Get Started with Python

Course outline:

- → Module 1 – Hello, Python!
- → Module 2 – Function and Conditional statements.
- → Module 3 – Loop and strings.
- → Module 4 – Data structure in Python.
- → Module 5 – Course-2 end-of-course project.

Week 01: Hello, python!

✳ Programming language comparison:

Features by Software	Python	R	Java	C++
Speed	Slower	Depends on configuration and add-ons	Faster	Very fast
Approachability	Easy to learn	Complex	Easy to learn	Complex
Variable	Dynamic	Dynamic	Static	Declarative
Data science focus	Machine learning and automated analysis	Exploratory data analysis and building extensive statistical libraries	Used across projects with open-source assets	Not as widely used but very powerful implementations
Programming Paradigm	Object-oriented	Functional language	Object-oriented	Multi-paradigm (imperative & object-oriented)

✳ **Key-board shotcut for Jupyter notebook:**

→ add one cell Above: Esc + A

→ add one cell below: Esc + B

→ Delete current cell: Esc + dd

→ Run selected cell: Ctrl + Enter.

→ Run selected cell and move next: Ctrl + Enter.

→ Run selected cell and add new cell bellow: Alt + Enter.

✳ **Object-Oriented Programming:** A programming system that is based around objects, which can contain both data and code that manipulate that data.

✳ **Objects:** An instance of a class; A fundamental building block of python.

✳ **Class:** A class is an objects data type that bundles data and functionality together.

✳ **Method:** A function that belongs to a class and typically performs an action or operations.

✳ **Dot Notation:** ~ used to access the methods and attributes that belong to an instance of a class.

✳ **Core Python classes:**

- → Integers
- → Floats
- → Strings
- → Booleans
- → Lists
- → Dictionaries
- → Tuples
- → sets
- → Frozensets
- → Functions
- → Ranges
- → None.

✳ **Attributes:** A value associated with an object or class which is referenced by name using dot notation.

✳ **Naming Conventions:** Consisting guidelines that describe the content, creation time, and version of a file in its name.

✳ **Naming Restrictions:** Rules build into the syntax of the language itself that must be followed.

✳ **Python naming conventions:**

1. keywords are not allowed.
2. Variable only include letters, numbers, and underscore.
3. Must starts with letter or underscore.
4. case-sensitive. | 5. can't use parentheses.

✳ **Type conversion in python:** Two types of conversion.

⇒ **Implicit conversion:** Python automatically convert one data type to another without user involement.

Example: 1 + 3.5 ⇒ 4.5

int → double; double → double

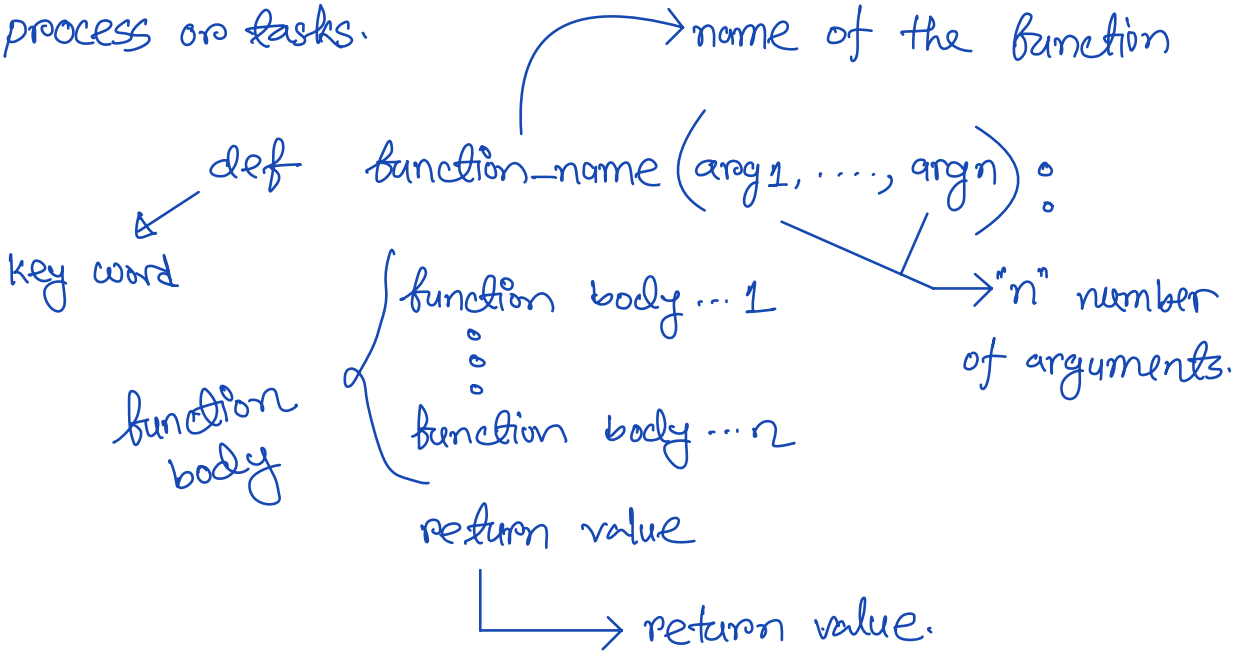
⇒ **Explicit conversion:** Have to manually convert the type of the data.

Example: 2 + int("2") ⇒ 4

↳ type casting.

Week 02: Function and Conditional statements

✳ **Function:** A body of reusable code for performing specific process or tasks.



✳ **Write clean code:**

1. **Maintain modularity:** The ability to write code in separate components that work together and that can be reused for other programs.
2. **Refactoring:** The process of restructuring code while maintaining its original functionality.
3. **Self-documenting code:** Code written in a way that is readable and makes its purpose clear.

4. **Docstring:** A string at the begining of a function's body that summarizes the function's behavior and explains its arguments and return values

* **Branching:** The ability of a program to alter its execution sequence.

Week – 04 : Data structures in Python

* **Data Structure :** collection of data values or objects that contains different data type.

* **Dictionary :**
  key-value pair.
  * keys are only immutable data.
  * dictionaries are un-order.

* **Vectorization :** Enable operation to be performed on multiple components of a data object at the same time.

* **Numpy-array :**
  → arr.dtype – return the data-type of the array.
  → arr.shape – the shape of the array.
  → arr.dim – the dimension of the array.
  → arr.reshape (new-dimension) – reshape the array

Week—05: Course 2 end-of-project scenario

TikTok

Project goal:

The TikTok data team is developing a machine learning model for classifying claims made in videos submitted to the platform.

Background:

TikTok is the leading destination for short-form mobile video. The platform is built to help imaginations thrive. TikTok's mission is to create a place for inclusive, joyful, and authentic content–where people can safely discover, create, and connect.

Scenario:

As a data analyst on TikTok's data team, you'll help by preparing the data needed for the claims classification project. You'll build a dataframe, organize the claims data for the process of exploratory data analysis, and update the team on your progress and insights.

Course 2 tasks:

- Build a dataframe for the TikTok dataset
- Read in data from TikTok csv file
- Display rows within dataframe
- Examine data type of each column
- Gather descriptive statistics
- Visualize the TikTok data in Python
- Report to TikTok's data team through an executive summary

Note: *The story, all names, characters, and incidents portrayed in this project are fictitious. No identification with actual persons (living or deceased) is intended or should be inferred. And, the data shared in this project has been created for pedagogical purposes.*

		like_per_view			comment_per_view			share_per_view		
		count	mean	median	count	mean	median	count	mean	median
claim_status	author_ban_status									
claim	active	6566	0.329542	0.326538	6566	0.001393	0.000776	6566	0.065456	0.049279
	banned	1439	0.345071	0.358909	1439	0.001377	0.000746	1439	0.067893	0.051606
	under review	1603	0.327997	0.320867	1603	0.001367	0.000789	1603	0.065733	0.049967
opinion	active	8817	0.219744	0.218330	8817	0.000517	0.000252	8817	0.043729	0.032405
	banned	196	0.206868	0.198483	196	0.000434	0.000193	196	0.040531	0.030728
	under review	463	0.226394	0.228051	463	0.000536	0.000293	463	0.044472	0.035027