Data Science

A collaboration between SHPE and DSI

How to get involved!

DSI

SHPE

Table of contents

Overview

Setting up your environment for the workshop

Activity 1

Data Grouping and Comparison using Pandas

Activity 2

Data Visualization Techniques using Matplotlib

Activity 3

Data Merging and Personalization

Next Week

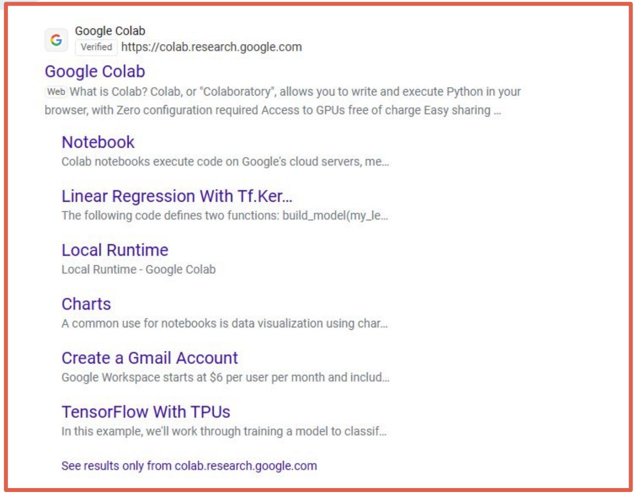
Brief overview of what to expect in next week's follow up

Overview

Setting up your Environment

Follow the next steps to get started!

Step by Step



Google Colab
Verified https://colab.research.google.com

Google Colab
Web What is Colab? Colab, or "Colaboratory", allows you to write and execute Python in your browser, with Zero configuration required Access to GPUs free of charge Easy sharing ...

Notebook
Colab notebooks execute code on Google's cloud servers, me...

Linear Regression With Tf.Ker...
The following code defines two functions: build_model(my_le...

Local Runtime
Local Runtime - Google Colab

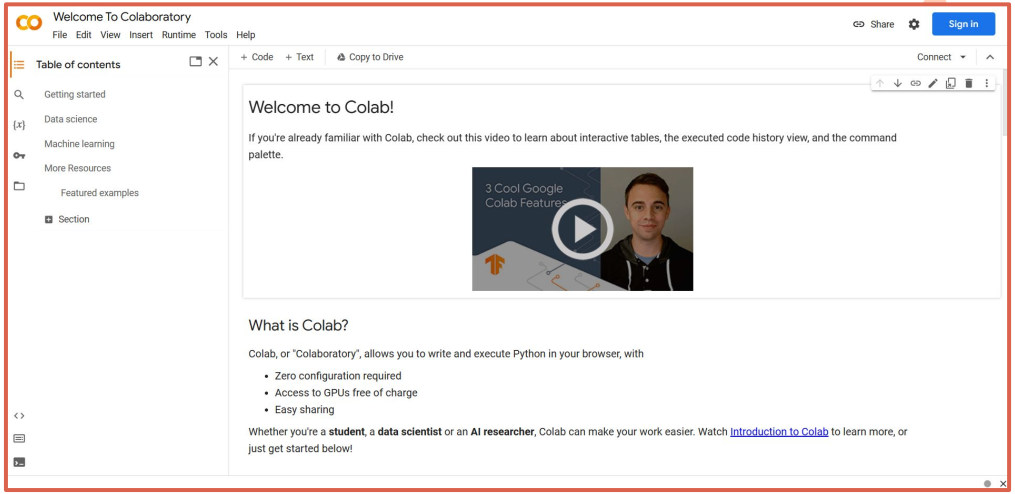
Charts
A common use for notebooks is data visualization using char...

Create a Gmail Account
Google Workspace starts at $6 per user per month and includ...

TensorFlow With TPUs
In this example, we'll work through training a model to classif...

See results only from colab.research.google.com

Search **"Google Collab"** and click on first link



Welcome to Colab!

If you're already familiar with Colab, check out this video to learn about interactive tables, the executed code history view, and the command palette.

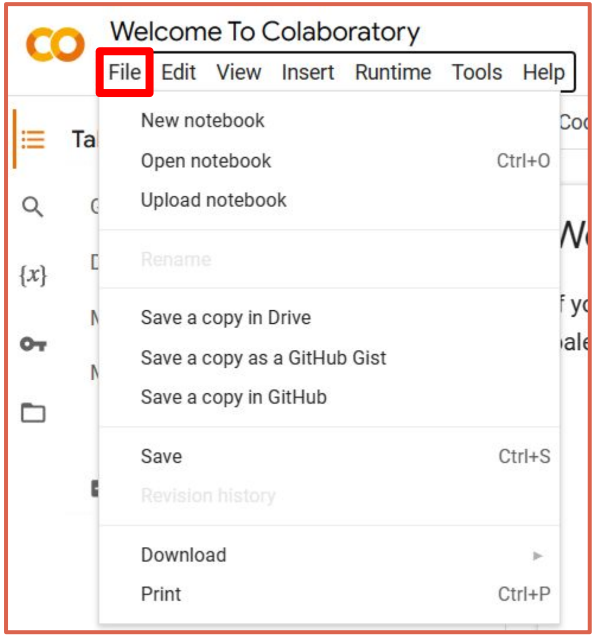
What is Colab?

Colab, or "Colaboratory", allows you to write and execute Python in your browser, with

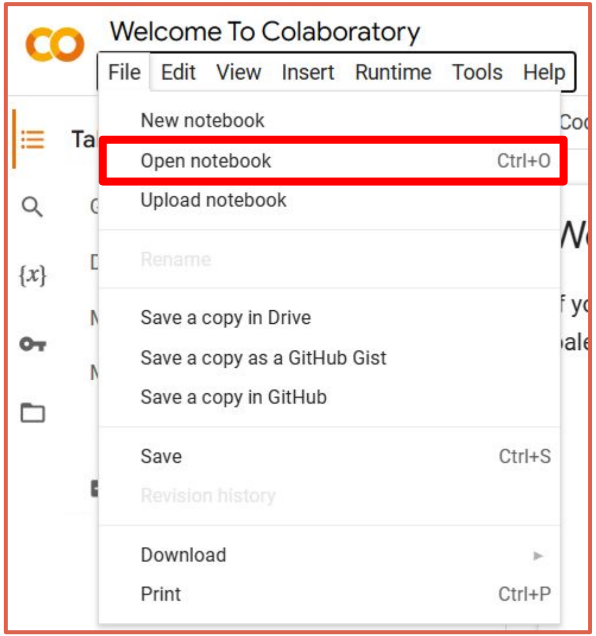
- Zero configuration required
- Access to GPUs free of charge
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch Introduction to Colab to learn more, or just get started below!

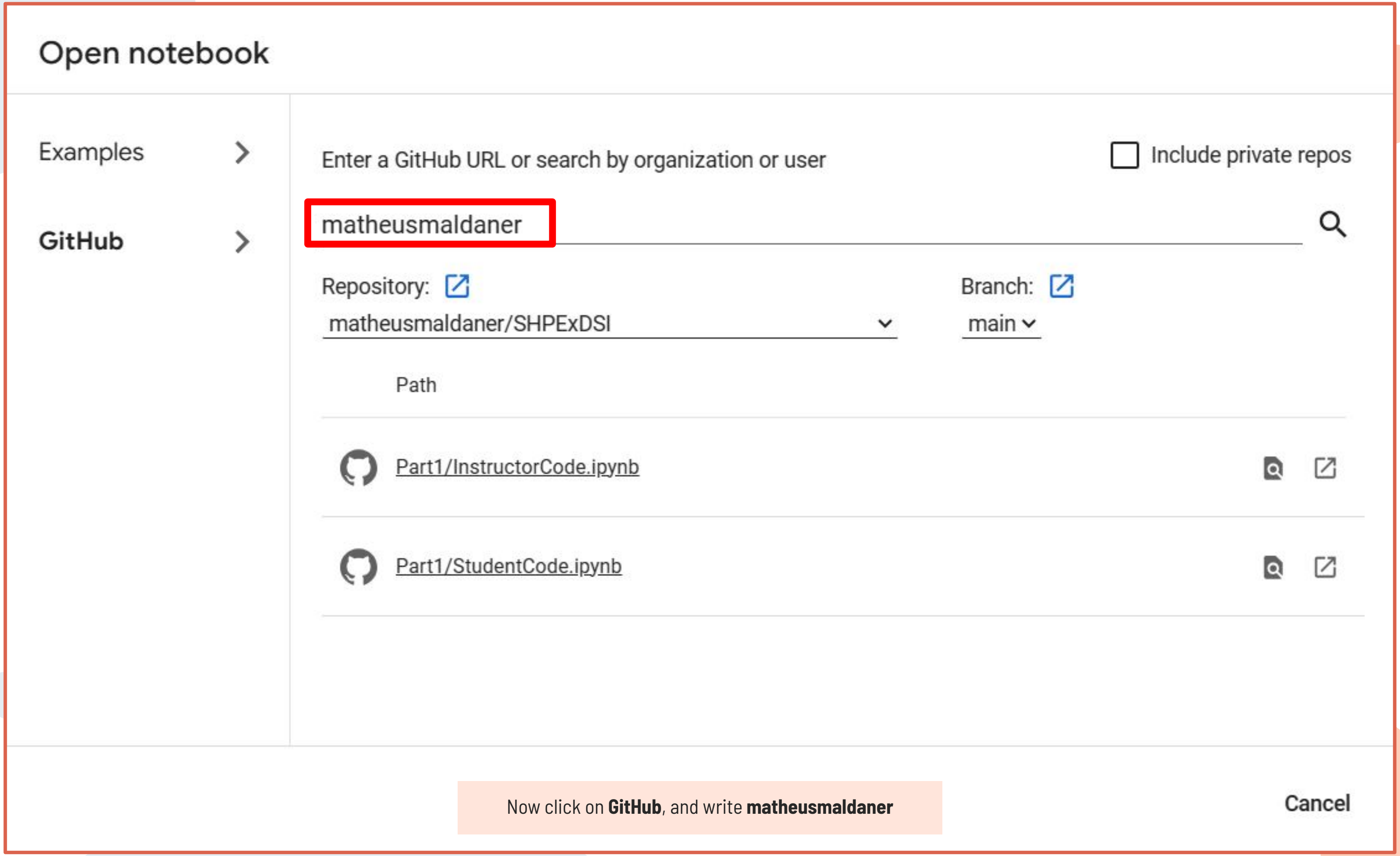
Your screen should look like this



Next click on **File**, on the upper left of the screen



Then click on **Open notebook**



Enter a GitHub URL or search by organization or user

□ Include private repos

matheusmaldaner

Repository: matheusmaldaner/SHPExDSI

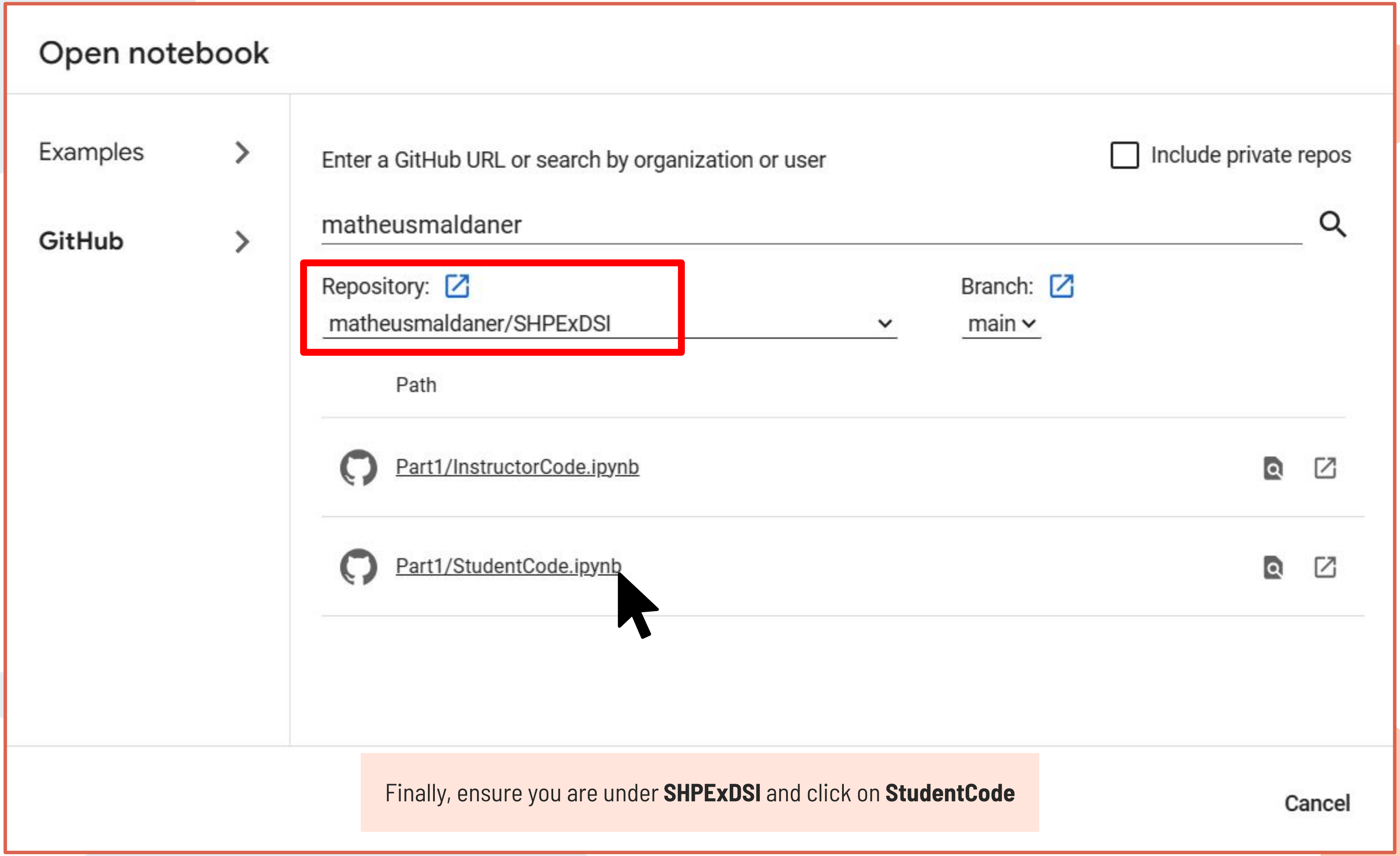
Branch: main

Path

Part1/InstructorCode.ipynb

Part1/StudentCode.ipynb

Now click on **GitHub**, and write **matheusmaldaner**



Open notebook

Enter a GitHub URL or search by organization or user

matheusmaldaner

Repository: matheusmaldaner/SHPExDSI

Branch: main

Path

Part1/InstructorCode.ipynb

Part1/StudentCode.ipynb

Finally, ensure you are under **SHPExDSI** and click on **StudentCode**

Workshop Material

Why Python?

Python is Versatile
- Used in web development, data science, artificial intelligence, and more.

Python is Beginner-Friendly
- Readable syntax that resembles English.

Python in Data Science
- The go-to language for data analysis, machine learning, and scientific computing.

Python is Open Source
- Free to use and distribute, even for commercial purposes.

About Pandas

Essential for Data Handling
- Optimized for performance in data manipulation and analysis, especially with tabular data.

Simplifies Data Analysis
- Offers intuitive data structures and functions for complex tasks like merging, pivoting, and slicing.

Pandas in Data Science
- Critical tool for data preprocessing, cleaning, and analysis in Python-based data science workflows.

Easy Data Exploration
- Includes tools for summary statistics and can be used with other libraries for data visualization.

Other Libraries

Matplotlib
- Plotting library for creating static, interactive, and animated visualizations in Python.

Seaborn
- Based on Matplotlib, seaborn offers a higher-level interface for creating pretty statistical graphics.

Numpy
- Package for scientific computing in Python, with a collection of mathematical functions.

Tensorflow/Pytorch
- Machine learning libraries used for numerical computation and building neural networks

Package = Library

Activity #1

How to manipulate data using Pandas.

Learn to group data and perform basic calculations

Grouping Data with Pandas

GroupBy:

- Grouping is essentially organizing data into categories based on some criteria.
- **groupby( )** is a powerful method in Pandas for grouping data for analysis.

Syntax:

- **DataFrame.groupby(columns)** where **columns** are the attributes you want to group by.
- The result of a **groupby( )** is not a DataFrame, but a GroupBy object with information about the groups.

Functions:

- Applying aggregation functions like **size( ), count( ), sum( ),** to groups to get meaningful insights.
- **variable_name.size( )** will return the size of your grouped columns

Aggregating and Sorting Data

Understanding Aggregation:

- Process of turning the values of a dataset (or a subset of it) into one single value.
- Explain how **size()** calculates the number of entries in each group.

Resetting Index:

- **reset_index()** function and how it transforms the GroupBy object back into a usable DataFrame.
- Naming the aggregation result using **reset_index(name='count')**.

Sorting Data:

- **sort_values()** method to sort data, with parameters like **by** for column name and **ascending=False** for descending order.

Filtering Data:

- Filtering data to focus on recent years **(DataFrame[DataFrame['release_year'] >= 2013]).**

Activity #2

Data Visualization Techniques

Visualizing data patterns using Python's Matplotlib library.

Data Visualization with Matplotlib

Matplotlib:

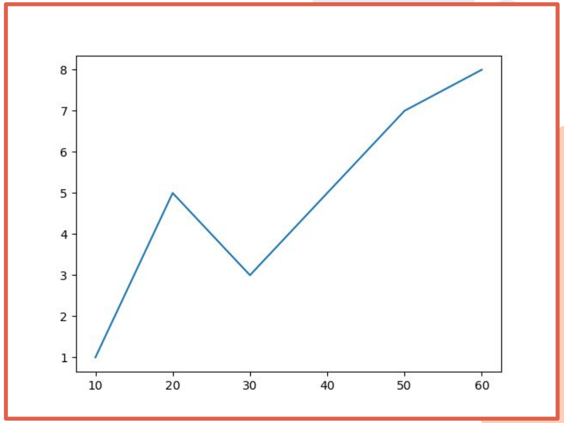
- Comprehensive library for creating static, animated, and interactive visualizations in Python.
- Widely used in the industry and academia for its robustness and versatility.

Basic Plotting:

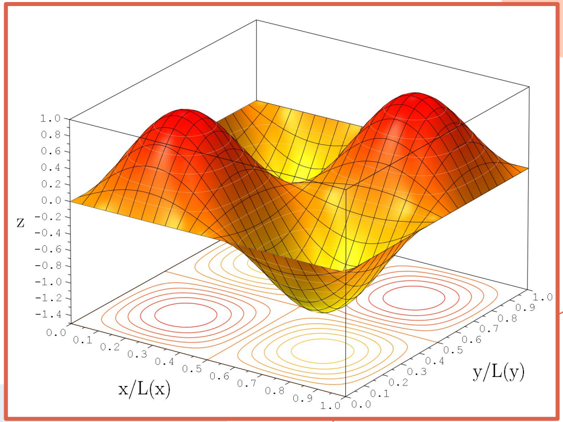
- How to import Matplotlib's Pyplot module with **import matplotlib.pyplot as plt**.
- Basic plotting function **plot()** and introduce other types such as bar for bar charts.

Plot Customization:

- Options like **xlabel**, **ylabel**, **title**, and **legend** to enhance the readability of charts.



Simple 2d graph



Complex 3d graph

Unstacking:

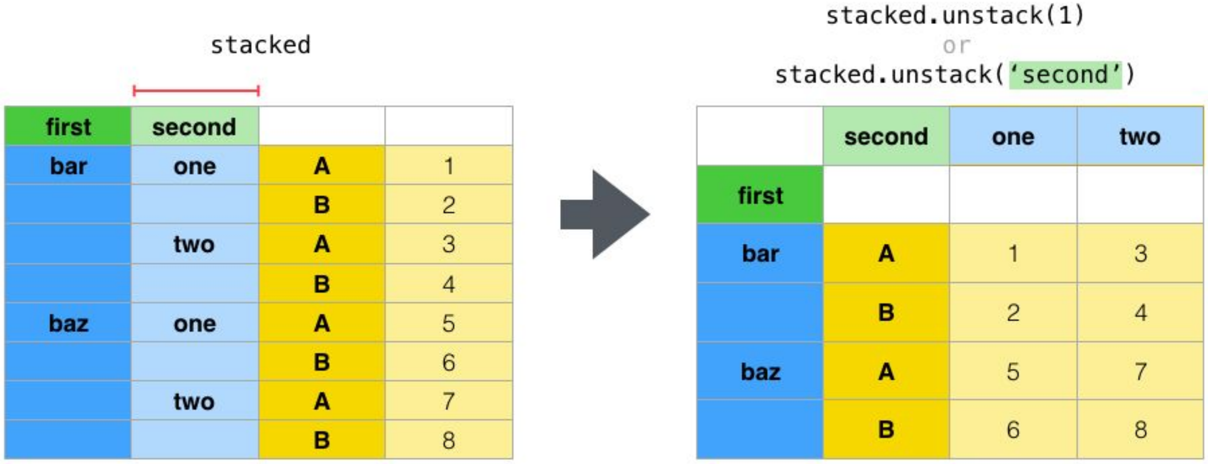
- **unstack()** function in Pandas and how it reshapes the data, turning an index level into a column,
- Useful for preparing data for plotting.

Creating Pivot Tables:

- Unstacked data creates a pivot table that can help in comparing different categories side by side.

Plotting Bar Charts:

- Plotting the unstacked data with **kind='bar'** to create a bar chart to compare categorical data



stacked

first	second		
bar	one	A	1
		B	2
	two	A	3
		B	4
baz	one	A	5
		B	6
	two	A	7
		B	8

```
stacked.unstack(1)
or
stacked.unstack(‘second’)
```

	second	one	two
first			
bar	A	1	3
	B	2	4
baz	A	5	7
	B	6	8

Activity #3

Data Merging and Personalization

More data manipulation techniques and merging different datasets.

Merging DataFrames with Pandas

Introduction to Merging:

- Merging combines two datasets based on a common key.
- Used in data science for enriching datasets and preparing them for analysis.

Pandas Merge Function:

- **merge()** function in Pandas with parameters such as **on** and **how**

Types of Joins:

- Different ways to merge the data frame: **inner**, **outer**, ***left***, and **right**.

```
DataFrame.merge(right, how='left', on=None, left_on=None,
right_on=None, left_index=False, right_index=False, sort=False,
suffixes=('_x', '_y'), copy=None, indicator=False,
validate=None)
```

Pandas dataframe merge syntax

Personalization Through Filtering

Understanding Filtering:

- Filtering allows us to select data based on criteria.

Implementing User Preferences:

- Applying conditions to filter data based on user inputs such as favorite genres or actors.

Iterating Over DataFrames:

- Loops with **iterrows( )** to iterate over DataFrame rows for more complex filtering.

Thank You!