

# Configuration Manual

MSc Research Project  
Data Analytics

Stefano Leone  
Student ID: 20198019

School of Computing  
National College of Ireland

Supervisor: Jorge Basilio

**National College of Ireland**  
**MSc Project Submission Sheet**  
**School of Computing**



**Student Name:** Mr. Stefano Leone  
**Student ID:** 20198019  
**Programme:** Data Analytics **Year:** 1  
**Module:** MSc Research Project  
**Lecturer:** Jorge Basilio  
**Submission Due Date:** 2021-09-23  
**Project Title:** Tracking Error of Passive Equity Funds. Data Analysis Using Morningstar Financial Data  
**Word Count:** 630 **Page Count:** 4

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:** Stefano Leone

**Date:** 2021-09-21

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
<b>Attach a Moodle submission receipt of the online project submission,</b> to each project (including multiple copies).	<input type="checkbox"/>
<b>You must ensure that you retain a HARD COPY of the project,</b> both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

<b>Office Use Only</b>	
Signature:	
Date:	
Penalty Applied (if applicable):	

# Configuration Manual

Stefano Leone  
Student ID: 20198019

The tools installed and configured for the MSc project “*Tracking Error of Passive Equity Funds. Data Analysis Using Morningstar Financial Data*” can be easily replicated following the two sections for the ETL and Python setup.

## 1 Setting up the ETL environment

The ETL (Extract, Transform, Load) process was performed using the three following tools:

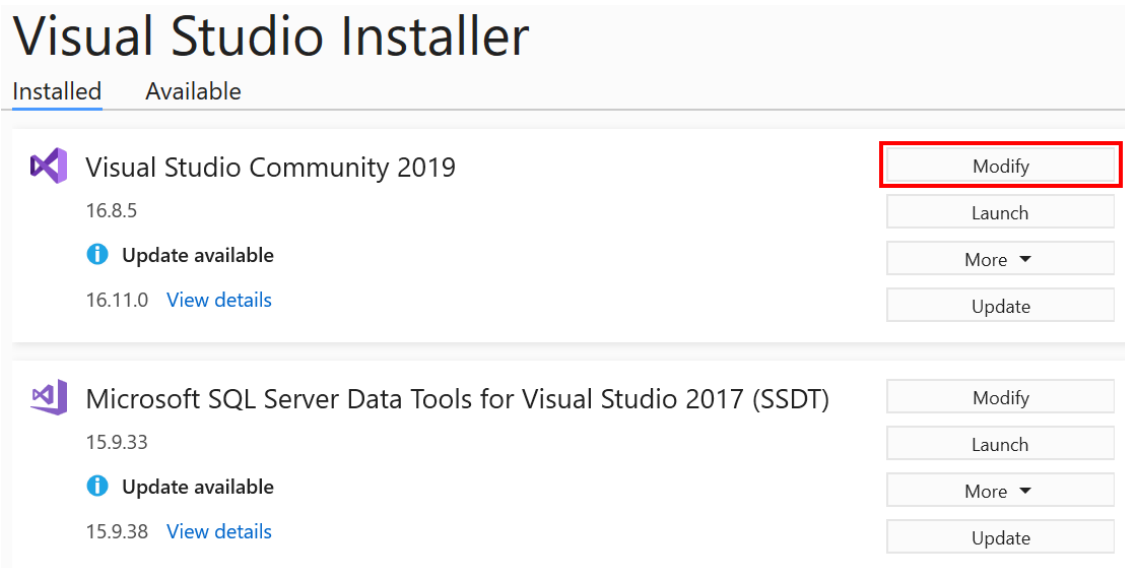
- SQL Server Management Studio (SSMS): the Microsoft application that allows to create, manipulate, and maintain the database where the funds data is stored;
- SQL Server Integration Services (SSIS): the Microsoft package that permits to generate an Integration Services Project, where the user can build complex ad-hoc solutions for any data cleaning task;
- Visual Studio: the interface application that creates and opens the Integration Services Project in which the ETL can be tested and finalized.

The three applications above can all be downloaded from the “SQL Server Data Tools (SSDT)” page on Microsoft<sup>[1]</sup>, taking into account that all the following tools need to be installed from the “*Visual Studio 2017 (SSDT)*” instance:



Figure 1: Required selection in the Visual Studio instance

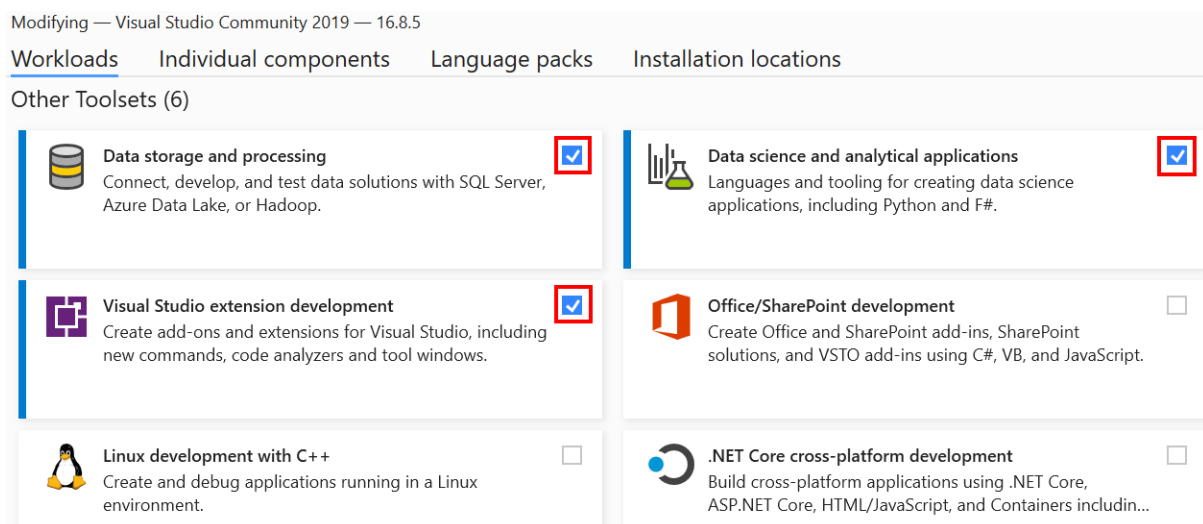
The Visual Studio Installer also allows to select specific components that will be available for the ETL. The components can be selected in the “Modify” button of the “Visual Studio Community 2019” application:



**Figure 2: Modification on Visual Studio components**

The ETL process used in the MSc project “*Tracking Error of Passive Equity Funds. Data Analysis Using Morningstar Financial Data*” requires only the database connections, the Python integrations, and the Visual Studio extension.

The Visual Studio components to be selected are the ones highlighted in the image below:



**Figure 3: Required selection of Visual Studio components**

The files with the Integration Services project have been attached to the “ICT Solution Artefact” zipped submission and can be re-opened and re-run within the Visual Studio environment, after downloading the data uploaded on Kaggle at the URL below:

[www.kaggle.com/dataset/71dd01a5303c3fb1a3894deca254572e91b6fc48517a261580a476a272417088](https://www.kaggle.com/dataset/71dd01a5303c3fb1a3894deca254572e91b6fc48517a261580a476a272417088)

## 2 Installing Python and Jupyter Notebook

Python is often the cornerstone of data analytics works, and the MSc project “*Tracking Error of Passive Equity Funds. Data Analysis Using Morningstar Financial Data*” is not an exception.

Python 3 is required to be installed for two reasons:

- Running the scraping scripts within the ELT process, which permit to collect the list of passive funds, their financial information, and their prices along the ones of the indices they track;
- Performing the EDA and the tuning of machine learning models on the cleaned data. Both tasks are completed in the Jupyter Notebook environment, which can be created by installing the *jupyter* library.

The application can be installed from the official Python website<sup>[2]</sup>, and any version 3.x would work in order to replicate the steps completed for this project.

The Python installation does not need specific guidance, as the default installation is sufficient:



**Figure 4: Default Python 3.x installation**

Once the Python 3.x installation is completed, the required libraries below also need to be installed using the *pip* command<sup>[3]</sup>:

- matplotlib
- numpy
- pandas
- seaborn
- sklearn
- xgboost
- xlrd

After setting up the Python application, the Jupyter Notebook environment can be opened using the *command prompt*<sup>[4]</sup> – *cmd* application in the search box on the Windows taskbar – changing the current directory to the Python executable path (using the *cd* command), and type *jupyter notebook* to launch Jupyter Notebook.

The steps to take in the command prompt are displayed in the image below, which shows also that the *jupyter* library is already installed in Python:

```
Command Prompt - jupyter notebook
Microsoft Windows [Version 10.0.19043.1165]
(c) Microsoft Corporation. All rights reserved.

C:\Users\steee>cd C:\Program Files\Python39\Scripts

C:\Program Files\Python39\Scripts>pip install jupyter
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: jupyter in c:\program files\python39\lib\site-packages (1.0.0)
Requirement already satisfied: qtconsole in c:\program files\python39\lib\site-packages (from jupyter) (5.0.2)
Requirement already satisfied: ipykernel in c:\program files\python39\lib\site-packages (from jupyter) (5.4.3)
Requirement already satisfied: ipywidgets in c:\program files\python39\lib\site-packages (from jupyter) (7.6.3)
Requirement already satisfied: nbconvert in c:\program files\python39\lib\site-packages (from jupyter) (6.0.7)
Requirement already satisfied: notebook in c:\program files\python39\lib\site-packages (from jupyter) (6.2.0)
Requirement already satisfied: jupyter-console in c:\program files\python39\lib\site-packages (from jupyter) (6.2.0)
Requirement already satisfied: ipython>=5.0.0 in c:\program files\python39\lib\site-packages (from ipykernel->jupyter) (7.20.0)
.....
.....

C:\Program Files\Python39\Scripts>jupyter notebook
[W 11:11:56.452 NotebookApp] Terminals not available (error was No module named 'winpty.cygwinpty')
[I 11:11:57.202 NotebookApp] Serving notebooks from local directory: C:\Program Files\Python39\Scripts
[I 11:11:57.202 NotebookApp] Jupyter Notebook 6.2.0 is running at:
[I 11:11:57.217 NotebookApp] http://localhost:8888/?token=20cdc42e3683776a4df9e96fa96c6476383520f7b50c47df
[I 11:11:57.217 NotebookApp] or http://127.0.0.1:8888/?token=20cdc42e3683776a4df9e96fa96c6476383520f7b50c47df
[I 11:11:57.217 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 11:11:57.467 NotebookApp]

To access the notebook, open this file in a browser:
file:///C:/Users/steee/AppData/Roaming/jupyter/runtime/nbserver-13624-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=20cdc42e3683776a4df9e96fa96c6476383520f7b50c47df
or http://127.0.0.1:8888/?token=20cdc42e3683776a4df9e96fa96c6476383520f7b50c47df
```

Figure 5: Opening Jupyter Notebook from the command prompt (cmd)

The Python scripts for scraping (in *.py* format) and for the Jupyter Notebook environment (in *.ipynb* format) have been attached to the “ICT Solution Artefact” zipped submission and can be simply re-opened and re-run in any Python IDE and in Jupyter Notebook respectively.

## References

- [1] Microsoft (2020) *Download SQL Server Data Tools (SSDT) for Visual Studio*. Available at: <https://docs.microsoft.com/en-us/sql/ssdt/download-sql-server-data-tools-ssdt?view=sql-server-ver15> [Accessed 10 August 2021].
- [2] Python community (2021) *Download the latest version for Windows*. Available at: <https://www.python.org/downloads/> [Accessed 10 August 2021].
- [3] Python community (2021) *Installing Packages*. Available at: <https://packaging.python.org/tutorials/installing-packages/> [Accessed 10 August 2021].
- [4] Tim Fisher (2020) *Command Prompt: What It Is and How to Use It*. Available at: <https://www.lifewire.com/command-prompt-2625840> [Accessed 10 August 2021].