ReDI School of Digital Integration

Introduction to Data Analytics Spring 2023

Lecture 1

We use tech to connect human potential and opportunity with dignity & humility

Welcome!

How we work

Lectures
17:30 - 18:45
80% attendance required
Ask questions!

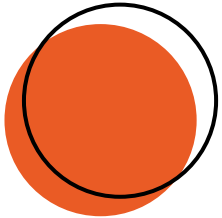
Exercises
19:00 - 20:30
Be active! Make mistakes!

Homeworks
Hand-in until Monday
Individually on Google Classroom

Kahoot Quiz!
Every second week

Communication
Slack! Keep up to date!

Project
Talk to your mentor



Setup Checklist

Am I ready for class?

- Do I have access to Google Classroom?
- Do I have access to the Slack channel #data-analytics-team-4?
- Do I have Kahoot installed?
- Do I know who is in my team?
- Did I complete the Python Installation Guide?

A few words of advice

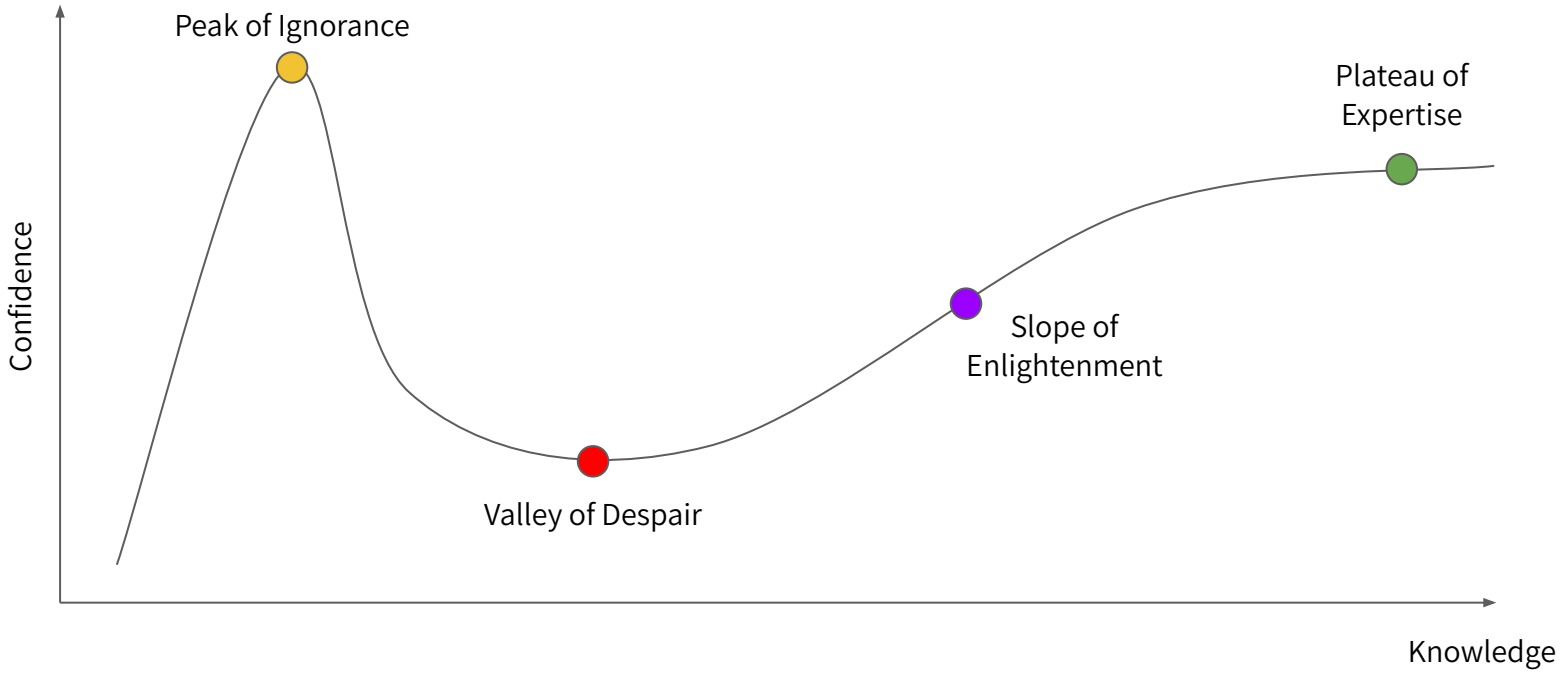
Mantras

Understand your code!

Learn how to Google!

Take a break!

Dunning-Kruger effect

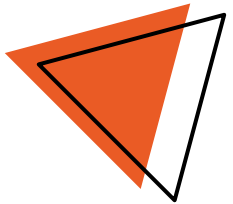


The Essence of the course

The goal of this course is to teach you how to:

"Collect, transform, model, analyze, and visualize a broad range of datasets using Python and its open source libraries, through concepts of Data Analytics."

Important: You are responsible for your own learning!



Let’s get started!

What is Data Analytics?

Purpose

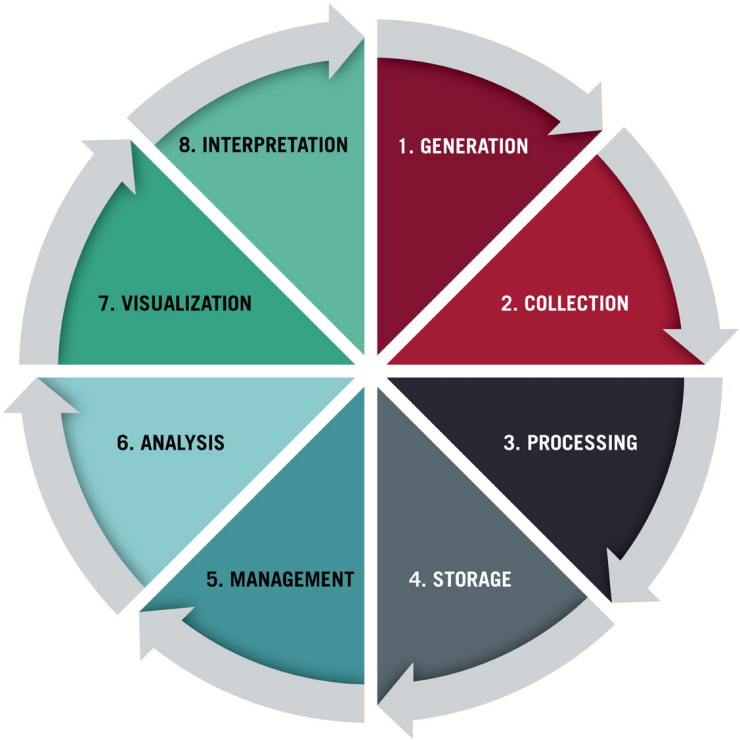
"DATA SCIENCE is the process of building, cleaning, and structuring datasets to analyze and extract meaning.

DATA ANALYTICS refers to the process and practice of analyzing data to answer questions, extract insights, and identify trends."

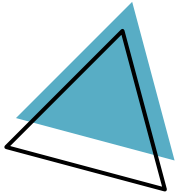
~Harvard Business School Online

Data Ecosystem & Lifecycle

- **Data Ecosystem** describes the tools and setup of an organisation to collect, store, and manage data.
- **Data Lifecycle** describes the journey of data from when it is collected to when it interpreted.

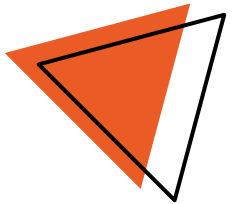


The Programming Tools



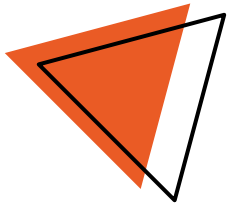
What is Python?

“Python is a programming language that lets you work more quickly and integrate your systems more effectively.



Why Python for Data Analytics?

- Easy to Learn and Use
- Cross-Platform Compatibility (Windows, Mac, Linux, …)
- Great Open Source Community
- Modular With Many Libraries
- Great Data- and Visualisation Tools
- High Demand on the Job Market



Components of Programming in Python

Python language

Used to write the code

```
print('Hello, world!')
```

IDE

The editor in which the code is written

```
Visual Studio Code, Atom, Browser,
Text Editor
```

Console / Terminal

The system tool in which the code is run

```
python main.py
python -m jupyterlab
```

PIP / libraries / packages

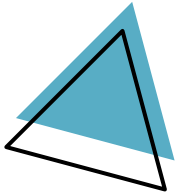
A repository of python software written by the open-source community

```
pip install jupyterlab
```

Python Packages

Math	Data Exploration	Data Visualisation	Environment
NumPy	Pandas	Matplotlib	Jupyter
SciPy	Dask	Plotly	Spyder
Scikit-learn	PySpark	Seaborn	Anaconda
TensorFlow	Altair	Bokeh	Cloud Solutions

Programming Basics



Data Types

Text

str “Hello, world” / “4”

Numeric

int -1 / 0 / 4 / 23

float -1.5 / 0.2 / 5.3

Iterable / Sequence

list [“hello”, “world”]
[4, 2, 10]

tuple (“apple”, “pear”, “pear”)

range range(1,10)

Mapping

dict {
“name”: “John”,
“lastname”: “Doe”,
“age”: 23
}

Unique collection

set { “apple”, “banana”, “pear” }

Boolean

Bool True, False

Null Type

NoneType None

Variables

“Variables are containers for storing data values."

- A Variable has with a name, a type, and a value.
- Writing to a variable is called “assigning to a variable”
- Variables can be reassigned (overwrite)

Examples:

```
name = “Clara”
isStudent = True
age = 23
name = “Emma”
```

Operators

"Operators are used to perform operations on variables and values."

Arithmetic	Assignment	Comparison	Logical	Membership
`3+2`	`x = 3`	`x == y`	`x>3 and x<10`	`1 in [1,2,3]`
`2*3`	`x += 5`	`5 != 3`	`x<3 or x>10`	`2 not in [4,5,3]`
`12%3`	`x *= 2`	`3 > 5`	`x not 5`	`"Clara" not in ["Emma", "Tom"]`
`2**3`	`x /= 3`	`5 <= 3`	`x not 3 && not 5`	`20 in ["20", "25"]`

Functions

“A function is a block of code which only runs when it is called.”

- You can pass data, known as parameters, into a function.
- A function can return data as a result.

Example:

```
print(“Hello, world”)
marks = [65, 71, 68, 74, 61]
print(sum(marks))
```

Break - Then it’s your turn!