LECTURER: TAI LE QUY

Introduction to Programming with Python

Who are you?

- Name
- Employer
- Position/responsibilities
- Fun Fact
- Previous knowledge? Expectations?

INTRODUCTION TO PROGRAMMING WITH PYTHON
TOPIC OUTLINE

Introduction to Python	1
Variables and Data Types	2
Statements	3
Functions	4
Errors and Exceptions	5

Modules and Packages

6

UNIT 1

Introduction to Python

STUDY GOALS

- Understand why software developers choose Python
- Describe strengths and weaknesses of Python programming language
- Know how to download and install Python
- Explain and use various parts of a Python development environment

EXPLAIN SIMPLY

1. Why do software developers choose Python?
2. What are the strengths and weaknesses of the Python programming language?
3. What are the benefits of JupyterLab?

Overview

Python is an open source programming language. It was created by Guido van Rossum in the late 1980. It is an interpreted programming language, as such its codes are processed at runtime without the need of compiling. Consequently, programs can be run interactively.

Development history

Dec 1989

- First implementation of Python started.

Feb 1991

- First version of Python (0.9.0) was released.

Oct 2000

- Python 2.0 was released. Python 2.7.18 (Apr 2020) was the last in that series.

Dec 2008

- Python 3.0 was released. As of this writing (August 2022), 3.10.6 was the latest release.

Strengths & Weaknesses

Weaknesses

- Python lacks security and warranty
- It is slow compared to other programming language like Java, C++
- It lacks support for mobile application development
- It uses a large amount of memory
- It lacks support for relational databases

Strengths

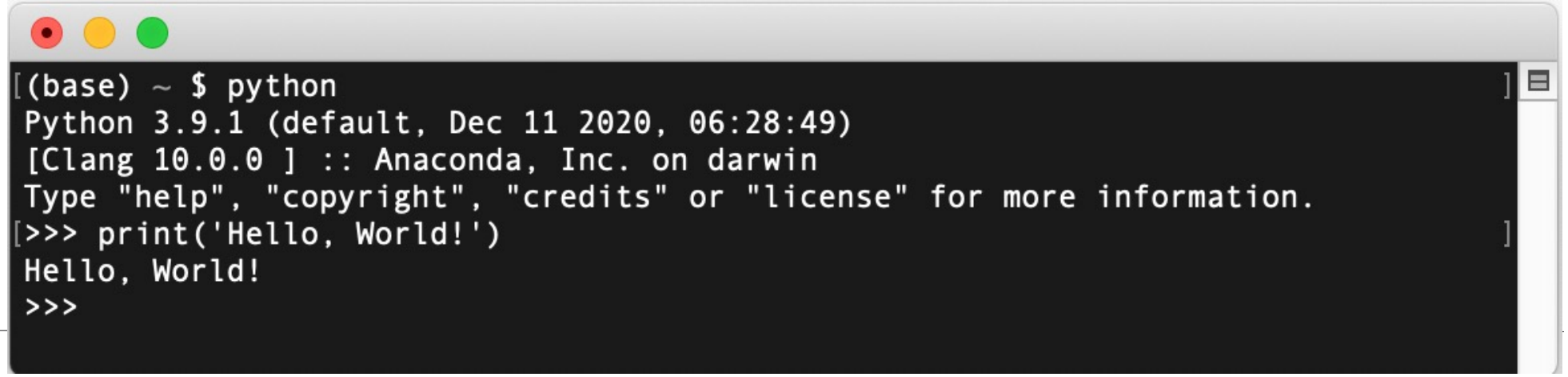
- Python is free and open-source
- It is interpreted and interactive
- It has concise syntax and is easy to learn and write
- It has a large and active developer community
- It has a vast number of libraries, especially for data science
- It is cross-platform compatible with Windows, MacOS and Linux
- It supports web development

INSTALLATION

Resource	There are a number ways to obtain Python e.g., direct from source, via package managers etc. In this course, we focus on using the Anaconda distribution. The advantage of using the Anaconda distribution is that includes some crucial data science tools i.e., NumPy, SciPy and Pandas.
Anaconda distribution	• Go to https://www.anaconda.com/products/distribution • Download the version for your platform e.g., Windows, MacOS or Linux. As of August 2022, the Python version supported by Anaconda is Python 3.9 • Click on the downloaded Anaconda executable and initiate the installation (as of August 2022, the latest Anaconda version is Anaconda 3) • Follow the instructions in the installation window and complete the installation

Testing

Windows	MacOS
• Click the start button, “Anaconda” and then “Anaconda Prompt” • Type `python` at the prompt to start the Python interpreter • Type `print('Hello, World!')`	• Open Launchpad. Click on “Other” and then “Terminal” • Type `python` at the prompt to start the Python interpreter • Type `print('Hello, World!')`



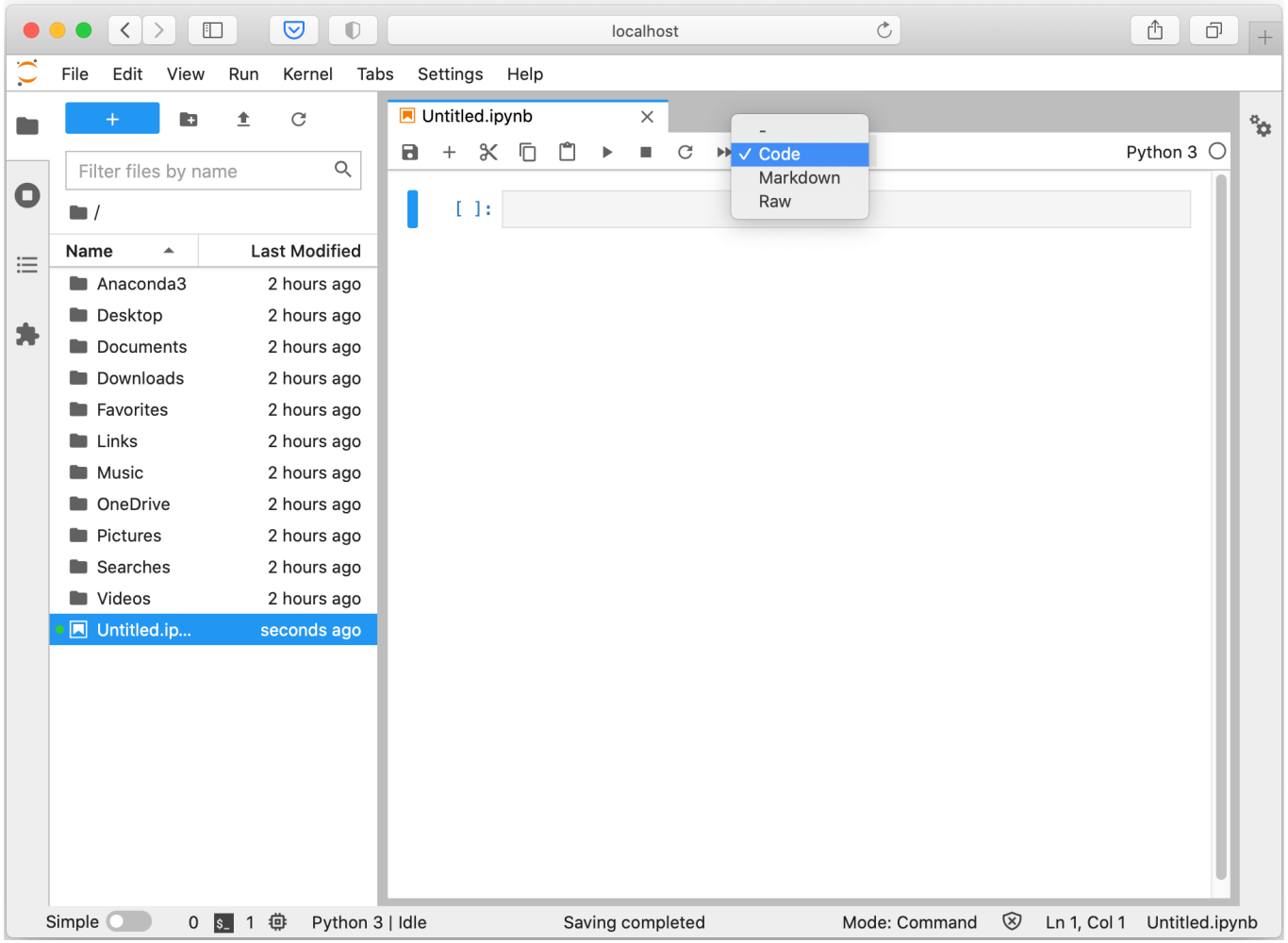
Overview

Jupyter Notebook provides an alternative way of running Python codes. Instead of using the terminal prompt environment, Jupyter Notebook uses the web browser. The biggest advantage of this web-based approach is that it combines rich-formatted text, inline code, mathematical formulas, plots and graphs, and other media into a single document.

JupyterLab on the other hand is the successor of Jupyter Notebook, as such JupyterLab includes the Jupyter Notebook as well as some other very useful tools for Python development.

JupyterLab (Jupyter Notebook) can be invoked by typing `jupyter lab` (`jupyter notebook`) at the terminal prompt.

JupyterLab frontend



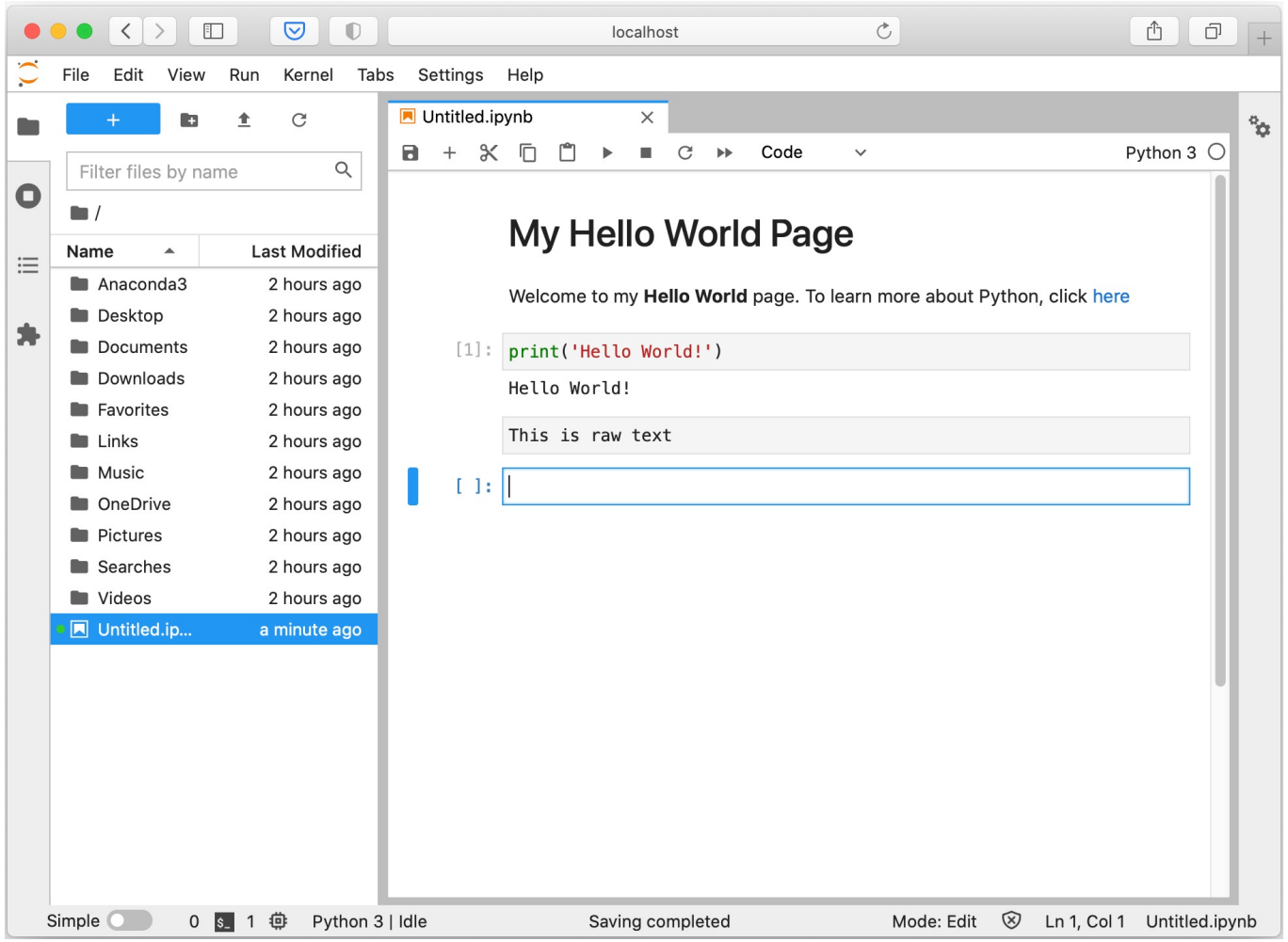
Source of the image: Own creation.

Code: for inputting Python codes

Markdown: for formatting text using simple syntax

Raw: for inputting raw text without formatting

JupyterLab frontend



Markdown

Code input

Code output

Raw

Source of the image: Own creation.

REVIEW STUDY GOALS

- Understand why software developers choose Python
- Describe the strengths and weaknesses of the Python programming language
- Know how to download and install Python
- Explain and use the various parts of a Python development environment

SESSION 1

TRANSFER TASK

TRANSFER TASK

Create a markdown cheat sheet for JupyterLab. Use Raw to show the syntax and Markdown to show the result. Your cheat sheet should show the following elements:

Text Style	Section Style
Title (# Title)	Indentation (>Indentation)
Heading (## Heading)	Bullet points (bullet 1, bullet 2) (- Bullet)
Subheading (### Subheading)	Numbered items (item 1, item 2) (1. Item)
Bold text (**Bold**)	Blue coloured box for tips and notes ( )
Italic text (*Italic*)	Orange (or Green) coloured box for examples ( )
Monospace text (`Monospace text`)	Red coloured box for warnings ( )
	Horizontal line (***)

Source of the text: Basic Syntax, n.d..

TRANSFER TASK
PRESENTATION OF THE RESULTS

Please present your results.

The results will be discussed in plenary.

1. Which of the following is not a reason why Python is such a popular software development language?
   a) Python is relatively easy to read.
   b) Python code is compiled into an executable, which is helpful for web development.
   c) Python is very concise, meaning you can do more with less code.
   d) Python has a large community of active developers building libraries and tools for Python development.

2. What is the primary function of the Python interpreter?

3. What are the benefits of JupyterLab?

LIST OF SOURCES

Basic Syntax (n.d.). *Markdown Guide.* https://www.markdownguide.org/basic-syntax/