

# Data Processing in Python Part 2

Benjamin Rudski  
February 11, 2025



**McGill**

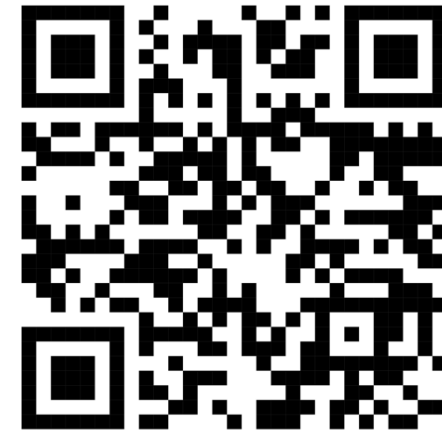
Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

QLS-MiCM mission statement: deliver quality workshops designed to help biomedical researchers develop the skills they need to succeed.



Location: 550 Sherbrooke  
Street, Montreal, Quebec



Scan the QR code to sign up  
for our **mailing list**

Contact: [workshop-micm@mcgill.ca](mailto:workshop-micm@mcgill.ca)

# Workshop Series

Workshop	Date	Location	Registration
How to think in Code	Jan. 28 1PM-3PM	EDUC 133	Closed
Intro to Git & GitHub	Jan. 30 1PM-5PM	EDUC 133	Closed
Intro to Unix	Feb. 6 1PM-5PM	EDUC 133	Closed
Intro to Python (Part 1)	Feb. 11 1PM-5PM	EDUC 133	<a href="#">Open</a>
Intro to R (Part 1)	Feb. 13 1PM-5PM	EDUC 133	<a href="#">Open</a>
Exploring MATLAB	Feb. 18 1PM-5PM	EDUC 133	<a href="#">Open</a>
Statistics in R (Part 2)	Feb. 20 1PM-5PM	EDUC 133	<a href="#">Open</a>
Data Processing in Python	Feb. 25 1PM-5PM	EDUC 133	<a href="#">Open</a>
Intro to Machine Learning	Mar. 13 1PM-5PM	EDUC 133	TBA
Intro to R (Part 1)	TBA	EDUC 133	TBA
Intro to Python (Part 1)	TBA	EDUC 133	TBA

<https://www.mcgill.ca/micm/training/workshops-series>



McGill

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

# Outline

- 1. Module 1 – Modules and Packages (40 minutes)**
  - a. Using Modules
  - b. A Brief Intro to Package Management
  - c. Exercise
- 2. Module 2 – Introduction to NumPy Arrays (50 minutes)**
  - a. Introduction to Arrays
  - b. Introducing NumPy
  - c. Array Operations
  - d. Exercise



# Outline

## 3. **Module 3 – Visualising Data with Matplotlib (50 minutes)**

- a. Creating Plots with Matplotlib
- b. Exploring the Matplotlib Documentation
- c. Exercise

## 4. **Module 4 – Intro to Tabular Data with Pandas (30 minutes)**

- a. Fundamentals of pandas
- b. Exploring the pandas Documentation

## 5. **Module 5 – A Brief Guide to Exploring the Unknown (10 minutes)**

- a. What to learn next? How?
- b. How to get help and how not to get help
- c. Other cool programming topics



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

# Interactive Workshop!

- That's pretty much all that will be in the slides... For the rest, we'll go to a Jupyter Notebook:



**To the repository!**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

# Module 2

## Introduction to NumPy Arrays



**my\_array**

<b>(0, 0)</b>	<b>(0, 1)</b>	<b>(0, 2)</b>	<b>(0, 3)</b>	<b>(0, 4)</b>	<b>(0, 5)</b>
<b>(1, 0)</b>	<b>(1, 1)</b>	<b>(1, 2)</b>	<b>(1, 3)</b>	<b>(1, 4)</b>	<b>(1, 5)</b>
<b>(2, 0)</b>	<b>(2, 1)</b>	<b>(2, 2)</b>	<b>(2, 3)</b>	<b>(2, 4)</b>	<b>(2, 5)</b>
<b>(3, 0)</b>	<b>(3, 1)</b>	<b>(3, 2)</b>	<b>(3, 3)</b>	<b>(3, 4)</b>	<b>(3, 5)</b>
<b>(4, 0)</b>	<b>(4, 1)</b>	<b>(4, 2)</b>	<b>(4, 3)</b>	<b>(4, 4)</b>	<b>(4, 5)</b>

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant



**my\_array[0]?**

(0, 0)	(0, 1)	(0, 2)	(0, 3)	(0, 4)	(0, 5)
(1, 0)	(1, 1)	(1, 2)	(1, 3)	(1, 4)	(1, 5)
(2, 0)	(2, 1)	(2, 2)	(2, 3)	(2, 4)	(2, 5)
(3, 0)	(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)
(4, 0)	(4, 1)	(4, 2)	(4, 3)	(4, 4)	(4, 5)

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

**my\_array[0]**

<b>(0, 0)      (0, 1)      (0, 2)      (0, 3)      (0, 4)      (0, 5)</b>					
<b>(1, 0)      (1, 1)      (1, 2)      (1, 3)      (1, 4)      (1, 5)</b>					
<b>(2, 0)</b>	<b>(2, 1)</b>	<b>(2, 2)</b>	<b>(2, 3)</b>	<b>(2, 4)</b>	<b>(2, 5)</b>
<b>(3, 0)      (3, 1)      (3, 2)      (3, 3)      (3, 4)      (3, 5)</b>					
<b>(4, 0)</b>	<b>(4, 1)</b>	<b>(4, 2)</b>	<b>(4, 3)</b>	<b>(4, 4)</b>	<b>(4, 5)</b>

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

**my\_array[:, 0]?**

<b>(0, 0)</b>	<b>(0, 1)</b>	<b>(0, 2)</b>	<b>(0, 3)</b>	<b>(0, 4)</b>	<b>(0, 5)</b>
<b>(1, 0)</b>	<b>(1, 1)</b>	<b>(1, 2)</b>	<b>(1, 3)</b>	<b>(1, 4)</b>	<b>(1, 5)</b>
<b>(2, 0)</b>	<b>(2, 1)</b>	<b>(2, 2)</b>	<b>(2, 3)</b>	<b>(2, 4)</b>	<b>(2, 5)</b>
<b>(3, 0)</b>	<b>(3, 1)</b>	<b>(3, 2)</b>	<b>(3, 3)</b>	<b>(3, 4)</b>	<b>(3, 5)</b>
<b>(4, 0)</b>	<b>(4, 1)</b>	<b>(4, 2)</b>	<b>(4, 3)</b>	<b>(4, 4)</b>	<b>(4, 5)</b>

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

**my\_array[:, 0]**

<b>(0, 0)</b>	<b>(0, 1)</b>	<b>(0, 2)</b>	<b>(0, 3)</b>	<b>(0, 4)</b>	<b>(0, 5)</b>
<b>(1, 0)</b>	<b>(1, 1)</b>	<b>(1, 2)</b>	<b>(1, 3)</b>	<b>(1, 4)</b>	<b>(1, 5)</b>
<b>(2, 0)</b>	<b>(2, 1)</b>	<b>(2, 2)</b>	<b>(2, 3)</b>	<b>(2, 4)</b>	<b>(2, 5)</b>
<b>(3, 0)</b>	<b>(3, 1)</b>	<b>(3, 2)</b>	<b>(3, 3)</b>	<b>(3, 4)</b>	<b>(3, 5)</b>
<b>(4, 0)</b>	<b>(4, 1)</b>	<b>(4, 2)</b>	<b>(4, 3)</b>	<b>(4, 4)</b>	<b>(4, 5)</b>

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

**my\_array[1:3, 2:4]?**

<b>(0, 0)</b>	<b>(0, 1)</b>	<b>(0, 2)</b>	<b>(0, 3)</b>	<b>(0, 4)</b>	<b>(0, 5)</b>
<b>(1, 0)</b>	<b>(1, 1)</b>	<b>(1, 2)</b>	<b>(1, 3)</b>	<b>(1, 4)</b>	<b>(1, 5)</b>
<b>(2, 0)</b>	<b>(2, 1)</b>	<b>(2, 2)</b>	<b>(2, 3)</b>	<b>(2, 4)</b>	<b>(2, 5)</b>
<b>(3, 0)</b>	<b>(3, 1)</b>	<b>(3, 2)</b>	<b>(3, 3)</b>	<b>(3, 4)</b>	<b>(3, 5)</b>
<b>(4, 0)</b>	<b>(4, 1)</b>	<b>(4, 2)</b>	<b>(4, 3)</b>	<b>(4, 4)</b>	<b>(4, 5)</b>

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

**my\_array[1:3, 2:4]**

(0, 0)	(0, 1)	(0, 2)	(0, 3)	(0, 4)	(0, 5)
(1, 0)	(1, 1)	(1, 2)	(1, 3)	(1, 4)	(1, 5)
(2, 0)	(2, 1)	(2, 2)	(2, 3)	(2, 4)	(2, 5)
(3, 0)	(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)
(4, 0)	(4, 1)	(4, 2)	(4, 3)	(4, 4)	(4, 5)

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

**my\_array[0:5:2]?**

(0, 0)	(0, 1)	(0, 2)	(0, 3)	(0, 4)	(0, 5)
(1, 0)	(1, 1)	(1, 2)	(1, 3)	(1, 4)	(1, 5)
(2, 0)	(2, 1)	(2, 2)	(2, 3)	(2, 4)	(2, 5)
(3, 0)	(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)
(4, 0)	(4, 1)	(4, 2)	(4, 3)	(4, 4)	(4, 5)

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

**my\_array[0:5:2]**

(0, 0)	(0, 1)	(0, 2)	(0, 3)	(0, 4)	(0, 5)
--------	--------	--------	--------	--------	--------

(1, 0)	(1, 1)	(1, 2)	(1, 3)	(1, 4)	(1, 5)
--------	--------	--------	--------	--------	--------

(2, 0)	(2, 1)	(2, 2)	(2, 3)	(2, 4)	(2, 5)
--------	--------	--------	--------	--------	--------

(3, 0)	(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)
--------	--------	--------	--------	--------	--------

(4, 0)	(4, 1)	(4, 2)	(4, 3)	(4, 4)	(4, 5)
--------	--------	--------	--------	--------	--------

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant



**my\_array**

(0, 0)					(0, 5)
(1, 0)	(1, 1)	(1, 2)	(1, 3)	(1, 4)	(1, 5)
(2, 0)	(2, 1)	(2, 2)	(2, 3)	(2, 4)	(2, 5)
(3, 0)	(3, 1)	(3, 2)	(3, 3)	(3, 4)	(3, 5)
(4, 0)	(4, 1)	(4, 2)	(4, 3)	(4, 4)	(4, 5)

**Shape: (5, 6)**



**McGill**

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

# Module 4

## Intro to Tabular Data with pandas



By name: `my_df.loc`

	Series 0	Series 1	Series 2
Row 0	<code>["row 0", "series 0"]</code>	<code>["row 0", "series 1"]</code>	<code>["row 0", "series 2"]</code>
Row 1	<code>["row 1", "series 0"]</code>	<code>["row 1", "series 1"]</code>	<code>["row 1", "series 2"]</code>
Row 2	<code>["row 2", "series 0"]</code>	<code>["row 2", "series 1"]</code>	<code>["row 2", "series 2"]</code>



By numerical index: `my_df.iloc`

	Series 0	Series 1	Series 2
Row 0	[0, 0]	[0, 1]	[0, 2]
Row 1	[1, 0]	[1, 1]	[1, 2]
Row 2	[2, 0]	[2, 1]	[2, 2]



# To summarize

- ✓ **Modules** and **packages** allow for code written by others to be easily imported and reused.
- ✓ **NumPy arrays** allow easily storing many numbers and performing operations without having to loop.
- ✓ **Matplotlib** can be used to generate many different types of plots.
- ✓ **pandas DataFrames** represent data in tables.
- ✓ Big projects have **documentation** to explain their functionality.

## Now you are ready to:

- Import code from existing modules and packages.
- Use NumPy to easily process multidimensional data.
- Use Matplotlib to generate different types of plots to visualise data.
- Approach a new package and explore its documentation and examples.



McGill

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant

# Acknowledgements

- Thank you to QLS-MiCM for giving me this opportunity and for helping me along the way.
- Thank you to the professors from the McGill School of Computer Science for helping me along my programming journey and for inspiring me to share my programming experience with others.
- Thank you to Professor Mathieu Blanchette, whose COMP 204 course helped introduce me to Python (back in Fall 2018).
- Thank you to the Python, NumPy, Matplotlib and pandas communities!



McGill

Quantitative Life  
Sciences

Sciences quantitatives  
du vivant