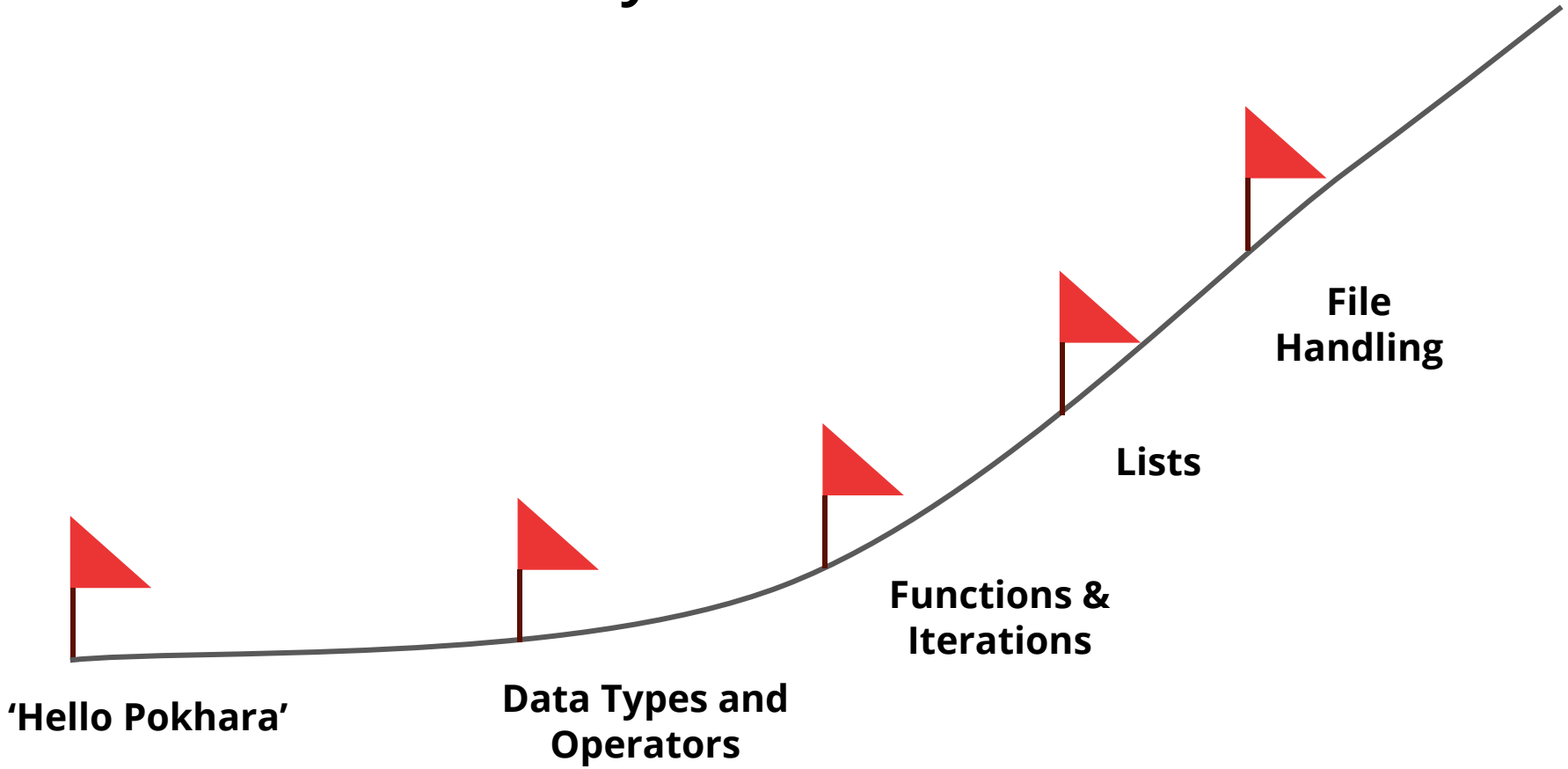
Python Basics



Python File-handling


File handling in Python
01
Opening files
02
Reading from files
03
Writing to files
04
Adding to files
05
Closing the files

Python Basics



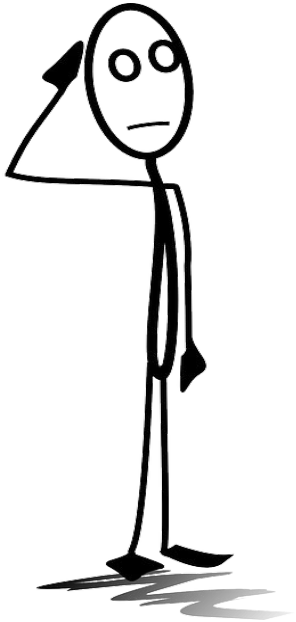
NumPy

Libraries/ Packages

NumPy

Numpy is a Python Library that provides a multidimensional array object.

It offers various built-in operations and functionalities around arrays.



But what about Python Lists?

Don’t they work good enough?

Numpy Array vs. Python Lists

Numpy Array it is a lot faster than a regular Python list.
A Python list can contain different kinds of data types such as integers, strings, Boolean, True, False and even lists.

On the other hand, NumPy arrays can hold only one type of data, and therefore doesn’t have to check the type of data type for every single element of the array when it is doing the computations.

This adds speed.

Array

List

Vector

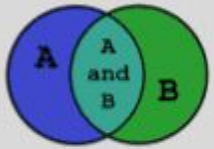
Matrix

Recap: Data Structure

List

Tuple

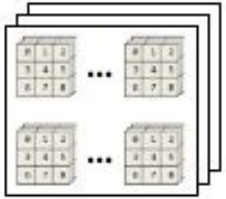
Dictionary



Sets

1. A list in Python is a heterogeneous container for items.
2. Tuple is immutable List. You cannot change items in tuple.
3. Python dictionary holds key-value pairs.
4. Set is a container that does not contain duplicate values and is unordered.

What is an array?

Dimensions	Example	Terminology
1	0 1 2	Vector
2	0 1 2 3 4 5 6 7 8	Matrix
3	0 1 2 3 4 5 6 7 8	3D Array (3rd order Tensor)
N		ND Array

Scalar	Vector	Matrix	Tensor
1	$\begin{bmatrix} 1 \\ 2 \end{bmatrix}$	$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$	$\begin{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 7 \end{bmatrix} & \begin{bmatrix} 3 & 2 \\ 5 & 4 \end{bmatrix} \end{bmatrix}$

Array Operations

Creating Arrays

Command

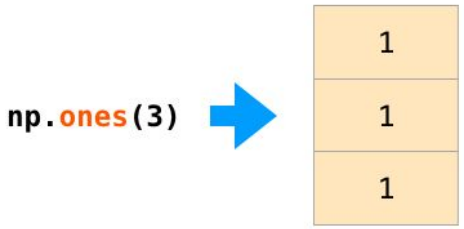
```
np.array([1,2,3])
```

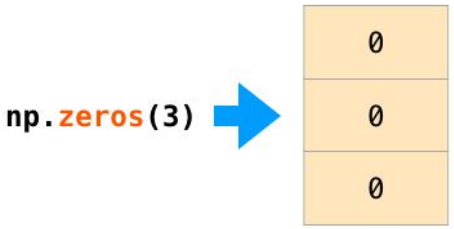
NumPy Array

1
2
3

http://jalammar.github.io/visual-numpy/

You can initialize arrays in different ways





Array Arithmetic

```
data = np.array([1,2])
```

data

1
2

```
ones = np.ones(2)
```

ones

1
1

```
data + ones
```

=

data

1
2

+

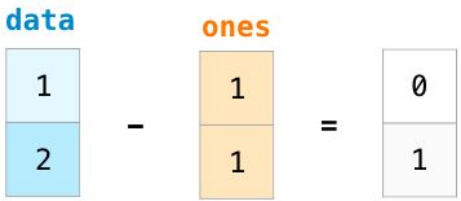
ones

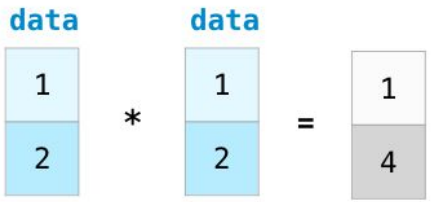
1
1

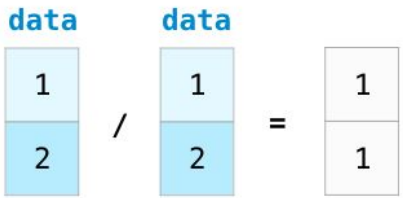
=

2
3

More Mathematical Operations



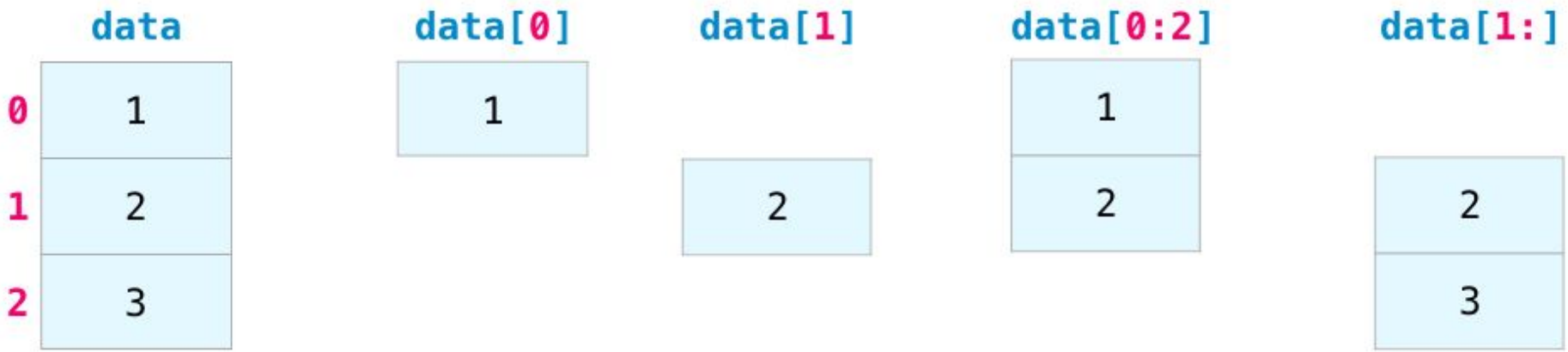




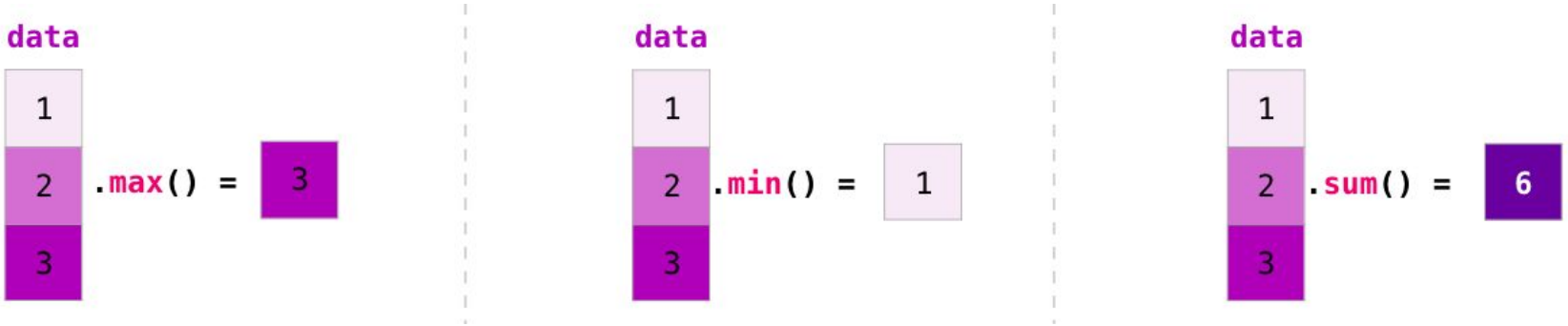
Scalar and Vector

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix} * 1.6 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} * \begin{bmatrix} 1.6 \\ 1.6 \end{bmatrix} = \begin{bmatrix} 1.6 \\ 3.2 \end{bmatrix}$$

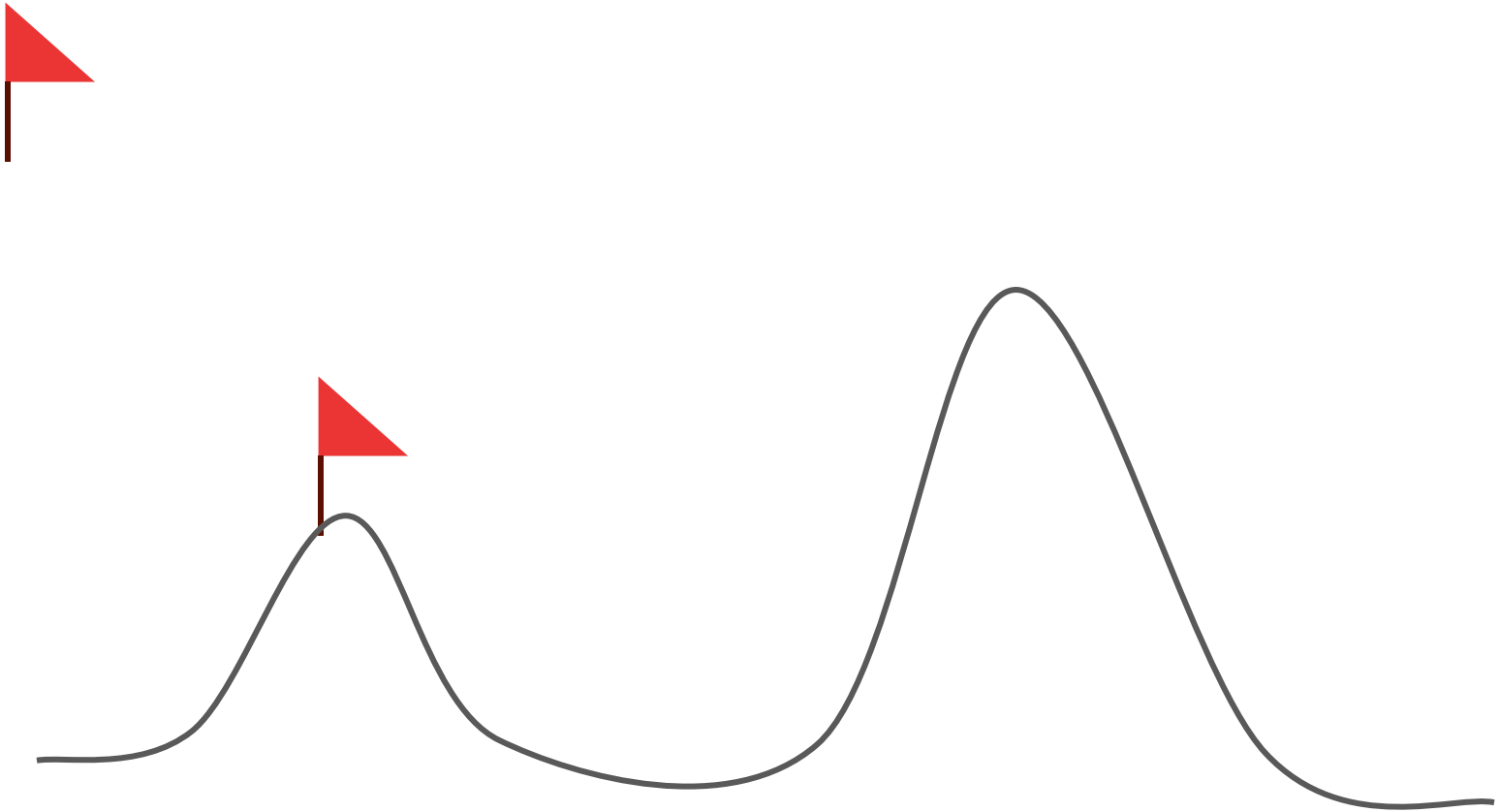
Indexing and Slicing



Aggregation



Numpy



Creating Matrix

```
np.array([[1,2],[3,4]])
```

1	2
3	4

```
np.ones((3,2))
```

2

3

1	1
1	1
1	1

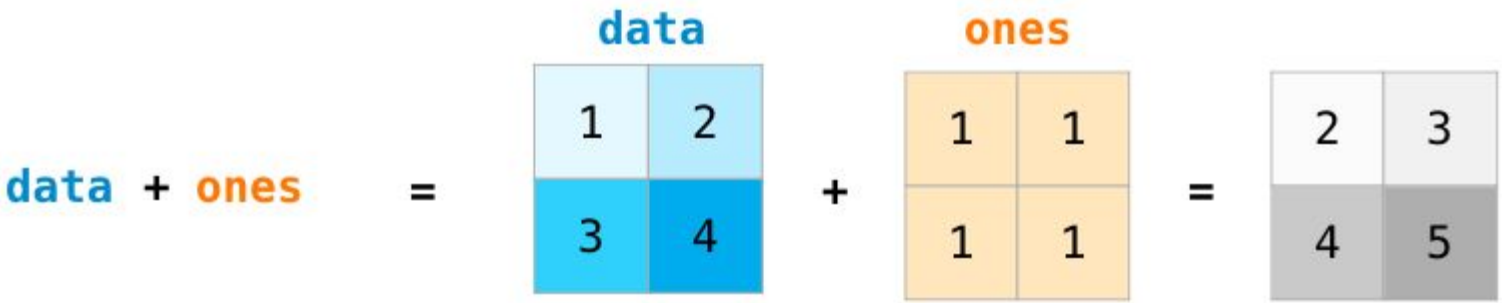
```
np.zeros((3,2))
```

0	0
0	0
0	0

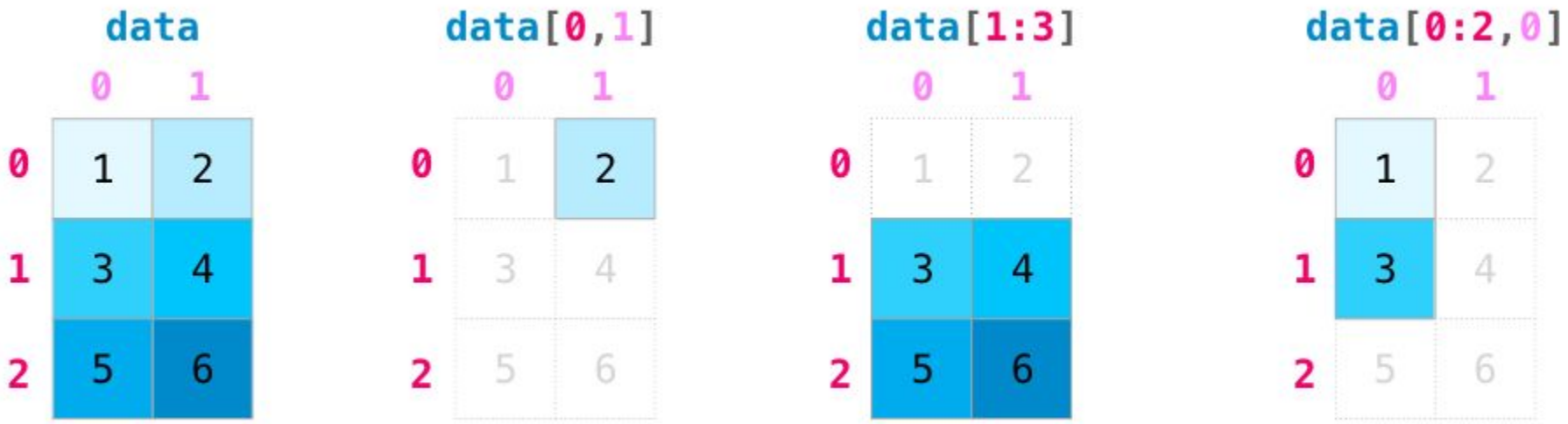
```
np.random.random((3,2))
```

0.37	0.88
0.75	0.79
0.63	0.16

Matrix Operations



Matrix Indexing & Slicing



Data[0, 3:5]

0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23
24	25	26	27	28	29
30	31	32	33	34	35

Data[4:, 4:]

0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23
24	25	26	27	28	29
30	31	32	33	34	35

Data[:, 2]

0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23
24	25	26	27	28	29
30	31	32	33	34	35

Transposing

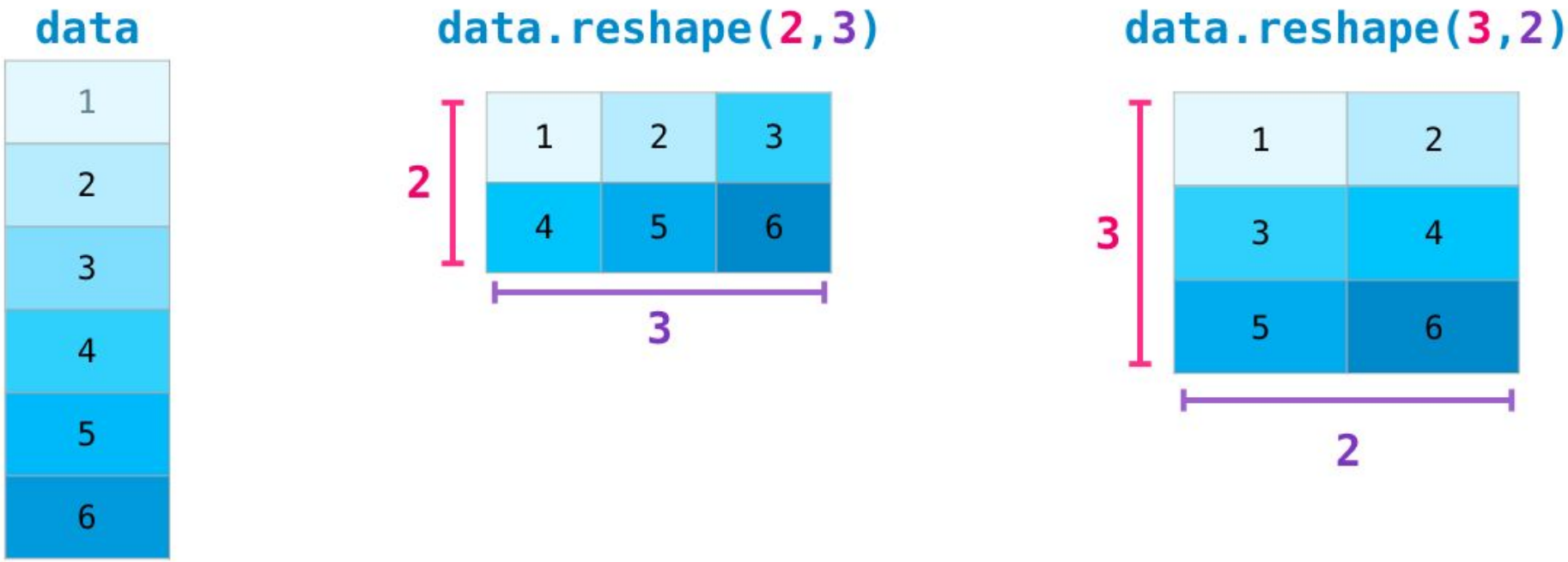
data

1	2
3	4
5	6

data.T

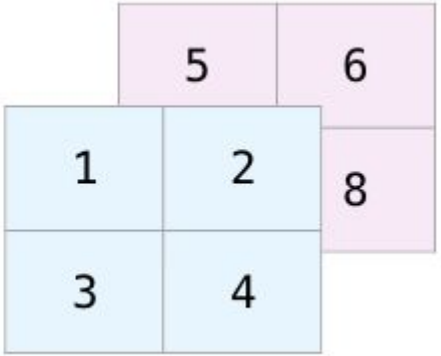
1	3	5
2	4	6

Reshaping

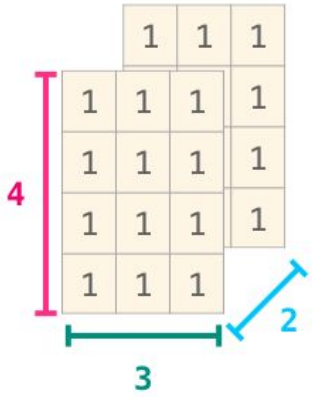

data
1
2
3
4
5
6
data.reshape(2,3)
1 2 3
4 5 6
2
3
data.reshape(3,2)
1 2
3 4
5 6
3
2

More Dimensions

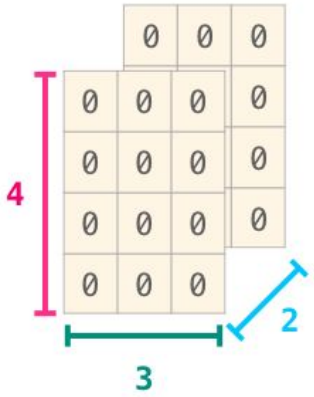
```
np.array([ [[1,2],[3,4]],
           [[5,6],[7,8]] ])
```



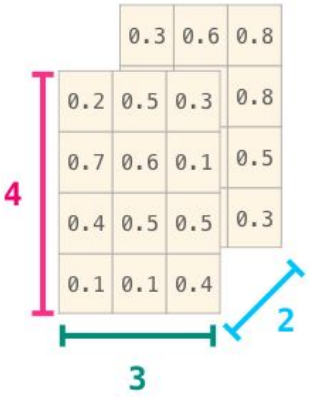
```
np.ones((4,3,2))
```



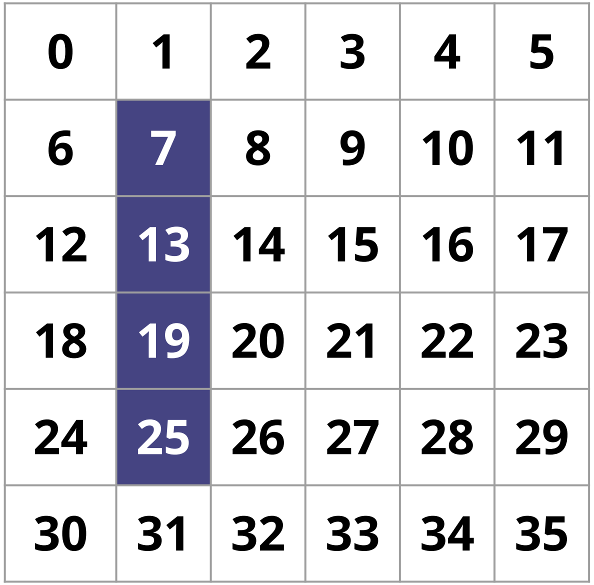

```
np.zeros((4,3,2))
```



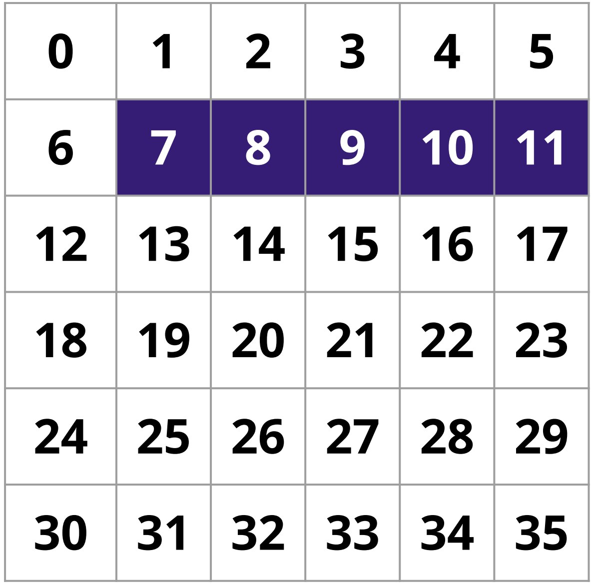

```
np.random.random((4,3,2))
```



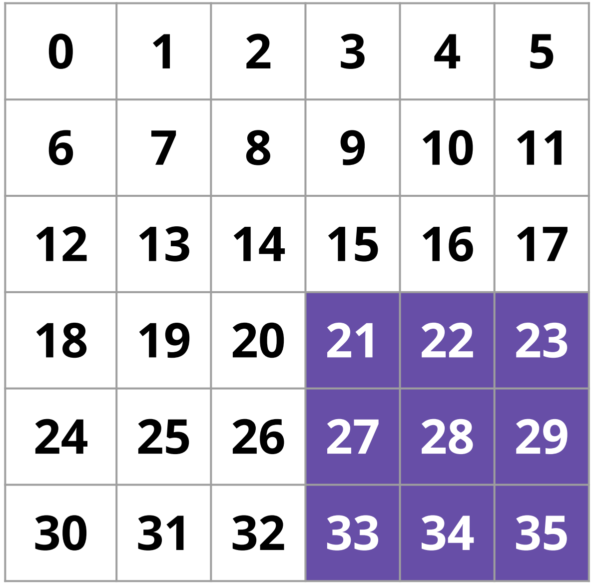
Exercise



0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23
24	25	26	27	28	29
30	31	32	33	34	35

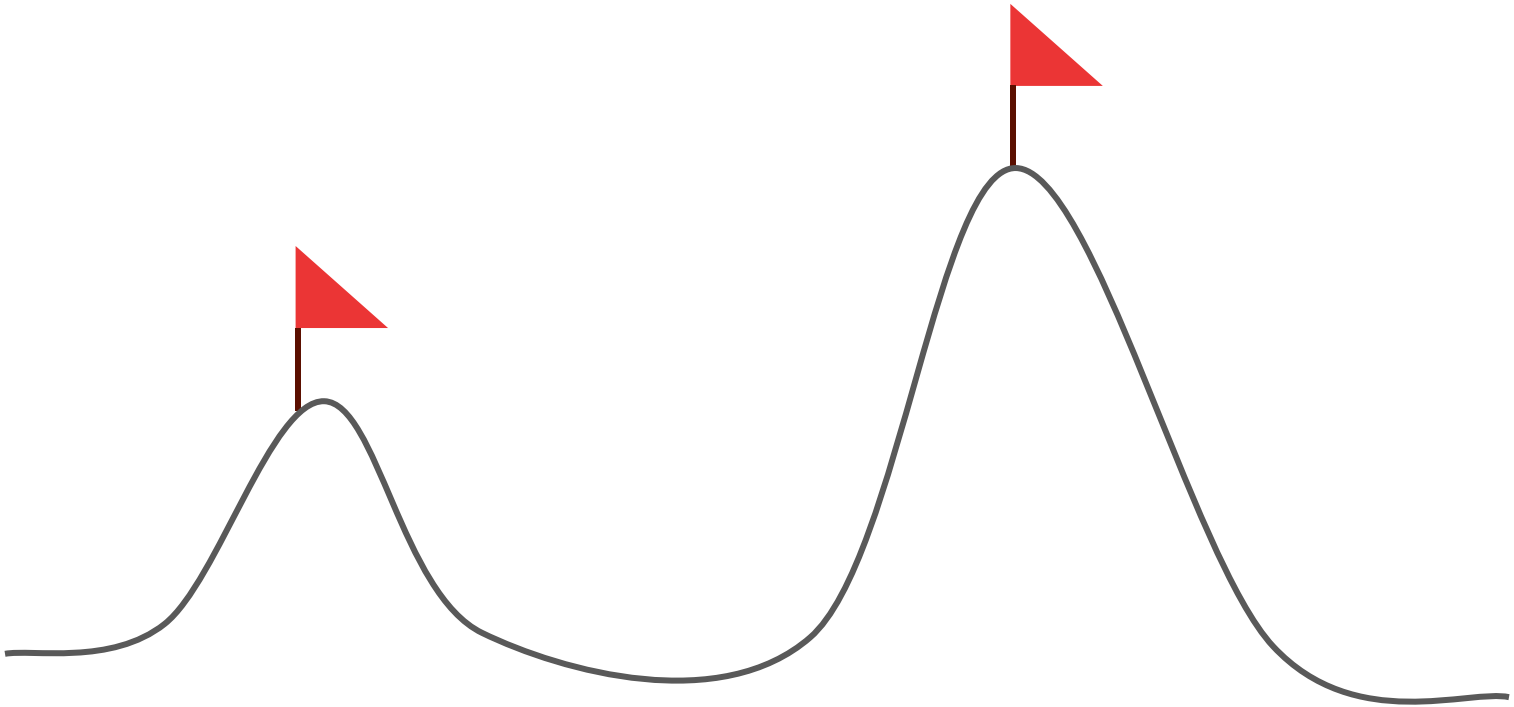


0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23
24	25	26	27	28	29
30	31	32	33	34	35



0	1	2	3	4	5
6	7	8	9	10	11
12	13	14	15	16	17
18	19	20	21	22	23
24	25	26	27	28	29
30	31	32	33	34	35

Numpy



pandas

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

Pandas is a Python library for data manipulation and analysis.

Pandas provides two classes:

- a **Series** object, which handles a single column of data;
- a **DataFrame** object, which handles multiple columns (like an Excel spreadsheet).

	NAME	AGE	DESIGNATION
1	a	20	VP
2	b	27	CEO
3	c	35	CFO
4	d	55	VP
5	e	18	VP
6	f	21	CEO
7	g	35	MD

Series

Panda Series

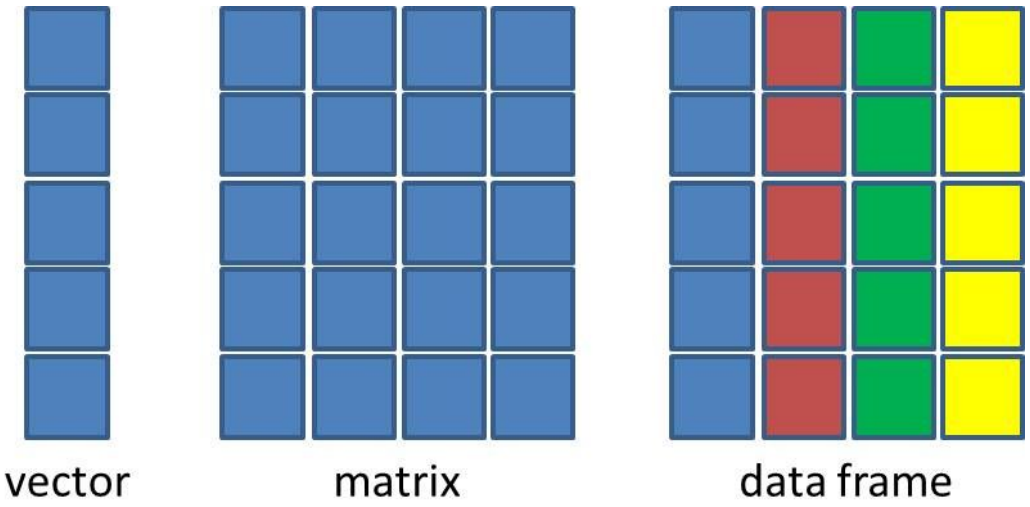
Pandas Series is a one-dimensional labeled array capable of holding any data type.
Pandas Series is equivalent to a column in an excel sheet.

	NAME	AGE	DESIGNATION
1	a	20	VP
2	b	27	CEO
3	c	35	CFO
4	d	55	VP
5	e	18	VP
6	f	21	CEO
7	g	35	MD

Series

Panda DataFrame

Equivalent to Excel Sheet



Data frames are collection of vectors of equal length stuck together.

They are different to matrices as they can contain vectors of different types.

Pandas