

# **Introducción al Python**

Osmar de Jesús Pérez Lozada

2025-07-10

# Tabla de contenidos

<b>Preface</b>	<b>3</b>
<b>Introducción</b>	<b>4</b>
<b>1 Fundamentos</b>	<b>5</b>
1.1 Imagen estática . . . . .	5
1.2 Figura generada por Python . . . . .	5
1.3 Tabla Markdown . . . . .	6
1.4 Tabla desde código test . . . . .	7
<b>2 Summary</b>	<b>8</b>
<b>References</b>	<b>9</b>
<b>Anexos</b>	<b>10</b>
Operadores aritméticos . . . . .	10
Operadores de asignación . . . . .	10
Operadores relacionales . . . . .	11
Operadores lógicos . . . . .	11
Operadores de bits . . . . .	11

# Preface

This is a Quarto book.

To learn more about Quarto books visit <https://quarto.org/docs/books>.

# Introducción

Este libro tiene como propósito servir de apoyo al aprendizaje de quienes se inician en el estudio del Lenguaje Python, ofreciéndole una secuencia que facilite el camino de la formación. También se ha previsto que sirva como guía para aquellos que conociendo el lenguaje requieran realizar consultas.

# 1 Fundamentos

Este capítulo muestra cómo referenciar Figuras y Tablas (ver Figura 1.2 y Tabla 1.2).

## 1.1 Imagen estática

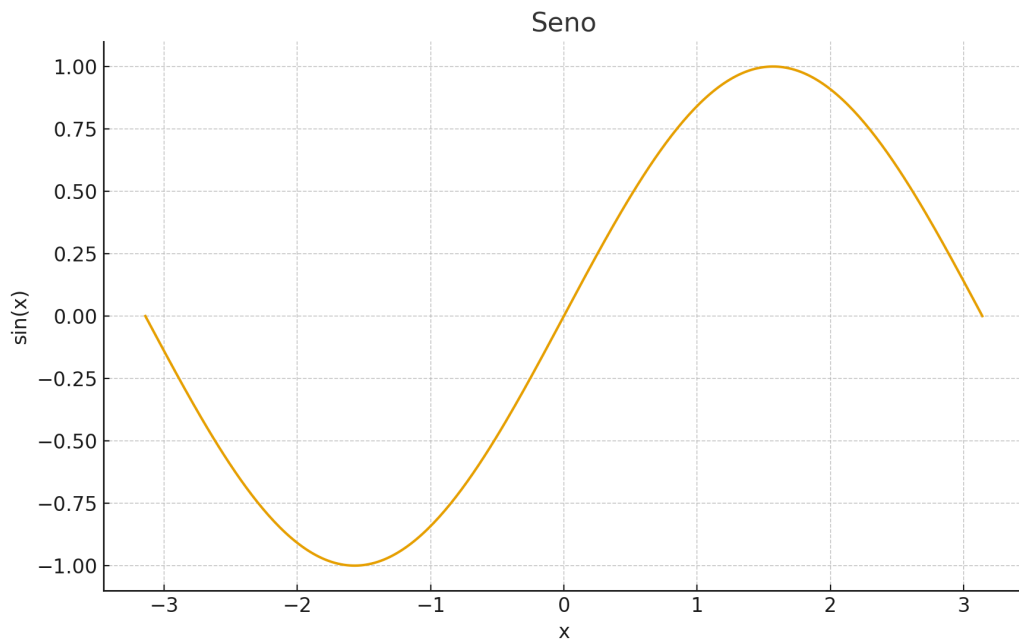


Figura 1.1: Curva seno en  $([-\pi, \pi])$ .

## 1.2 Figura generada por Python

```
import numpy as np
import matplotlib.pyplot as plt
```

```
x = np.linspace(-np.pi, np.pi, 300)
plt.figure()
plt.plot(x, np.sin(x))
plt.xlabel("x"); plt.ylabel("sin(x)"); plt.title("Seno")
plt.show()
```

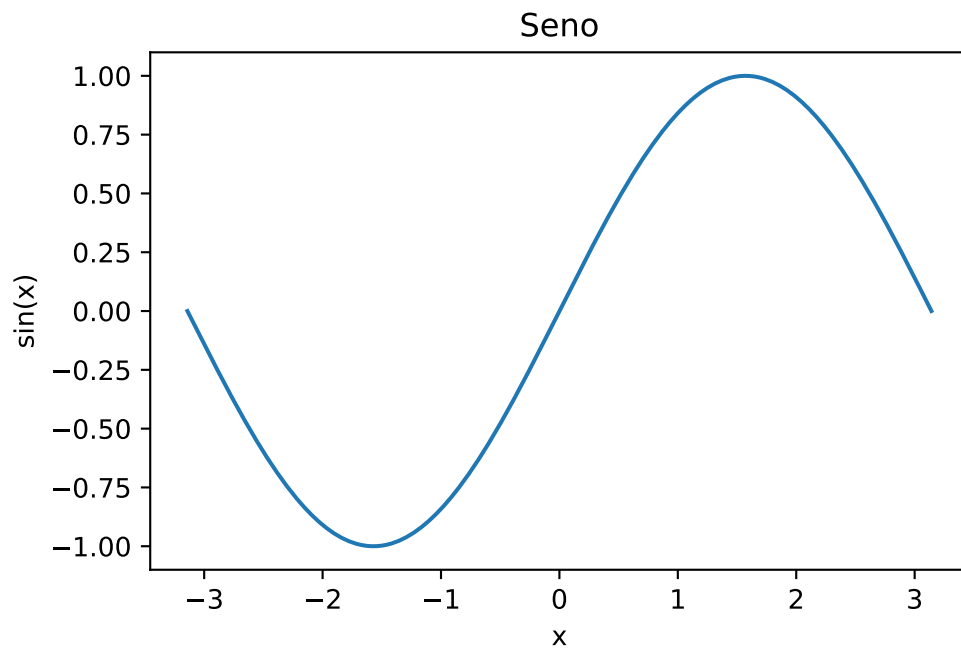


Figura 1.2: Relación entre  $x$  y  $\sin(x)$  en  $[-\pi, \pi]$ .

### 1.3 Tabla Markdown

Operación	Ejemplo	Resultado
Suma	21 + 3	24
Potencia	2 ** 3	08

Tabla 1.1: Operaciones básicas en Python

## 1.4 Tabla desde código test

```
import pandas as pd, numpy as np
pd.DataFrame({"x": range(1,6), "y": np.random.randn(5).round(2)})
```

	x	y
0	1	-0.54
1	2	1.23
2	3	1.34
3	4	-1.02
4	5	0.37

Tabla 1.2: Datos aleatorios de ejemplo.

## 2 Summary

In summary, este libro no tiene contenido.



## References

# Anexos

## Operadores aritméticos

Operador	Descripción	Ejemplo	Resultado
+	Suma	<code>c3 + 2</code>	5
-	Resta	<code>5 - 2</code>	3
*	Multiplicación	<code>4 \* 3</code>	12
/	División real	<code>7 / 2</code>	3.5
//	División entera	<code>7 // 2</code>	3
%	Módulo	<code>7 % 2</code>	1
**	Potencia	<code>2 ** 3</code>	8
@	Producto matricial	<code>A \ B</code>	Matriz resultante

## Operadores de asignación

Operador	Descripción	Ejemplo	Equivalente a
=	Asignación	<code>x = 5</code>	—
+=	Suma y asigna	<code>x += 3</code>	<code>x = x + 3</code>
-=	Resta y asigna	<code>x -= 2</code>	<code>x = x - 2</code>
*=	Multiplica y asigna	<code>x *= 4</code>	<code>x = x * 4</code>
/=	Divide y asigna	<code>x /= 2</code>	<code>x = x / 2</code>
//=	División entera y asigna	<code>x //= 2</code>	<code>x = x // 2</code>
%=	Módulo y asigna	<code>x %= 2</code>	<code>x = x % 2</code>
**=	Potencia y asigna	<code>x **= 2</code>	<code>x = x ** 2</code>
=, &=, ^=, <<=, >>=	Asignaciones con bits	<code>x &amp;= 3</code>	<code>x = x &amp; 3</code>
:=	Asignación por expresión ( <i>walrus</i> )	<code>if (n := len(lista)) &gt; 3:</code>	Asigna dentro de la expresión

## Operadores relacionales

Operador	Descripción	Ejemplo	Resultado
<	Menor que	3 < 5	True
<=	Menor o igual que	3 <= 3	True
>	Mayor que	7 > 4	True
>=	Mayor o igual que	7 >= 7	True
==	Igual que	5 == 5	True
!=	Distinto de	5 != 3	True

## Operadores lógicos

Operador	Descripción	Ejemplo	Resultado
and	Verdadero si ambos son True	(5 > 3) and (2 < 4)	True
or	Verdadero si alguno es True	(5 > 3) or (2 > 4)	True
not	Invierte el valor lógico	not (5 > 3)	False

## Operadores de bits

Operador	Descripción	Ejemplo (a=6 b=3)	Resultado
&	AND	a & b	2
	Or	a   b	7
^	XOR	a ^ b	5
~	NOT	~a	-7
<<	Desplazamiento izq.	a << 1	12
>>	Desplazamiento der.	a >> 1	3