

Mi lenguaje elegido es Python. Por mi apellido Parra.

Python es un lenguaje interpretado de alto nivel de muy alta popularidad y uso común. Su popularidad se basa en parte a su facilidad de escritura, así como la comodidad que tiene no necesitar compilarlo para poder probar lo que se ha realizado. Se origino a finales del siglo 20, entre los 80 y 90 para ser exactos. Actualmente se usa en todas las áreas de la programación, ya sea Data Science, Machine Learning, Web Develoment, ya que al ser Open-Source la gente poco a poco ha ido creando y puliendo las librerías y módulos que posee el lenguaje.

Entre las estructuras de control de flujo tenemos los más básicos que son:

- **If/elif/else:** Los condicionales de uso común para hacer comparaciones. El if y el else se mantienen de lo usual. Pero a diferencia de algunos lenguajes que suele colocar else if, en este es elif.
- **While/for:** Tenemos los ciclos, while que itera indefinidamente dada una condición y el for es algo distinto a lo usual. En vez del típico **for int i; condición, i++** el for funciona con rangos, o para iterar listas/arreglos. Esto se hace colocando **for <elemento> in <lista>** y en cada iteración <elemento> es un elemento de la lista. Existe la función range para darle un rango. Permitiéndole así parecerse más a los for convencionales.
- **Iteradores:** Python posee iteradores, y como bien se ve en este parcial (Rompiendo la 4ta pared) son muy útiles para ciertos casos y permiten en algunos casos hasta hacer el código más compacto de lo usual para el mismo trabajo.

Python evalua las expresiones y funciones de derecha a izquierda. Siguiendo las precedencias tanto de los booleanos como de la aritmética. Posee a su vez *short circuiting* que lo ayuda a terminar de evaluar una expresión booleana si ve pertinente que ya tiene el resultado. Por ejemplo, si se da el caso de que desear evaluar **true or 2<9** al evaluar el true y ver que le sigue un or. Directamente retornará true ya que por propiedades de la lógica será true no importa que. Los argumentos de las funciones se leen de izquierda a derecha, aunque también se pueden colocar en el orden que quieras mientras le coloques el nombre del parámetro, ejemplo si tengo una función potencia(base, exponente) yo podría llamar a potencia como potencia(exponente: 4, base: 5) y funcionaria.

Python posee tipos pero no tiene una asignación rigurosa. Puedo colocar que `a = 5` y después colocar `a = "Hola"` y no pasará nada. Esto tiene sus ventajas y desventajas. Entre los tipos que posee Python, están: `char`, `bool`, `int`, `float`, `complex`, `str`. Como tipos básicos del lenguaje. A su vez ofrece creación de clases y registros para dar más versatilidad.