

Python Essentials 1

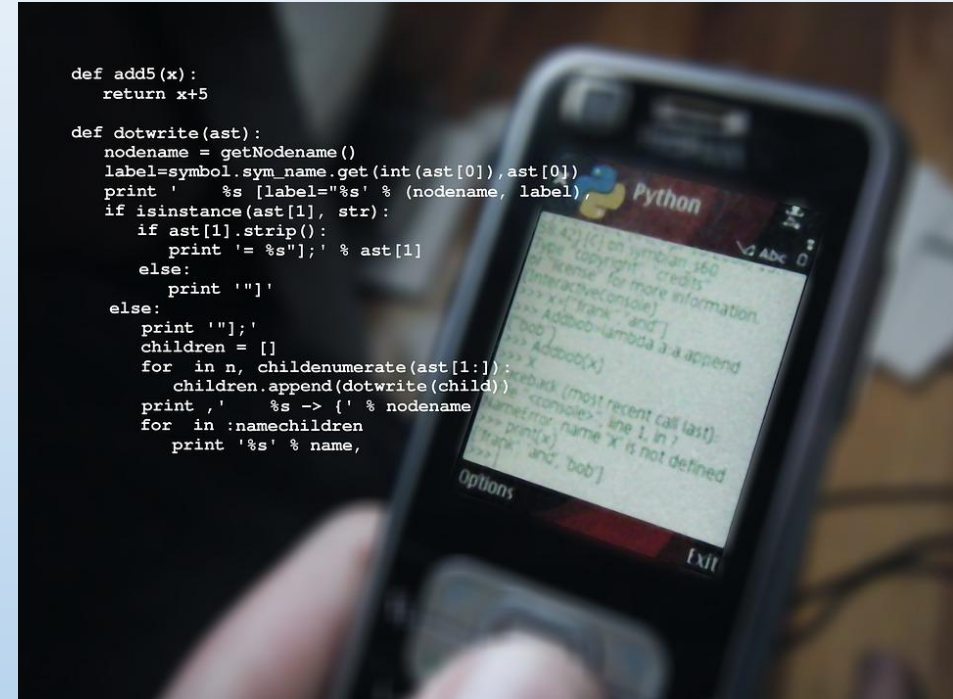


PCEP™ – Certified Entry-Level Python Programmer

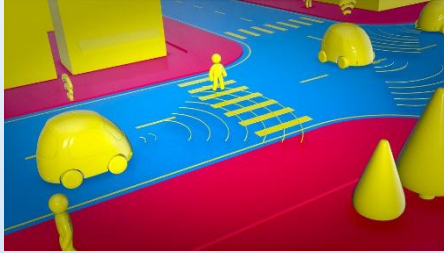
Isabel Maniega

Lenguaje de programación Python

- Python es un lenguaje de programación potente y fácil de aprender.
- Tiene estructuras de datos de alto nivel eficientes y un simple pero efectivo sistema de programación orientado a objetos.
- La elegante sintaxis de Python y su tipado dinámico, junto a su naturaleza interpretada lo convierten en un lenguaje ideal para scripting y desarrollo rápido de aplicaciones en muchas áreas, para la mayoría de plataformas.



Salidas Profesionales Python



Vehículos de
conducción autónoma



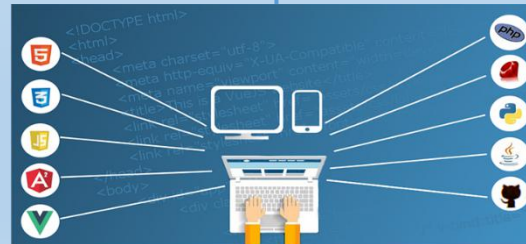
Detección de fraude



Big Data



Sistemas de
recomendación



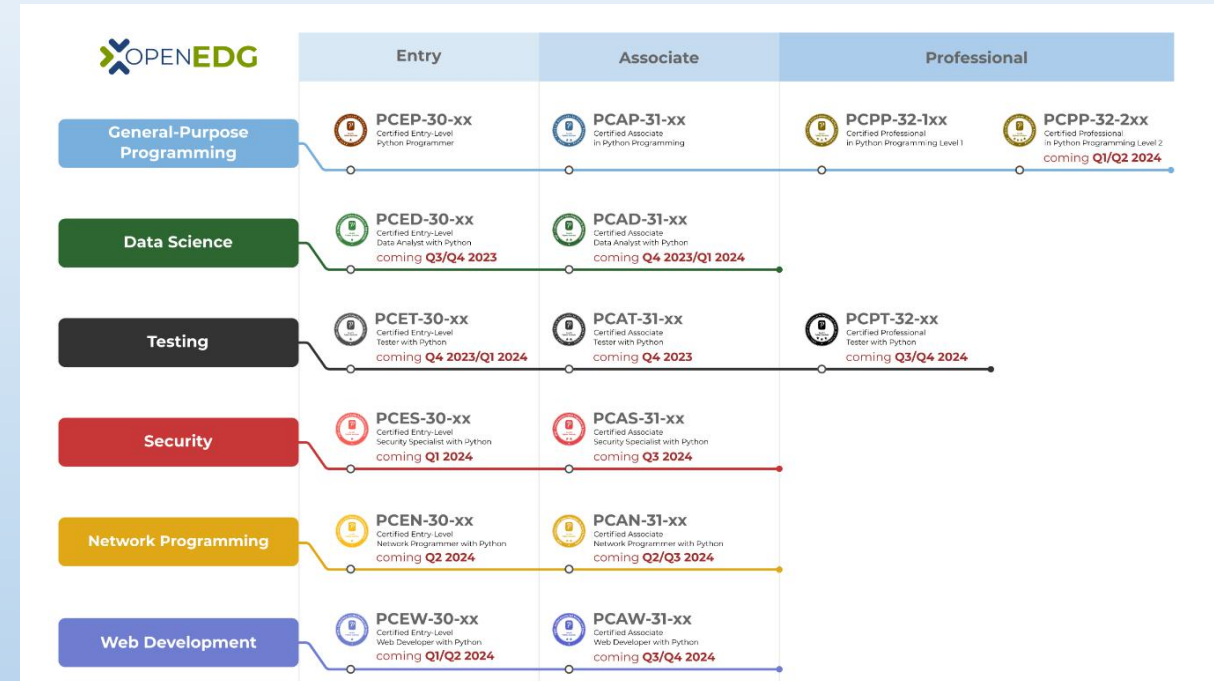
Frameworks webs



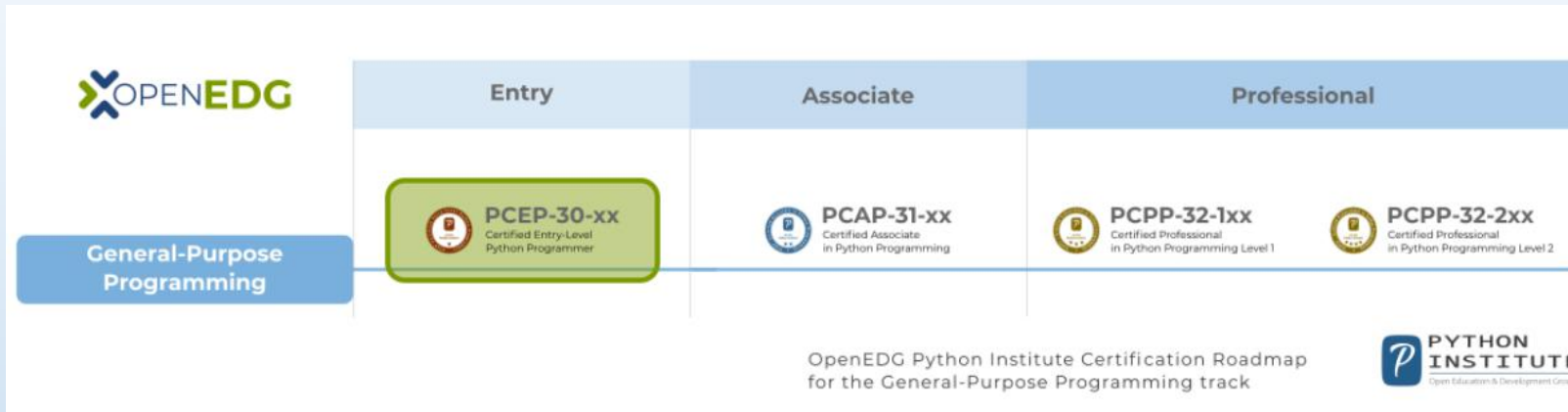
Deep Learning

PCEP. Python Institute y otras certificaciones

- Python Institute es una organización sin ánimo de lucro que se encarga de promover el lenguaje de programación Python.
- Dentro de la oferta de certificaciones que Python Institute pone a nuestro alcance podemos encontrar PCEP, PCAP, PCPP y CEPP. En la siguiente figura podemos ver un esquema general donde cada una de estas certificaciones está ordenada según el nivel de conocimientos y/o aptitudes que acredita la obtención de cada una de estas.



Certificación PCEP



En este curso aprenderás:

- Los conceptos universales de la programación en computadora.
- La sintaxis y la semántica del lenguaje Python.
- Habilidades prácticas para resolver desafíos típicos de implementación.
- A utilizar los elementos más importantes de la biblioteca estándar de Python.
- A instalar tu entorno de ejecución.
- A diseñar, escribir, probar y depurar tus propios programas Python.

El curso se divide en 4 módulos:

Módulo 1: Introducción a Python y a la programación de computadora.

Módulo 2: Tipos de datos, variables, operaciones básicas de entrada y salida, y operadores básicos.

Módulo 3: Valores booleanos, ejecución condicional, bucles, listas y su procesamiento, operaciones lógicas y de bit a bit.

Módulo 4: Funciones, tuplas, diccionarios y procesamiento de datos.

PCEP-30-02 Exam Syllabus

Status: **Live & Active**



The exam consists of four sections:

- Section 1 → 7 items, Max Raw Score: 180 (18%)
- Section 2 → 8 items, Max Raw Score: 290 (29%)
- Section 3 → 7 items, Max Raw Score: 250 (25%)
- Section 4 → 8 items, Max Raw Score: 280 (28%)

Section 1: Computer Programming and Python Fundamentals (18%)

Section 2: Control Flow – Conditional Blocks and Loops (29%)

Section 3: Data Collections – Tuples, Dictionaries, Lists, and Strings (25%)

Section 4: Functions and Exceptions (28%)

Fundamentos de Python

Python es un lenguaje de programación de alto nivel, interpretado, orientado a objetos y de uso generalizado que se utiliza para la programación de propósito general.

Un **lenguaje de alto nivel** es similar al lenguaje natural ya que usa símbolos, palabras y convenciones legibles para los humanos y nos permite expresar comandos a computadoras que son mucho más complejas que la lista de instrucciones (IL) que estas entienden.

Es importante conocer que hay dos formas diferentes de transformar un programa de un lenguaje de programación de alto nivel a un lenguaje de máquina:

- **Compilación** - El programa fuente se traduce una vez obteniendo un archivo que contiene el código máquina (por ejemplo, un .exe si se trabaja en windows) y es este el archivo que se distribuye. Cada vez que se modifica el código fuente debe repetirse esta traducción. El programa que realiza la traducción se llama compilador o traductor.
- **Interpretación** - El programa fuente se traduce a código máquina cada vez que se ejecuta el programa. El programa que realiza este tipo de transformación se denomina intérprete ya que interpreta el código cada vez que se ejecuta. El usuario final necesitará el código fuente más el intérprete para poderlo ejecutar.

Fundamentos de Python

	COMPILACIÓN	INTERPRETACIÓN
VENTAJAS	<ul style="list-style-type: none">• La ejecución del código traducido suele ser más rápida.• Solo el programador debe tener el compilador; el usuario final puede usar el código sin él.• El código traducido se almacena en lenguaje máquina, ya que es muy difícil de entender, es probable que tus propios inventos y trucos de programación sigan siendo un secreto.	<ul style="list-style-type: none">• Puedes ejecutar el código en cuanto lo completes; no hay fases adicionales de traducción.• El código se almacena utilizando el lenguaje de programación, no el de la máquina; esto significa que puede ejecutarse en computadoras que utilizan diferentes lenguajes máquina; no se compila el código por separado para cada arquitectura diferente.
DESVENTAJAS	<ul style="list-style-type: none">• La compilación en sí misma puede llevar mucho tiempo; es posible que no puedas ejecutar tu código inmediatamente después de cualquier modificación.• Tienes que tener tantos compiladores como plataformas de hardware en las que desees que se ejecute tu código.	<ul style="list-style-type: none">• No esperes que la Interpretación incremente tu código a alta velocidad: tu código compartirá la potencia de la computadora con el intérprete, por lo que no puede ser realmente rápido.• Tanto tú como el usuario final deben tener el intérprete para ejecutar el código.

Fundamentos de Python

Python es un lenguaje **interpretado**, es decir, no es necesario compilarlo para ejecutarlo, sino que se ejecuta directamente en el ordenador utilizando el **intérprete**, heredando todas las ventajas de los lenguajes interpretados:

- Permite ejecutar el código en cuanto se termina de escribir ya que no hay fases adicionales de traducción. En cambio, en los lenguajes compilados una vez se termina de escribir o modificar el código es necesario compilarlo lo que puede llevar mucho tiempo.
- El código se almacena utilizando el lenguaje de programación, no el de la máquina; esto significa que puede ejecutarse en máquinas que utilizan diferentes lenguajes máquina. Es decir, es multiplataforma. En cambio, en los lenguajes compilados el código se guarda en lenguaje máquina por lo que si se quiere utilizar en otra máquina con un lenguaje máquina diferente es necesario compilarlo de nuevo para esa máquina.



Fundamentos de Python

Python fue creado por **Guido van Rossum**, nacido en 1956 en Haarlem, Países Bajos. Por supuesto, Guido van Rossum no desarrolló y evolucionó todos los componentes de Python.

Aunque puede que conozcas a la pitón como una gran serpiente, el nombre del lenguaje de programación Python proviene de una vieja serie de comedia de la BBC llamada **Monty Python's Flying Circus**.

En el apogeo de su éxito, el equipo de Monty Python estaba realizando sus escenas en vivo para audiencias en todo el mundo, incluso en el Hollywood Bowl.

Dado que Monty Python es considerado uno de los dos nutrientes fundamentales para un programador (el otro es la pizza), el creador de Python nombró el lenguaje en honor al programa de televisión.

Guido
van
Rossum











Instalaciones necesarias

Python version 3.10.11: <https://www.python.org/downloads/>

Looking for a specific release?

Python releases by version number:

Release version	Release date	Click for more	
Python 3.11.4	June 6, 2023	 Download	Release Notes
Python 3.7.17	June 6, 2023	 Download	Release Notes
Python 3.8.17	June 6, 2023	 Download	Release Notes
Python 3.9.17	June 6, 2023	 Download	Release Notes
Python 3.10.11	April 5, 2023	 Download	Release Notes
Python 3.11.3	April 5, 2023	 Download	Release Notes
Python 3.10.10	Feb. 8, 2023	 Download	Release Notes
Python 3.11.2	Feb. 8, 2023	 Download	Release Notes

[View older releases](#)

- Windows se encuentra directamente en la tienda para su instalación (mejor opción ejecutable):

<https://www.python.org/downloads/release/python-31011/>

- Ubuntu viene instalado por defecto con python 3.10.12

<https://docs.python.org/es/3.10/tutorial/>

<https://docs.python.org/es/3.10/using/windows.html>

Python

- Comprobación de instalación: `python --version`

```
isabel@isabel-SVE1512E1EW:~$ python3 --version
Python 3.8.10
```

- Intérprete de Python:

```
isabel@isabel-SVE1512E1EW:~$ python3
Python 3.8.10 (default, Mar 15 2022, 12:22:08)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

- ¿Cómo realizar una instalación de librerías?

Existe una página con todos los repositorios de las librerías esa página es PYPI:

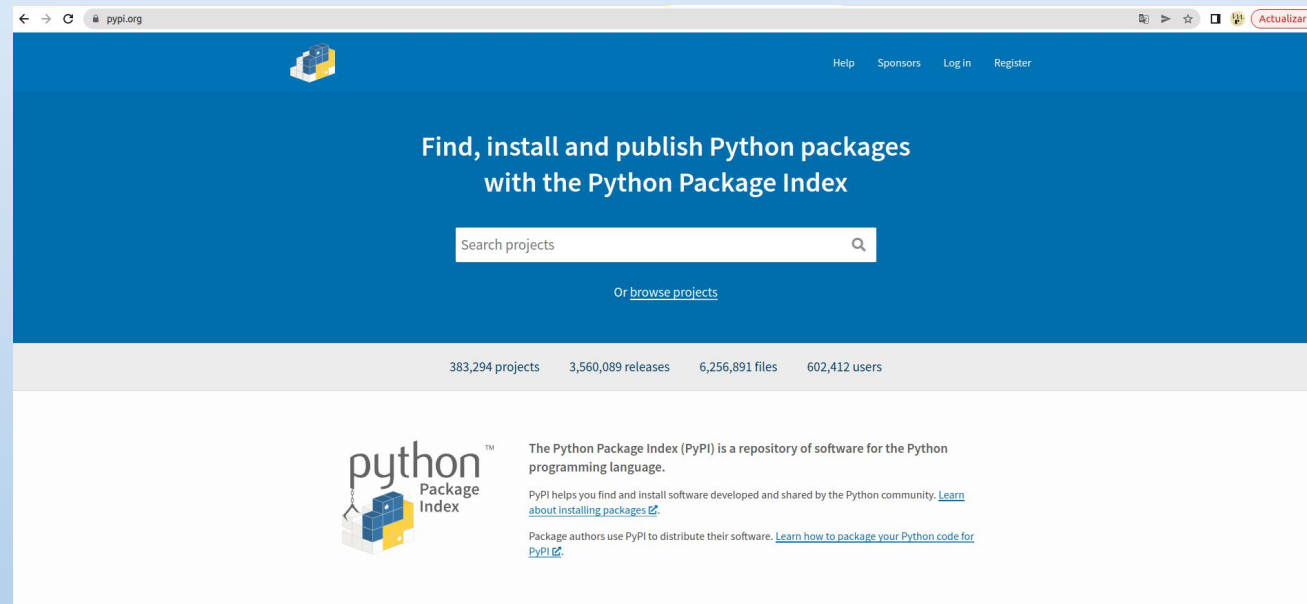
<https://pypi.org/>

Python

- ¿Cómo realizar una instalación de librerías?

Existe una página con todos los repositorios de las librerías esa página es PYPI:

<https://pypi.org/>



En esta página podemos encontrar toda la información relativa a documentación, ejemplos, como instalar, versiones disponibles, etc.

Python

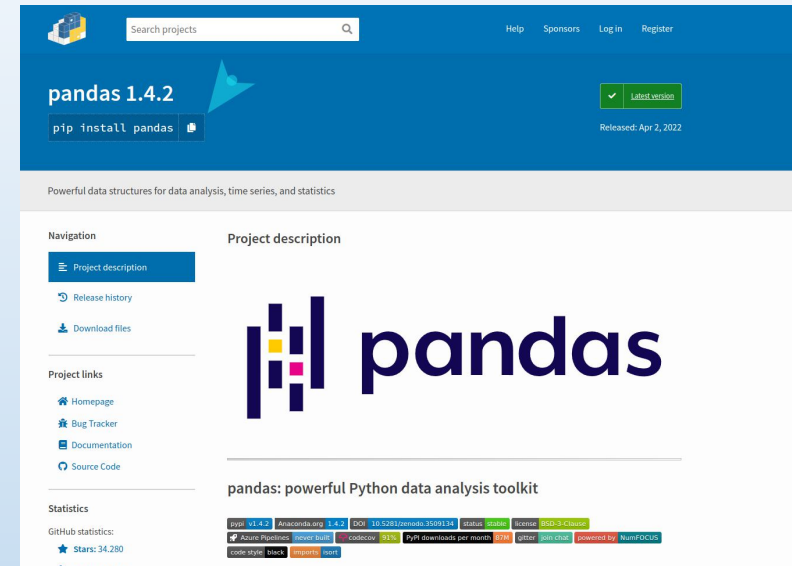
¿Cómo realizar una instalación de librerías?

- Para la realización de la instalación iremos a esa página y buscaremos la librería a instalar, copiaremos la instrucción para su instalación tal y como se indica en la imagen:

- Iremos a la cmd y pegaremos la instrucción y pulsaremos intro:

En algunos sistemas operativos se pide la instalación de pip, ya que aún no está instalada

- Veremos que nos a realizado con éxito la instalación.



```
(env) isabel@isabel-SVE1512E1EW:~/atom$ pip install pandas
Collecting pandas
  Using cached pandas-1.4.2-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (11.7 MB)
Collecting pytz>=2020.1
  Using cached pytz-2022.1-py2.py3-none-any.whl (503 kB)
Collecting numpy>=1.18.5; platform_machine != "aarch64" and platform_machine != "arm64" and python_version < "3.10"
  Downloading numpy-1.22.4-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.9 MB)
    | 16.9 MB 10.5 MB/s
Collecting python-dateutil>=2.8.1
  Using cached python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
Collecting six>=1.5
  Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: pytz, numpy, six, python-dateutil, pandas
Successfully installed numpy-1.22.4 pandas-1.4.2 python-dateutil-2.8.2 pytz-2022.1 six-1.16.0
(env) isabel@isabel-SVE1512E1EW:~/atom$
```


Instalaciones necesarias

- Jupyter notebook: <https://jupyter.org/install>



Jupyter Notebook

Install the classic Jupyter Notebook with:

```
pip install notebook
```

To run the notebook:

```
jupyter notebook
```

