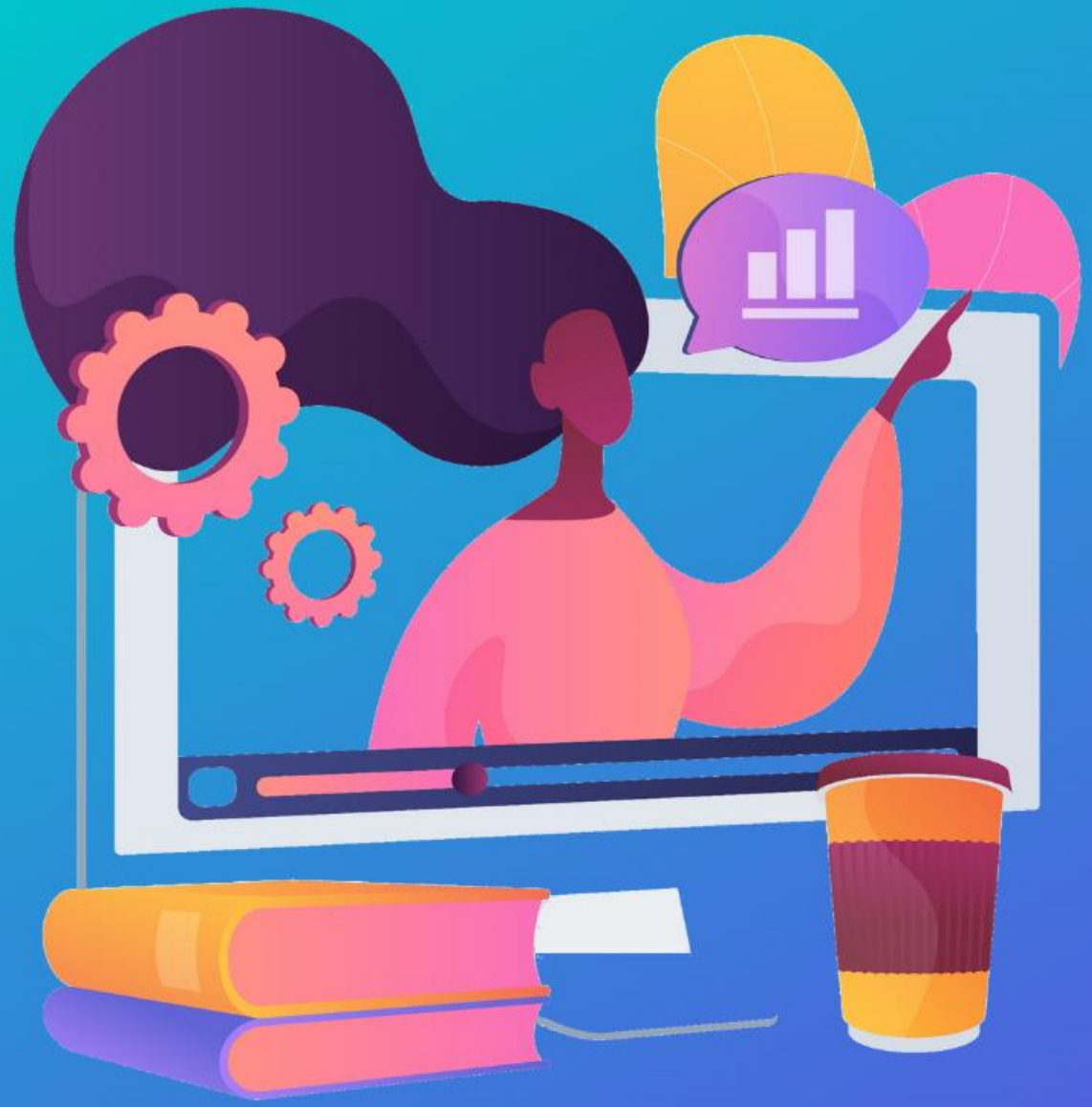


MBAZA NLP COMMUNITY

VIRTUAL TRAINING

22-24.06 | 3 - 5 PM

BASIC DATA WRANGLING WITH PANDAS
TEXT PRE-PROCESSING BASICS
NLP MODELING & ALGORITHMS

german cooperation
DEUTSCHE ZUSAMMENARBEIT

Implemented by
giz Deutsche Gesellschaft für Internationale Zusammenarbeit (GIZ) GmbH

Digital Transformation Center Rwanda

FAIR FORWARD
Artificial Intelligence for all.

DIGITAL UMUGANDA

Why are we here?

Module 1
Basic Data Wrangling with Pandas

Any Data Science work you do in the future

Module 2
Text Pre-Processing Basics

Any NLP work you do in the future

Dataset cleaning challenge in July

Module 3
NLP Modelling & Algorithms

Your introduction to Machine Learning

Community activities on Chatbots and Voice

Mbaza NLP Community

Knowledge sharing

- NLP, e. g. speech recognition and chatbots
- Software architecture design
- DevOps

Access to:

- Developed NLP models
- Datasets
- Infrastructure: Dev/test environments, computing resources

Use case development

- Gain practice & improve your skills
- Practical experiences to show to employers
- Work on projects with rewards

Learning Outcomes of Today

Module 1
Basic Data Wrangling with Pandas

You understand:

what Jupyter Notebooks are and how to use them
what the Pandas library does
the basic Pandas data structures

You can:

use Pandas to import CSV data
apply basic methods of inspecting your data
filter and select data
combine data from various sources
apply basic data manipulation methods

Jupyter Notebooks

- Web-based interactive coding environment for Python and other languages
- Also supports Markdown cells for descriptions and explanations

Code cell

Cell output

Markdown cell

jupyter matrix

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

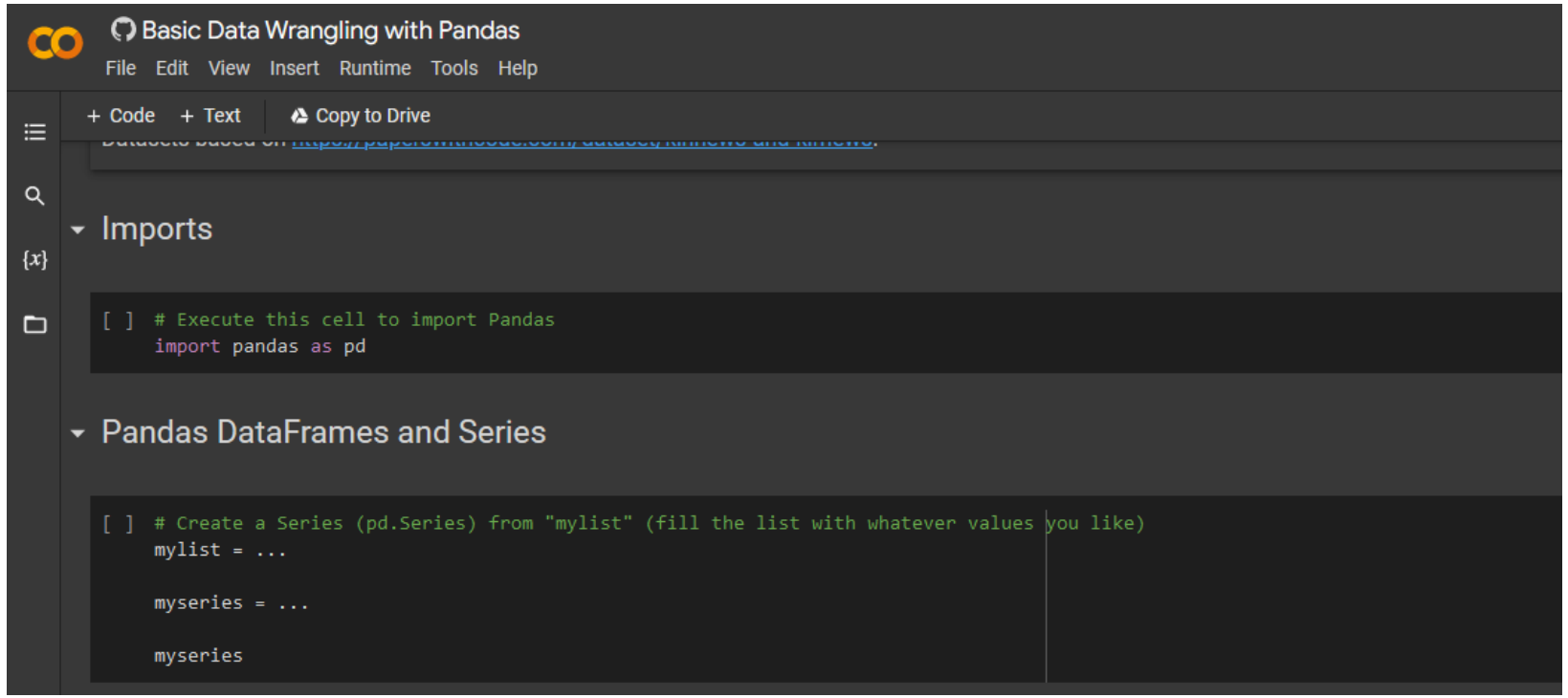
Run Code

```
In [1]: import numpy as np

In [2]: np.__version__

Out[2]: '1.15.4'

In [3]: n = 4 # n - order square matrix
        h = 0.5
        np.fromfunction(lambda i, j: h / ((i - j) ** 2 + h * h), (n, n), dtype=float)

Out[3]: array([[2.        , 0.4       , 0.11764706, 0.05405405],
               [0.4       , 2.        , 0.4       , 0.11764706],
               [0.11764706, 0.4       , 2.        , 0.4       ],
               [0.05405405, 0.11764706, 0.4       , 2.        ]])
```

Note. To calculate the matrix size 5000x5000, need to replace n = 4 in the previous input cell on n = 5000 и to push Shift+Enter

Google Colab

- Jupyter Notebook hosted online by Google with some added features
- Your code is executed on Google's servers



Let's get started!

Open the Colab link

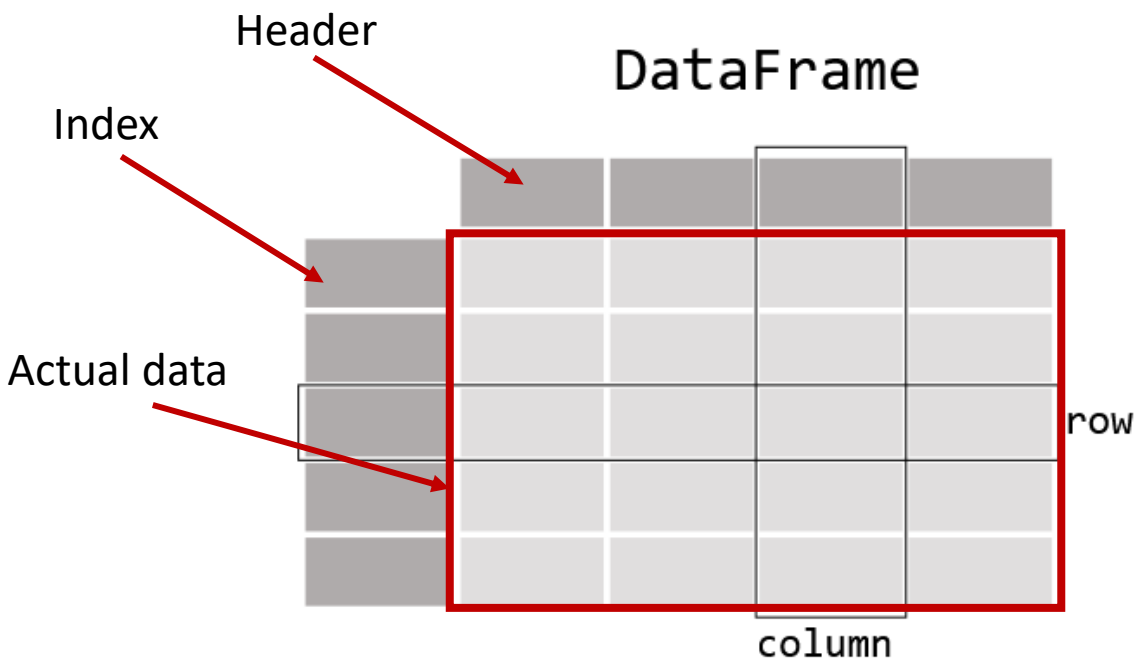
Pandas is the core library for Data Science with Python

- Open-source Python library for data analysis & management
- Provides data structures for table-like data as well as tools for data manipulation (e. g. filtering, merging, or sorting).

pandas

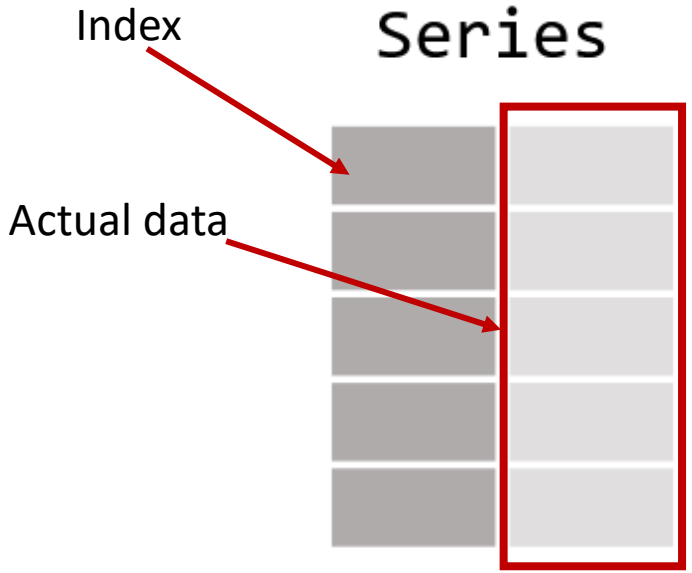
Pandas has two basic **data structures**

`DataFrame`: Two-dimensional, can be thought of as table




```
df = pd.DataFrame({'number': [1, 2, 3],
                   'squared': [1, 4, 9]})
```

`Series`: One-dimensional, more similar to a list




```
s = pd.Series([1, 2, 3])
```

Importing data

- A common file format in data science is **CSV** (comma-separated values)

Header row

data > labels.csv

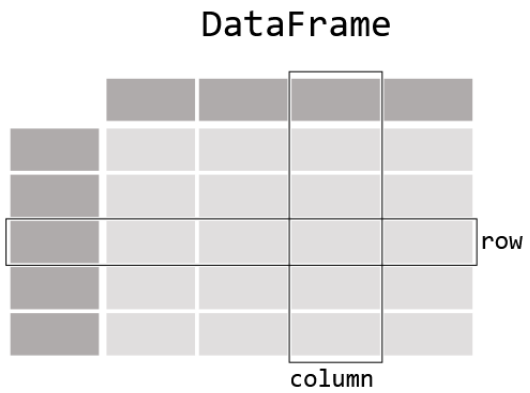
```
number,label
1,politics
2,sport
3,economy
4,health
5,entertainment
```

- Pandas makes it easy to import many file formats, including CSVs

```
data = pd.read_csv('filepath')
```

Understanding your data

There are several ways to inspect your data



1. Console output: Putting the variable name holding your data as the last row of the notebook cell prints it out

```
data
✓ 13.8s
```

	label	title
0	2	ikipe y' u rwanda amavubi yahesheje u rwanda a...
1	11	urubyiruko itorero erc giterane cy'ububyutse k...
2	4	rusizi bambaye udupfukamunwa n'ubwo bamwe bata...

2. The `shape` attribute of your data variable shows you the number of rows and columns of a DataFrame

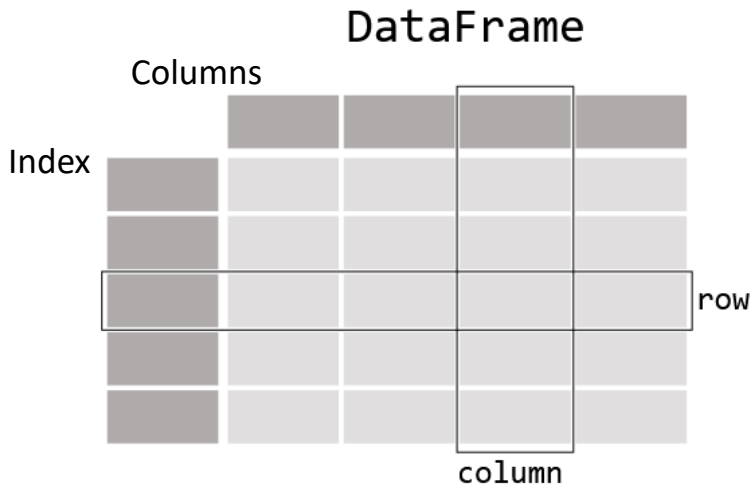
```
data.shape
✓ 1.8s
(500, 3)
```

3. The `head(x)` method returns the first `x` rows of your data

```
data.head(3)
✓ 0.6s
```

	label	title
0	2	ikipe y' u rwanda amavubi yahesheje u rwanda a...
1	11	urubyiruko itorero erc giterane cy'ububyutse k...
2	4	rusizi bambaye udupfukamunwa n'ubwo bamwe bata...

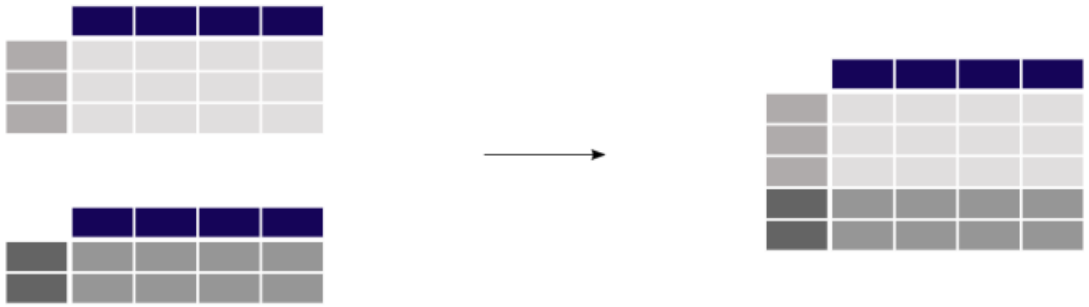
Filtering & selecting data



Task	Code	Output
Select one column of a `DataFrame`	`data["column_name"]`	`Series`
Select rows of a `DataFrame` by their Index number	`data[index_start:index_end]`	`DataFrame`
Select multiple columns of a `DataFrame` in the order you want them	`data[["column A", "column B"]]`	`DataFrame`
Select rows from a specific column	`data["column_name"][index_start:index_end]`	`Series`
Select only rows where some condition is met	`data[data["column_name"] == 5]`	`DataFrame`

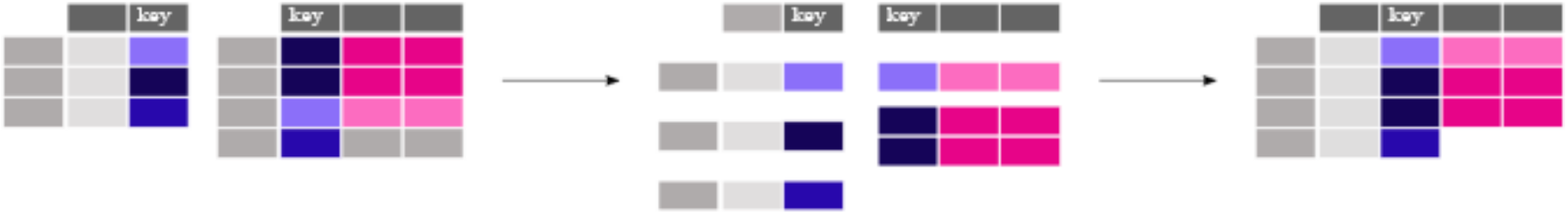
Combining data from multiple source

Concatenating two `DataFrames` – `pd.concat()`



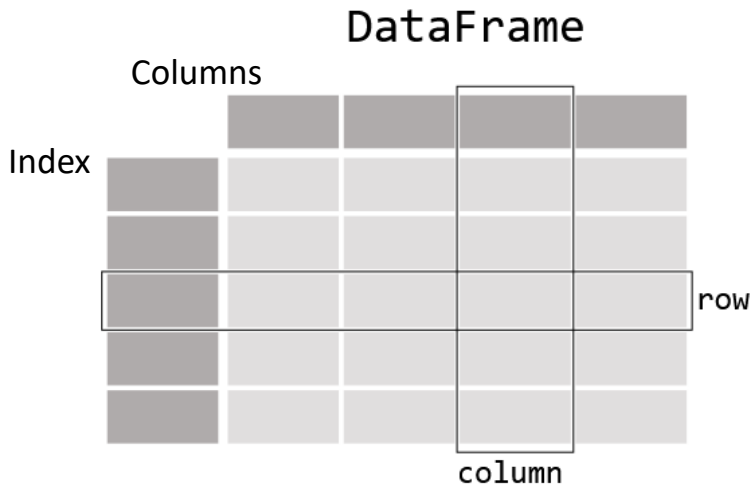
```
pd.concat([data, data2])
```

Join `DataFrames` using a common identifier – `pd.merge()`




```
pd.merge(all_data, text_labels, on='label')
```

Basic DataFrame operations

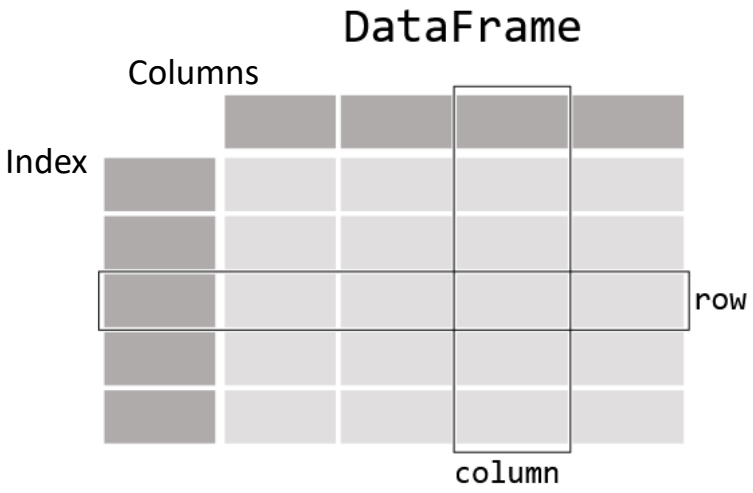


Task	Code	Output
Add column	`data["column_name"] = pd.Series(...)`	

Remove column	`data.drop("column_name", inplace=True)`
	or
	`data = data.drop("column_name")`
		`DataFrame`

Text functions on columns	`data["column_name"].str.len()`
	`.contains()`
	`.lower()`
	`.replace()`
	`.split()`
		`Series`

Basic DataFrame operations



Task	Code	Output

Aggregation functions on columns	`data[“column_name”].mean()`
	`.median()`
	`.std()`
	`.min()`
	`.max()`
		`float`

Remove duplicates	`data.drop_duplicates(“column_name”, inplace=True)`
	or
	`data = data.drop_duplicates(“column_name”)`
		`DataFrame`

Sorting by column	`data.sort_values(by='column_name', ascending=True)`	`DataFrame`
Sorting with function	`data.sort_values(by='content', key=lambda col: col.str.len())`	`DataFrame`

Report back
Q & A

Summary

Module 1
Basic Data Wrangling with Pandas

- **Jupyter Notebooks** provide a web-based interactive coding environment
- **Pandas** is an open-source Python library for data analysis & management
- It provides two basic data structures: `DataFrame` (two-dimensional) and `Series` (one-dimensional)
- `pd.read_csv()` allows you to import CSV files
- The `shape` attribute and `head()` method help you understand your data
- Pandas offers many methods to select data based on squared brackets:
  `data["column_name"][index_start:index_end]`
- Combine data by concatenation (`pd.concat()`) or via common identifier (`pd.merge()`)
- Basic operations: `.sort_values()` for sorting, `.drop_duplicates()` for dup removal
- Aggregation functions: `.mean()`, `.median()`, `.std()`, …
- Text functions: `.str.len()`, `.str.contains()`, `.str.lower()`, `.str.replace()`, …

Outlook for Tomorrow

Module 2
Text Pre-Processing Basics

Understand:

what NLP is good for
why text pre-processing is crucial for NLP
what parallel datasets are
what regular expressions are good for

Know:

how to cHaNgE cAsInG
how to remove text parts
how to handle empty values
how to use regular expressions in Python
how to deal with parallel datasets and their challenges

Join the Mbaza NLP Community!

WhatsApp

https://chat.whatsapp.com/BRlxzsFiZgsLmK5SBT2XUo

Slack

https://join.slack.com/t/mbazanlpcommunity/shared_invite/zt-19ie5idhj-f0yWfOBgTKzs7VOKCcr_pw

GitHub

https://github.com/MBAZA-NLP

Hugging Face

https://huggingface.co/organizations/mbazaNLP/share/mUKyOkYpSRisRpspbfuwUvoQgWyfdiJYqU