

TP1 (Python Version)

INF8808 : Data Visualisation

Department of computer and software engineering



POLYTECHNIQUE MONTRÉAL

Author : Olivia Gélinas

Teacher's Assistant : Hellen

Objectives

The goal of this practical work is to introduce the web development technologies which will be used throughout this semester. These technologies include HTML, CSS, and Python, as well as the data visualisation libraries Dash and Plotly. These technologies will help prepare us for the next TPs, in which we will also explore Plotly Express.

Please note that this introductory lab **is not evaluated**.

Before beginning, we recommend you have completed the following readings and practice exercises :

- Creating and Updating Figures

<https://plotly.com/python/creating-and-updating-figures/>

- Graph Objects

<https://plotly.com/python/graph-objects/>

Readings :

- "Introducing Dash" essay

<https://medium.com/plotly/introducing-dash-5ecf7191b503>

- Dash HTML Components

<https://dash.plotly.com/dash-html-components>

Exercises : TP1 exercises : 1, 2, 3, 4, 5

Introduction

Throughout this semester, we will write code in Python using the Dash, Plotly, and Plotly Express libraries to accomplish various data visualisations.

Plotly and Plotly Express are Python libraries for making interactive web-based graphs. They are declarative libraries abstracting D3.js and usable in Python code. In this course, we will use Python version **3.8**.

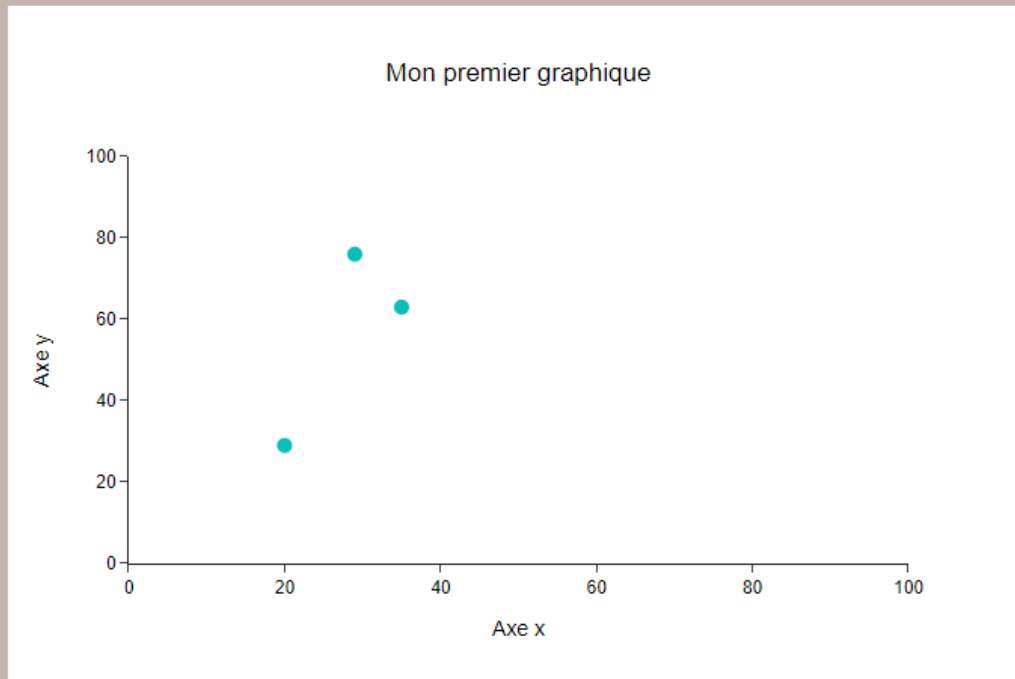
In the context of this TP, we recommend focusing on how Plotly may be used to declare graphs. As practice, we will start from Plotly's graph objects. Also make sure to explore how Dash is used to generate HTML elements and add interactivity to the web page. If needed, it is also important to take this time to familiarise yourself with HTML, CSS, and Python in general.

Description

In this work, you will have to complete the code to display a web page containing a data visualization. The webpage must contain a header, a scatter plot, and a panel with some text and an update button. The scatter plot displays randomly generated data, which is updatable by clicking on the update button. Figure 1 illustrates the desired result.

TP1

Bienvenue au cours INF8808 : Visualisation de données.



Il y a : 3 points

Actualiser

Figure 1: The scatter plot with 3 random data points

File Structure

In this TP, we provide you with 2 Python files used to accomplish the desired visualization :

- `server.py` :

This file is used to launch the application. You do not need to modify this file.

- `app.py`

This file contains all the code related to the chart.

Before beginning, take a look at each function's code and comments to understand how they work. To complete this work, you will need to fill the various `TODO` sections in the `app.py` file. The comments in the code explain in more detail the steps to take.

In the next subsections, we will describe in more detail the various features of the web page.

Header

The webpage should have a header at the top displaying a title and a short description welcoming you to the course.

Data

The data to be displayed must consist of randomly generated coordinates. There should be a random number of data points between 1 and 10 (inclusive). Each point should contain randomly generated x and y coordinates consisting of an integer between 1 and 99 (inclusive).

Scatter plot

The scatter plot should display the randomly generated data. It must have a title and labeled x and y axes. The marks used in the scatter plot are SVG circles with the color and radius as indicated in the code.

Information text

The information text below the scatter plot must display the number of data points currently being displayed. The number itself should be displayed in bold.

Update button

When pressed, the update button makes the random data regenerate. The scatter plot and information text are then also updated to reflect the newly generated data.

Submission

As mentioned at the beginning of this document, this practical work **is not evaluated**. Thus, no submission is required.