Introducción a Python

Andrés Quintero Zea

Programación 2024-1

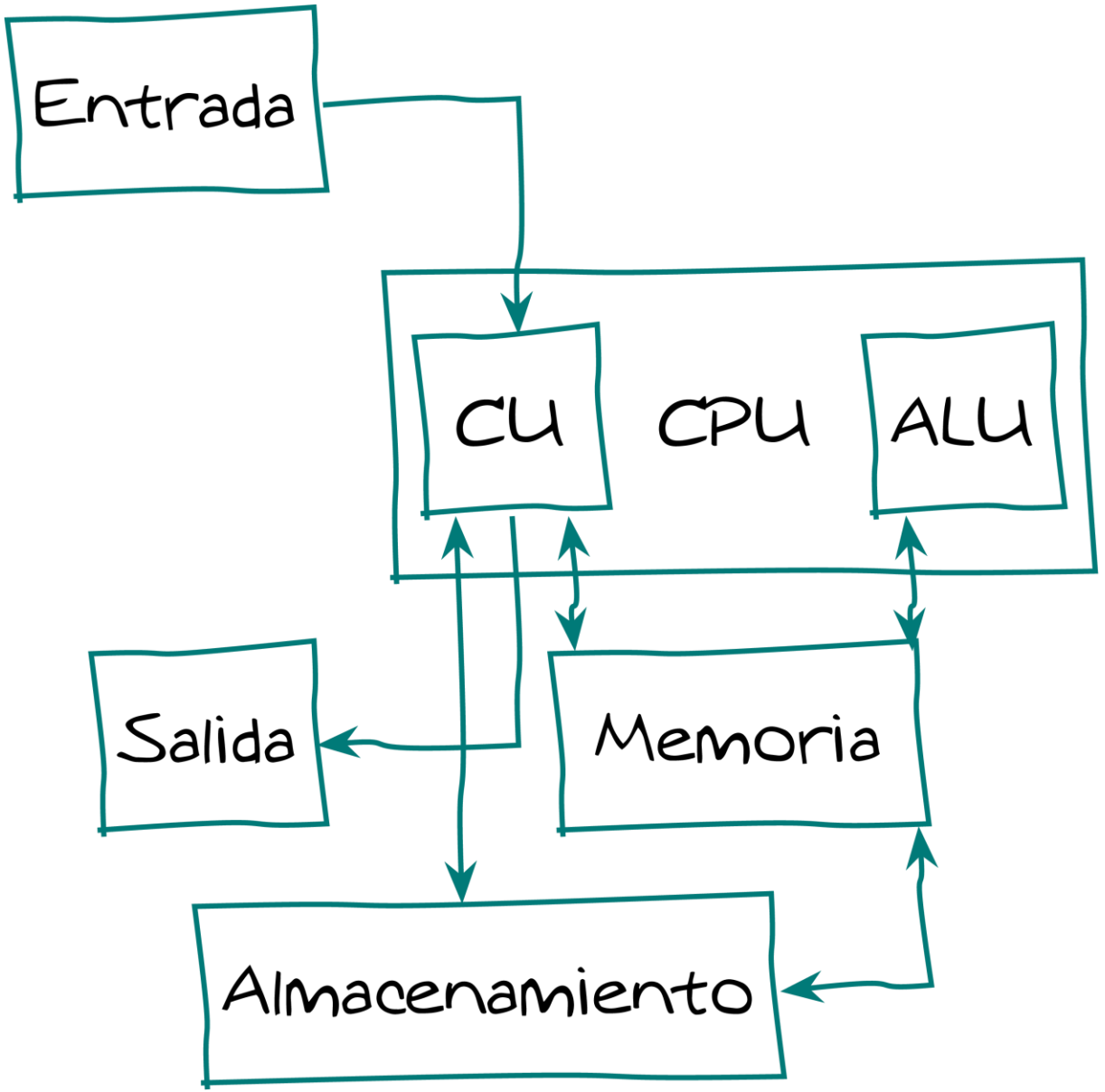
Resumen de la sesión

- Visión general sobre computación y programación
- IPython
- *Scripts* en Python
- Jupyter Notebooks

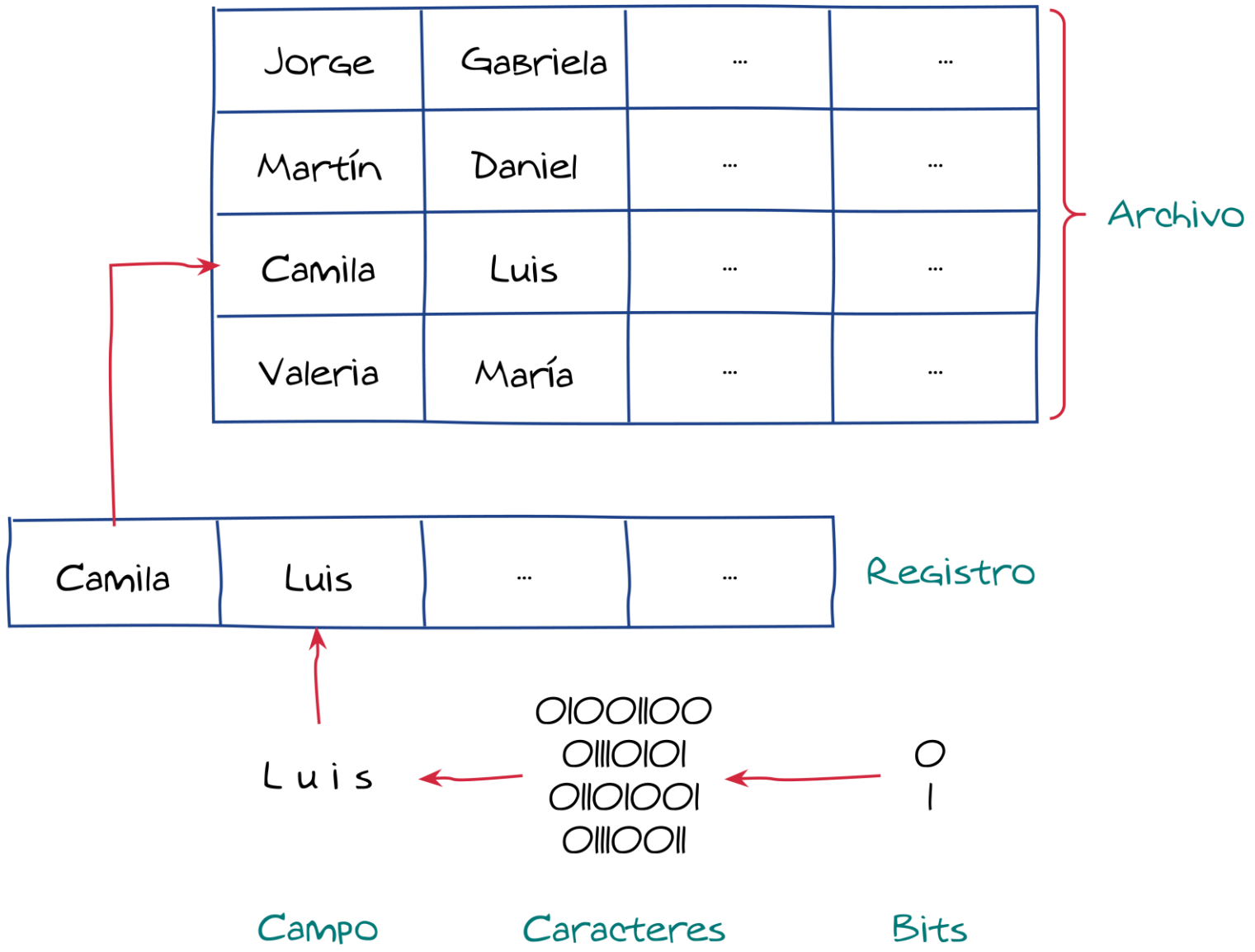
Hardware y Software

Un computador se compone de varios dispositivos físicos denominados *hardware* y procesan datos bajo el control de secuencias de instrucciones llamadas *programas* o *software*

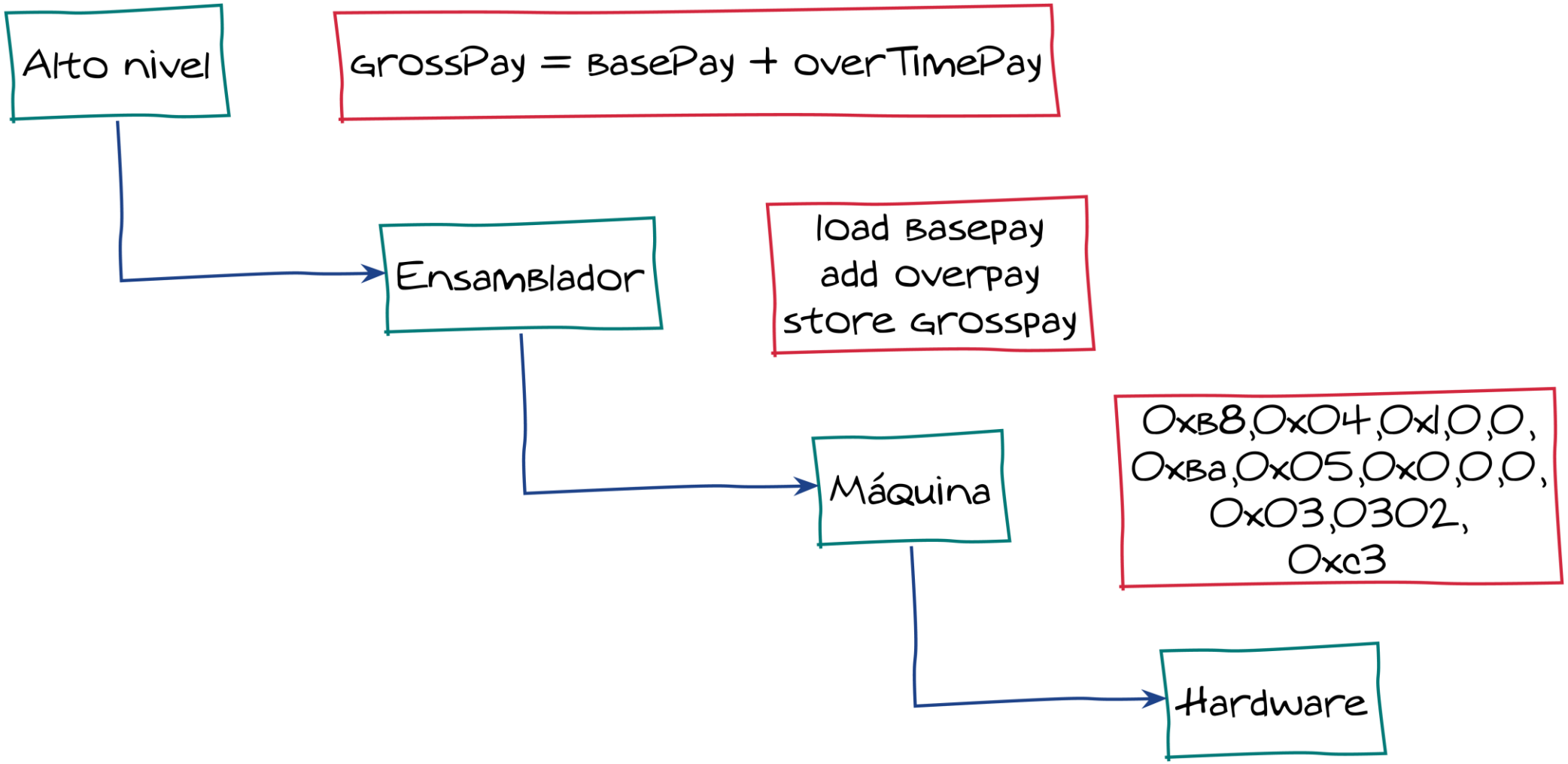
Componentes de un computador



Jerarquía de datos



Jerarquía de los lenguajes



Python

Lenguaje de programación de alto nivel, interpretado y de propósito general
Se trata de un lenguaje de paradigma imperativo, que permite tanto la programación orientada a procesos como la orientada a objetos

Zen of Python

IPython Console

Console 1/A

```
Python 3.11.7 | packaged by Anaconda, Inc. | (main, Dec 15 2023, 18:05:47) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.20.0 -- An enhanced Interactive Python.

In [1]: import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!

In [2]:
```

Librerías estándar de Python

`collections`—Additional data structures beyond lists, tuples, dictionaries and sets.
`csv`—Processing comma-separated value files.
`datetime`, **`time`**—Date and time manipulations.
`decimal`—Fixed-point and floating-point arithmetic, including monetary calculations.
`doctest`—Simple unit testing via validation tests and expected results embedded in docstrings.
`json`—JavaScript Object Notation (JSON) processing for use with web services and NoSQL document databases.
`math`—Common math constants and operations.
`os`—Interacting with the operating system.
`timeit`—Performance analysis.
`queue`—First-in, first-out data structure.
`random`—Pseudorandom numbers.
`re`—Regular expressions for pattern matching.
`sqlite3`—SQLite relational database access.
`statistics`—Mathematical statistics functions like `mean`, `median`, `mode` and `variance`.
`string`—String processing.
`sys`—Command-line argument processing; standard input, standard output and standard error streams.

Librerías adicionales

Scientific Computing and Statistics

NumPy (Numerical Python)—Python does not have a built-in array data structure. It uses lists, which are convenient but relatively slow. NumPy provides the more efficient `ndarray` data structure to represent lists and matrices, and it also provides routines for processing such data structures.

SciPy (Scientific Python)—Built on NumPy, SciPy adds routines for scientific processing, such as integrals, differential equations, additional matrix processing and more. `scipy.org` controls SciPy and NumPy.

StatsModels—Provides support for estimations of statistical models, statistical tests and statistical data exploration.

Data Manipulation and Analysis

Pandas—An extremely popular library for data manipulations. Pandas makes abundant use of NumPy's `ndarray`. Its two key data structures are `Series` (one dimensional) and `DataFrames` (two dimensional).

Visualization

Matplotlib—A highly customizable visualization and plotting library. Supported plots include regular, scatter, bar, contour, pie, quiver, grid, polar axis, 3D and text.

Seaborn—A higher-level visualization library built on Matplotlib. Seaborn adds a nicer look-and-feel, additional visualizations and enables you to create visualizations with less code.

Librerías ML, DL, RL

Machine Learning, Deep Learning and Reinforcement Learning

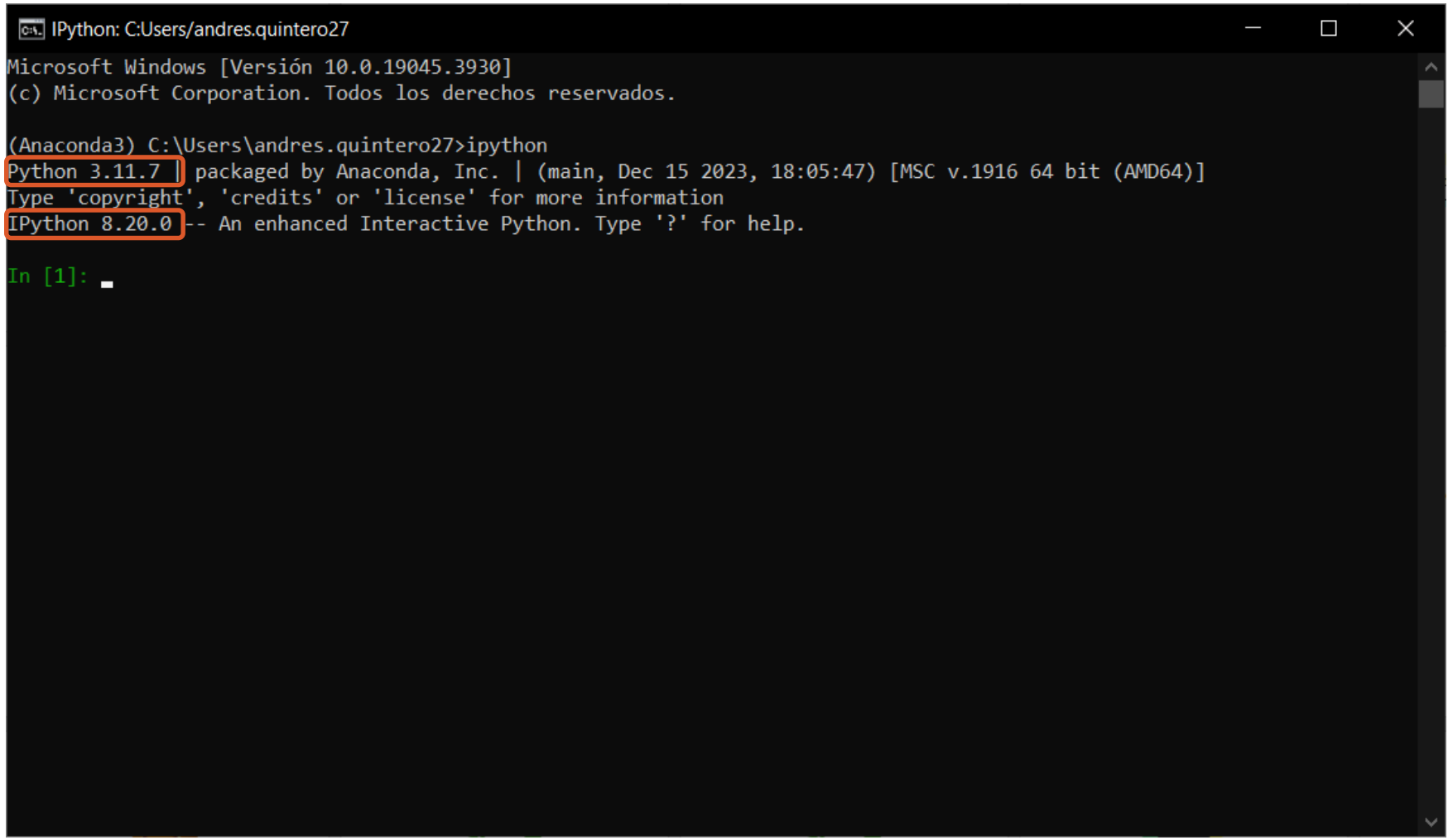
scikit-learn—Top machine-learning library. Machine learning is a subset of AI. Deep learning is a subset of machine learning that focuses on neural networks.

Keras—One of the easiest to use deep-learning libraries. Keras runs on top of TensorFlow (Google), CNTK (Microsoft's cognitive toolkit for deep learning) or Theano (Université de Montréal).

TensorFlow—From Google, this is the most widely used deep learning library. TensorFlow works with GPUs (graphics processing units) or Google's custom TPUs (Tensor processing units) for performance. TensorFlow is important in AI and big data analytics—where processing demands are enormous. You'll use the version of Keras that's built into TensorFlow.

OpenAI Gym—A library and environment for developing, testing and comparing reinforcement-learning algorithms. You'll explore this in the Chapter 16 exercises.

Intérprete IPython



Jupyter Notebook