**Week #1 – from 13.7 to 17.7**



# Expected Learning Outcomes for the week

* Technical knowledge
  + Python & Anaconda Environments - essentials python libraries setting up & testing
  + Python IDE - working with jupyter notebook & Google Colab
  + SQL Environment- setting up SQL properly in python & working with MySQL
  + Power BI Environment- setting up & working in power BI desktop
  + Basic knowledge of Kaggle & Zindi - accounts opening & features exploration
  + GitHub version control - account setting up & features exploration
* Career knowledge
  + Starting to explore what DS careers are possible
  + Looking at getting your CV ready
  + What do we mean by profile and portfolio
* Attitudes
  + Curiousity
  + Self-Teaching
  + How to communicate results
  + Students will learn how datasets can be used to tell stories with real-world implications as opposed to being merely abstract, quantitative sets of information.

# How this week relates to previous weeks’ content

* This is the first week of the 12 weeks training. During pre-assessment week we went quickly through all working environments & basics data science workflow with the following purposes:
  + to give students the opportunity to understand the format and intensity of the 10 Academy program.
  + to help us understand who has the time and energy to follow our intensive program
  + to help us understand who has the required knowledge and skill
  + to help us test our system

This week will focus on powerful environment setting-up like python - anaconda - jupyter - colab - SQL- powerBI & more.

# Implementation

Context

What is the purpose of this week?

* to give students the opportunity to set up their working environment properly
* to guide student on how to test all their programming environments & softwares
* to introduce learners to 10 academy weekly format & workflow
* to introduce learners to the data science problem solving pipeline

With this in mind, the context of this week is to have a powerful working environment including most of the data science tools like python, SQL, Power BI, Tableau. This week’s training can be considered as a first week working in a company as a Data Scientist. You are expected to understand everything related to your working environment in depth.

We have a number of tutorials, eight in total, to help you test your setup and ensure you can acquire data, process data, and communicate the resulting insights. These tutorials come with some tasks for your action. See them as a guideline to test all your environments and help you get comfortable with your work. You’re not expected to submit these tasks but you are required to attempt all quizzes to the best of your knowledge and skill.

We have one challenge per week that you are expected to work on the whole week, and submit by the beginning of the following week. You are required to submit these challenges. Your performance in these challenges will make the bulk of your 10 Academy portfolio, that we will share with potential employers.

Weekly timetable

Daily (all times given below are in a 24hr format in the German/Rwandan time zone. Adjust these times for your location. Add 1hr if you are in Ethiopia & Kenya; subtract 1hr if you are in Nigeria; subtract 2hrs if you are in Ghana)

* 10.30-10.45 AM daily briefing and standup - all the necessary information about the day schedule and what is expected from you will be communicated here. The recorded videos are shared via youtube link. You will have a chance to share your progress, roadblocks and plans in this meeting.
* 10.45-12hr self learning - self learn by reading the references below, but feel free to explore the web to get concrete knowledge around the topics mentioned in the learning outcome section above. This is also a good time to work on your weekly challenge and the small tasks contained in the tutorials.
* 12-13hr Tutorial I - GMeet call (recommended) or slack attendance is mandatory. If your internet connection is weak and you are unable to join the GMeetcall, you can participate in this session via Slack. Write your opinions and questions to the respective Slack channel. You should follow the tutorial from the pre-recorded video you will find when you download the daily folder.
* 13-15hr practices and work on tasks & weekly challenges - develop deeper understanding/skill of the topics covered in Tutorial I. Work on challenges/tasks given for that day or the ones you didn’t finish from the previous days & apply your knowledge on your weekly challenge
* 15-15.30hr Tutorial II - attend tutorial sessions live or via pre-recorded video. Use slack to ask questions and interact with the tutors and mentors.
* 15.30-17.30hr - develop deeper understanding/skill of the topics covered in Tutorial II. Work on challenges/tasks given for that day or the ones you didn’t finish from the previous days and apply your knowledge to solve your weekly challenge
* 17.30hr - Attempt quiz of the day

# Important locations

1. Google meet link for all video sessions (daily briefs, standups, and tutorial sessions). To join the video meeting, click this link:   
   [**https://meet.google.com/xvq-aixz-vws**](https://meet.google.com/xvq-aixz-vws)
2. Youtube link for all past videos  
   <https://www.youtube.com/channel/UCDwr854iwfpjDk1ESK5ofvQ>
3. Website for weekly program  
   <https://sites.google.com/10academy.org/learners/announcements>
4. Google drive folders for each day

|  |  |
| --- | --- |
| Monday | https://drive.google.com/drive/folders/1eP2EqSzcLYJOr2RmOjKjLdsWAD6jWK8a?usp=sharing |
| Tuesday | https://drive.google.com/drive/folders/1sQqKTBJUJ8qegDZgND6zuf1fQwe-A1lR?usp=sharing |
| Wednesday | https://drive.google.com/drive/folders/1hfszIDuvS\_dxM8162gMyatfKbv7hFBjU?usp=sharing |
| Thursday | https://drive.google.com/drive/folders/19\_ZYVKM93AHQvGZQURSvp-oZOHrhS20\_?usp=sharing |
| Friday | https://drive.google.com/drive/folders/1KLYbh9Rw-8I50goGZRyAKYSk2cckoPZo?usp=sharing |

# Expectations

1. By Day 3 of the week (Wednesday) you should be able to yes to all of the following  
   * Have a clear understanding of the week plan - use slack to communicate with tutors, mentors and admins
   * Have python, jupyter notebook and relevant packages installed and tested
   * Have a basic understanding some commands in terminal
   * Have an active github account, have forked one of our package, and you have used git clone, git add, git commit, and git pull & push
   * Is comfortable working with Google Colab
   * Obtained Twitter developer API access - you should be able to download twitter data from python
   * Have the necessary microsoft Power BI account and package to run a BI application
   * Know your relevant people (tutors, instructors, sys admins, your contact person)
   * Know where the data and course material are located
   * Have submitted the quizzes
2. By Day 5 of the week (Friday):
   * Sorted out all the major challenges you faced in the tutorials and challenge
   * Got comfortable with the 10 Academy weekly workflow - you have reported any challenge you faced otherwise
   * discussion on the tutorials & weekly challenge
   * Submitted all of the quizzes given throughout the week
   * Is clear with what is next - how and when you will be communicated on the next steps

# Materials & Resources

## Data

The most important dataset is the one related to your weekly challenge and this week is about testing your environments with twitter data. Everything about your data is described in the weekly challenge.

All other dataset if any will be small datasets to test your working environments

## Codes

About tutorials files - As we are progressing in the week, you will get them on a daily basis in the day folder together with your quizzes. The tutorial could contain themselves some tasks to direct you through how to test things in your environments.

Weekly Challenge  
You have the description of your weekly challenge in the weekly challenge : [week 1 document](https://docs.google.com/document/d/1DW0INB9iPAvU8XRHQsDkG3O7DFeRkPpcXWC1egJ81YM/edit?usp=sharing)

# Feedback System

There are a number of ways you get feedback from us, and the community in general, about your performance.

1. On Friday of the week, we ask you to submit the tutorials tasks you feel you did well. We will ask other students to comment on your work - complements and suggestions are then forward to you.
2. We will have a private call with you on a monthly basis. This will be the official channel to receive feedback from our tutors and instructors.
3. We will send you the comments and suggestions our tutors and instructors put on your weekly challenge submission.
4. Quick feedback to main roadblocks are addressed during the morning standup sessions.

# Assignments

There will be quizzes, containing 5-10 questions, at the end of every day. You should answer and submit it before the end of the day.

# Assessment & Grading

This week there are 120 points on offer. These points are distributed on three core categories:

1. 40% (48 points) - answering correctly all the daily quiz questions
2. 50% (60 points) - submission of the weekly challenge to the best of your knowledge and skill, qualifying for an overall pass grading

1. 30% (36 points) - active participation in standup sessions, tutorials, slack channels. This is an equivalent of being a good team in a real job.

## Grading Rubric for Weekly Challenges

* Technical
  + Creativity in approach to solving challenge
  + Answers all questions
  + Bonus points for going beyond the call of duty
  + Excellent commenting and layout of Notebook
* Presentation
  + Demonstrate strong understanding of work and quality of answers
  + Quality of communication
  + Clarity of materials presented and posted
  + Professionalism of materials (well designed, free of typos etc)

## Portfolio Contributions

By the end of this week all learners will get very good knowledge about how to set up a data science working environment including python & SQL, able to work using Google Colab , jupyter notebook & GitHub, build experience analysing twitter data using REST API, basic skill to set up environment to build dashboards.

!--Check your understanding--!

Make sure you are clear with the following points. If you are not sure, ask your contact person.

* The skills and knowledge you should have by the end of the week.
* You understand the problem to be solved as part of your daily tasks and challenges. What is expected from you.
* You know where to find the relevant information (e.g. data, code) for the daily and weekend challenge.
* When and how to submit your solutions.
* How your daily performance is measured, and accumulated.

# References Reading materials and external references

## **Setting up a data science Environment**

Here are links to help you start setting up your data science environment:

* [Setup a Data Science Environment on your Computer](https://www.datacamp.com/community/tutorials/setup-data-science-environment)
* [Setting up your Data Science and AI dev environment in 5 minutes](https://towardsdatascience.com/setting-up-your-data-science-and-ai-dev-environment-in-5-minutes-44804a324713)
* [How to setup a Python Data Science environment — Setting up Anaconda environments for working on data science problems using Python](https://towardsdatascience.com/python-data-science-environment-setting-up-anaconda-environments-for-working-on-data-science-8b4bfbc3cd39)

By the end of this upskilling process you should be able to do easily the following:

* edit Python or R scripts and run them from terminal and jupyter notebooks
* install necessary python packages using pip and conda
* able to create python virtual environments to be able to run some projects in isolation

## Code Editors

## Jupyter Notebook

## To set up your Jupyter Notebook environment in your computer, follow the guide [here](https://towardsdatascience.com/getting-started-with-google-colab-f2fff97f594c).

## Google CoLab Notebook

If you don't have Google CoLab account, please have account and learn the basics working in Jupyter Notebook [here](https://colab.research.google.com/).

## Remote Connection

Remote connection implies a client, which stands for anything that initiates the communication, sends a request to a server, which stands for anything that receives a request, returns a response - see figure below. This process is the most fundamental for any remote connection.

You can learn about the basics of ssh [here](https://help.ubnt.com/hc/en-us/articles/218850057-Intro-to-Networking-How-to-Establish-a-Connection-Using-SSH) and the basics of rsync and scp [here](https://medium.com/@elngovind/understanding-rsync-and-scp-46bcb20791d0).

## Containers

Learn the basics on setting up docker environment [here](https://docker-curriculum.com/) and python virtualenv [here](https://codeburst.io/understanding-python-installation-and-virtualenv-a-friendly-guide-for-beginners-and-2b82859b06ae) and [here](https://towardsdatascience.com/getting-started-with-python-environments-using-conda-32e9f2779307) (with conda).

## Version Control

#### Git Basics

##### Add local folder to git - track a folder with git

$ git init

##### Add new files or changes to files - add files, and commit in a git repo

$ git add \*.c #add also files with \*.c extension

$ git add LICENSE #add a file called LICENSE

##### Commit - describe briefly what has changed

$ git commit -m 'initial project version’

##### Push - now that you have added all files from your local computer and described what the files are or what has changed in those files, then push them to the remote git repository

$git push origin master

"git push" is the command. "origin" is the alias to the web address of your git repository. To see the address you can use the command (-v is for verbose, which allows git to show the remote address)

$git remote -v

"master " is the main branch name. You can have many branches to test different scenarios. If the test is fine you can integrate it with the main at a later point, if not you can discard it without affecting the master copy.

##### Clone - clone a package from git. If you want to try someone's code which is located in git, get the link then clone it to your local computer

$ git clone <https://github.com/libgit2/libgit2>

To learn the basic concepts of Git and have an account at github, follow the guide

* [git - the simple guide](http://rogerdudler.github.io/git-guide/)

## APIs

##### Website vs API

Website is optimal for humans but difficult for computers to understand. [APIs](https://moodle.10academy.org/mod/page/view.php?id=284) provide a platform for information humans can access via websites to be accessible by computers.

**Key terminologies**

* **Server**: A powerful computer that runs an API
* **API**: The "hidden" portion of a website that is meant for computer consumption
* **Client**: A program that exchanges data with a server through an API
* **Protocol**: An “extremely rigid” set of rules that govern how two computers can speak to each other.

##### Computer Protocols

* A computer protocol is an “extremely rigid” set of rules that govern how two computers can speak to each other.

##### "My favourite color is blue"

##### "Blue is my favorite color."

##### For humans (1) and (2) are the same while it is different for computers talking with a certain protocol.

* Any communication between computers require a certain protocol.
* Bluetooth for connecting devices, POP or IMAP for fetching emails. The web main protocol is the Hyper-Text Transfer Protocol –HTTP.
* Most [APIs](https://moodle.10academy.org/mod/page/view.php?id=284) use the HTTP protocol to communicate.
* The Secure SHell (SSH) protocol is a method for secure remote login from one computer to another. It provides several alternative options for strong authentication, and it protects the communications security and integrity with strong encryption.

##### 

##### HTTP “request” and “response”

Methods:

GET - Asks the server to retrieve a resource

POST - Asks the server to create a new resource

PUT - Asks the server to edit an existing resource

DELETE - Asks the server to delete a resource

##### 

##### API Authentication

* Authentication: process of the client proving its identity to the server
* Credentials: secret pieces of info used to prove the client's identity (username, password...)
* Basic Auth: scheme that uses an encoded username and password for credentials. Has full control.
* API Key Auth: scheme that uses a unique key for credentials. Has limited control.

##### Data Exchange

* Once a client makes a successful communication through an API, the server returns a data in a certain format.
* The most common formats found in modern APIs are JSON (JavaScript Object Notation) and XML (Extensible Markup Language).

To learn about the [basics](https://blog.restcase.com/restful-api-authentication-basics/) of protocols API authentication and authorisation click here.

* [Understanding Protocols & API Data exchanges](https://zapier.com/learn/apis/chapter-2-protocols/)
* [RESTful API Authentication Basics](https://blog.restcase.com/restful-api-authentication-basics/)

##### Development accounts

Many established companies like Google, Facebook, Twitter and similar others provide an API access to their systems such that developers can connect to their service within their codes.

For example, if you need to analyse the twitter trend about a particular topic in Africa, you two options

1. Go to twitter, search the topic based on search filter.
2. Write a code using twitter API, download data, do more processing

If you are doing it only once, option 1) may not be too bad. But it becomes difficult if you want to monitor the topic live over a specific time. In that scenario, option 2) is clearly the way to go.

To implement 2) you need to get a developer access from twitter. The login and password you have to login in the web is for you not for your code. To authorise your code, you must get API access key and secrete.

##### Web Reference:

* [Twitter Data Mining: A Guide to Big Data Analytics Using Python](https://www.toptal.com/python/twitter-data-mining-using-python)
* [Get started with Google API](https://developers.google.com/api-client-library/python/start/get_started)

## Data Science Concepts

Start with the following references to learn more about data science concepts.

* [30 common data science terms explained](https://hub.packtpub.com/common-data-science-terms/)
* [What Is Feature Engineering for Machine Learning?](https://medium.com/mindorks/what-is-feature-engineering-for-machine-learning-d8ba3158d97a)
* [A Visual Introduction to Machine Learning](http://www.r2d3.us/visual-intro-to-machine-learning-part-1/)

To learn more about reading and writing different data types in Python, start from these references

1. [Python — Reading and Writing data from Files](https://medium.com/datadriveninvestor/python-reading-and-writing-data-from-files-d3b70441416e)
2. [Image manipulation and processing using Numpy and Scipy](http://scipy-lectures.org/advanced/image_processing/)