

Python Cheatsheet – Crash Course on Python

Curso: Crash Course on Python

Módulo 2: Basic Python Syntax

Tema: Sintaxis básica y estructura de Python

Introducción

Python tiene una sintaxis **limpia y legible** que facilita la escritura de código.

Conceptos clave:

- Indentación define bloques de código (no `{}`)
 - Comentarios con `#`
 - Cada línea generalmente corresponde a una instrucción
 - Se pueden agrupar varias instrucciones en una línea con `;` (no recomendado)
-

Variables y Asignación

```
x = 10          # int
y = 3.14        # float
nombre = "Ana"  # str
activo = True   # bool
```

- Tipado dinámico: Python infiere el tipo automáticamente
 - Buenas prácticas: nombres descriptivos, letras minúsculas, separadas por `_`
-

Comentarios

```
# Esto es un comentario de una línea
"""
Comentario
de varias
líneas
"""
```

- Documentar código es fundamental para mantenibilidad
-

Operadores

Aritméticos

Operador	Función
+	Suma
-	Resta
*	Multiplicación
/	División
//	División entera
%	Módulo
**	Potencia

Comparación

Operador	Función
==	Igual
!=	Distinto
>	Mayor que
<	Menor que
>=	Mayor o igual
<=	Menor o igual

Lógicos

Operador	Función
and	AND lógico
or	OR lógico
not	Negación

◆ Tipos de Datos Comunes

- int, float, bool, str, list, tuple, set, dict
- Conversiones: `int()`, `float()`, `str()`, `bool()`
- Función `type(variable)` → muestra tipo de dato

◆ Input y Output

```
nombre = input("Ingresa tu nombre: ")
edad = int(input("Ingresa tu edad: "))
print(f"Hola {nombre}, tienes {edad} años")
```

- `input()` siempre devuelve **str** → convertir según necesidad
 - F-strings permiten interpolar variables
-

◆ Buenas Prácticas de Sintaxis

- Indentar con **4 espacios**
 - Nombres de variables claros
 - Evitar líneas muy largas (>79 caracteres)
 - Documentar funciones y bloques de código
-

📖 Referencias

- 📖 [Python Docs – Lexical analysis](#)
- 📖 [W3Schools – Python Syntax](#)