PYTHON BASICO

Como vamos a trabajar

Vamos a utilizar un entorno de trabajo colaborativo para que tengáis acceso a todo, mis códigos y que también tengáis vuestros códigos.

Lo que haremos será trabajar con **Github**.

1. Instalar Python

Necesitamos un IDE para trabajar. El IDE es **Visual Studio Code** que sirve para programar con Python

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

Debemos separa los conceptos:

1. Jupyter es un cuaderno de trabajo Python. Sirve para análisis de datos
2. Python básico se mueve en cualquier script.

Todo esto está reunido dentro de cualquier entorno.

Para trabajar con Jupyter, necesitamos un entorno.

Para trabajar con Script, necesitamos un entorno.

Para trabajar con Fast Api o Django, necesitamos un entorno.

La diferencia es que mediante Visual Studio Code, podemos utilizar TODO.

Mediante extensiones, podemos hacer que nuestro entorno funcione sin problemas dentro de VS Code.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Word

El contenido generado por IA puede ser incorrecto.

Una vez que tenemos esto, ya se ha ejecutado un elemento para poder crear aplicaciones de Python

Todas las extensiones de los ficheros Python (mentira) terminan en **.py**

Simplemente, cualquier fichero que tengamos se puede ejecutar mediante VS Code sin necesidad de nada más.

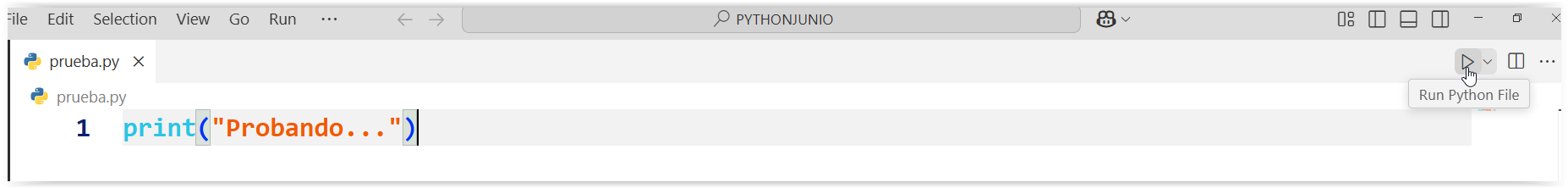
Para trabajar, simplemente tendremos una carpeta con nuestros ficheros Python.

Dependiendo del proyecto/framework se generan más carpetas. Pero en este tipo básico, simplemente necesitamos una carpeta en nuestro ordenador y dentro iremos creando todas las clases y características

PYTHONJUNIO

Interfaz de usuario gráfica, Aplicación, Tabla

El contenido generado por IA puede ser incorrecto.



Python es un lenguaje de programación. Contiene programación orientada a objetos y se comunica con múltiples elementos, por ejemplo, Web, IoT, Api, Análisis de datos, Big Data…

**Importante:**

* **Diferencia mayúsculas de minúsculas**
* **Las tabulaciones son básicas, delimitan el código del programa**
* **Las variables (mentira) no tienen tipado, simplemente le asignamos un valor.**

Las variables se declaran en el código

numero = 14

texto = “Soy un texto”

**print** en nuestro entorno será el que nos servirá para visualizar que sucede. Dependiendo del Framework que utilicemos, no se utiliza print

Dependiendo del tipo de variable, podremos sumar o restar o dividir o podremos buscar una letra o convertir a mayúsculas

Para trabajar y ser ordenados, iremos nombrando a nuestros ficheros de una forma determinada. En principio no importa el nombre del fichero, pero normalmente, en Python se declaran los programas en minúsculas.

Creamos un nuevo programa llamado **python01variables.py**

**print("Ejemplo de variables")**

**numero = 19**

**texto = "Hola mundo"**

**#Comentarios en Python**

**# Para comentar y descomentar código de una sola vez en VS Code.ç**

**# Comentar:  CONTROL + K + C**

**# Descomentar: CONTROL + K + U**

**print(numero)**

**print(texto)**

**# Tenemos la posibilidad de concatenar +: Verifica el tipado**

**# print("numero " + numero) # error**

**# Con coma no verifica tipado**

**print("numero ", numero)**

**print("texto " + texto)**

**#print f nos permite concatenar múltiples variables en un String sin**

**#importar el tipado: Cada variable irá entre llaves {variable}**

**print(*f*"El texto es {texto} y el numero es {numero}")**

**#Dentro de las variables tenemos funciones de conversión lo que permiten convertir**

**#entre un tipo y otro, por ejemplo:**

**#str(variable): Convierte un valor a string**

**#float(variable): Convierte variable a tipo decimal**

**#int(variable): Convierte variable a entero**

**print("numero " + *str*(numero))**

CONDICIONALES

Un condicional dentro de cualquier lenguaje implica que un programa no será lineal.

Para poder utilizar condicionales/preguntas en nuestro programa se utilizan las TABULACIONES para delimitar si la pregunta es correcta o no es correcta.

Al hablar de condiciones, estamos hablando de IF

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Para poder comparar, necesitamos la sintaxis de los operadores de comparación de Python.

> Mayor

< Menor

>= Mayor o igual

<= Menor o igual

== Igual

!= Distinto

Vamos a realizar un simple programa en el que indicaremos si un número es positivo o negativo o cero.

Pediremos el número al usuario mediante una función de Python llamada **input()**

Creamos un nuevo programa llamado **python02condicionales.py**

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

ELSE

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

ELSE IF ANIDADO

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Si tiene que ver con la misma pregunta, lo que podemos utilizar es la sentencia **elif**

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.