

## 1.3. Algunos IDE

---

### 1.3.1. IDLE

- 1.3.1.1. Configuración Básica
- 1.3.1.2. Entorno Interactivo (Shell)
- 1.3.1.3. Editor de Código
- 1.3.1.4. Ejecutar el Código
- 1.3.1.5. Instalación de Módulos

### 1.3.2. Jupyter

---

El escoger un entorno de trabajo depende de las necesidades, recursos y gustos, por ejemplo, yo empecé con IDLE y con él aprendí las bases de Python, luego me pasé a Spyder cuando tuve proyectos más grandes, sin embargo, para esta guía usé JupyterLab y VS Code, y hay muchos otros que no mencionaré ya que nunca he trabajado con ellos.

A continuación, se introducirán el IDE que viene con la instalación de Python, sin embargo, cabe resaltar que esta es sólo una pincelada de todas sus funciones.

### 1.3.1 IDLE

---

Una vez instalado IDLE, este se ubicará donde ustedes le indiquen en el proceso de instalación.

Al abrirlo podremos ver la ventana principal de IDLE, como se muestra en la *Figura 1*. Como pueden ver es una interfaz muy sencilla y no tiene todos los botones, opciones, ventanas divididas y otras características de los IDE más sofisticados.

Fig1\_Shell\_IDLE

*Figura 1: Ventana principal del IDLE*

Los tres “signos de mayor” (`>>>`) que se ven en la última línea son el prompt. IDLE nos presenta un entorno interactivo, lo que quiere decir que no es necesario que tengamos escrito un archivo aparte con nuestro código para empezar a experimentar con Python.

#### 1.3.1.1. Configuración Básica

Si queremos configurar aspectos de interacción con IDLE, lo podemos hacer en el menú **Options/Configure IDLE**. En la *Figura 2* se muestra la ventana de configuración.

Tres recomendaciones:

1. Utilicen un tipo de letra (Font) que sea no proporcional (monospaced), donde cada letra ocupa el mismo espacio. Tipos como Courier, Courier New, Terminal y muchos terminados en “Mono” (Liberation Mono,

Ubuntu Mono) son adecuados. Depende de las preferencias personales.

2. Recomiendo usar un tema oscuro, y aumentar el tamaño de la letra a 11 o 12 puntos, debido a que no desgasta tanto la vista al momento de trabajar por un tiempo prolongado.
3. Mantengan la [indentación](#) estándar en cuatro espacios. Más adelante será claro por qué esto es buena idea en Python.

**Nota:** La primeras dos recomendaciones aplican para la configuración de cualquier IDE, y la última para cualquiera con el que se vaya usar Python.

Fig2\_Configuracion\_Shell\_IDLE

*Figura 2: Configuración del IDLE*

### 1.3.1.2. Entorno Interactivo (Shell)

Como les decía, IDLE es interactivo. Eso quiere decir que podemos usar comandos de Python directamente en la ventana. Por ejemplo, en la Figura 3 podemos ver como asigno el valor 3 a la variable `a`, 4 a la variable `b`, y luego asigno a la variable `c` la multiplicación de `a*b` y si invoco el nombre de una variable `c` y oprimo veré en pantalla su valor actual.

Fig3\_Interaccion\_Shell\_IDLE

*Figura 3: Ejemplo interacción en el Shell*

### 1.3.1.3. Editor de Código

En nuestro caso el Shell nos va a servir para hacer ensayos, hacer consultas con el comando `help`, y ejecutar programas, sin embargo, cuando vayamos a escribir programas debemos abrir un documento `.py` en blanco (*Figura 4*), para esto debemos ir a **File/New File (Ctrl+N)**.

Fig4\_Hola\_Mundo\_IDLE

*Figura 4: Hola mundo en IDLE*

Descargar: [Programa 0 - Hola-mundo-IDLE.py](#)

### 1.3.1.4. Ejecutar el Código

Para ejecutar el programa, primero hay que guardarlo y luego dirigirse a **Run/Run Module** o solo oprimir **F5**, y el programa será ejecutado en el Shell, como se ve en la *Figura 5*.

Fig5\_Ejecucion\_Hola\_Mundo\_IDLE

*Figura 5: Ejecución Hola mundo en IDLE*

### 1.3.1.5. Instalación de Módulos

Para poder utilizar módulos de terceros estos deben ser instalados [aquí](#) puede encontrar las instrucciones para hacerlo en el sistema operativo Windows.

---

Anterioror - Siguiente		

Home