

# Métodos no tan vistos en Python: Strings

[Data Science](#) • 8 de jun. de 2023



Los métodos en Python son un concepto clave que debes conocer si quieres introducirte en el mundo de los datos. Aprende más en este artículo de nuestra guía.

En [Python](#), los métodos son funciones que se definen dentro de una clase y se asocian con objetos de esa clase en particular. Estas funciones pueden ser llamadas utilizando la sintaxis objeto.metodo(). Los métodos son una forma de encapsular la lógica relacionada con un objeto específico y permiten realizar operaciones y manipulaciones sobre los datos de ese objeto??

## Start the journey??

1. [Da tus primeros pasos en la ciencia de datos](#)
2. [Por qué aprender Python puede ayudarte a dar un salto en tu carrera](#)
3. [Programación Orientada a Objetos en Python: Descubre el poder de las historias interactivas](#)
4. [Explorando los diferentes tipos de estructura de datos en Python](#)
5. [Métodos no tan vistos en Python: Strings](#)
6. [Dudas frecuentes y un decálogo para rendir el Henry Challenge](#)

El `}` es un tipo especial de método que se utiliza para operar y manipular cadenas de caracteres en **Python**. Las cadenas son secuencias de caracteres y, al ser un tipo de dato fundamental en este lenguaje, se han definido diversos métodos para trabajar con ellas.

?Artículo recomendado: [Explorando los diferentes tipos de estructura de datos en Python](#)

Como ya sabes un **string (cadena de caracteres)** es una secuencia de caracteres, como letras, números o símbolos que se utiliza para representar texto. Por ejemplo, "¡Hola, mundo!" es un *string* que representa una frase. Y un **método es una función asociada a un objeto o tipo de dato en particular**. En el caso de los strings, los métodos son funciones especiales que se pueden aplicar a una cadena para realizar diferentes operaciones o manipulaciones. Te invitamos a que ejercites algunos de estos métodos:

### 1. 'upper()': Convierte todos los caracteres de la cadena a mayúsculas

```
cadena = "hola, mundo!"
cadena_mayusculas = cadena.upper()
print(cadena_mayusculas) # Salida esperada: HOLA, MUNDO!
```

### 2. 'lower()': Convierte todos los caracteres de la cadena a minúsculas

```
cadena = "HOLA, MUNDO!"
cadena_minusculas = cadena.lower()
print(cadena_minusculas) # Salida esperada: hola, mundo!
```

### 3. 'capitalize()': Convierte el primer carácter de la cadena a mayúscula y el resto a minúsculas

```
cadena = "hola, mundo!"
cadena_capitalizada = cadena.capitalize()
print(cadena_capitalizada) # Salida esperada: Hola, mundo!
```

### 4. 'strip()': Elimina los espacios en blanco al inicio y al final de la cadena

```
cadena = "  hola, mundo!  "  
cadena_sin_espacios = cadena.strip()  
print(cadena_sin_espacios) # Salida esperada: hola, mundo!
```

5. 'split()': Divide la cadena en una lista de subcadenas utilizando un separador

```
cadena = "Hola,amigo,cómo,estás?"  
subcadenas = cadena.split(",")  
print(subcadenas) # Salida esperada: ['Hola', 'amigo', 'cómo', 'estás?']
```

6. 'replace()': Reemplaza todas las ocurrencias de una subcadena por otra

```
cadena = "Hola, mundo!"  
cadena_reemplazada = cadena.replace("mundo", "amigo")  
print(cadena_reemplazada) # Salida esperada: Hola, amigo!
```

7. 'startswith()': Verifica si la cadena comienza con una subcadena específica

```
cadena = "Hola, mundo!"  
comienza_con_hola = cadena.startswith("Hola")  
print(comienza_con_hola) # Salida esperada: True
```

8. 'endswith()': Verifica si la cadena termina con una subcadena específica

```
cadena = "Hola, mundo!"  
termina_con_exclamacion = cadena.endswith("!")  
print(termina_con_exclamacion) # Salida esperada: True
```

9. 'find()': Busca la primera ocurrencia de una subcadena en la cadena y devuelve su índice

```
cadena = "Hola, mundo!"  
indice = cadena.find("mundo")  
print(indice) # Salida esperada: 6
```

10. 'count()': Devuelve el número de ocurrencias de una subcadena en el string

```
cadena = "Hola, hola, hola!"  
ocurrencias =  
cadena.count("hola")  
print(ocurrencias) # Salida  
esperada: 3
```

Estos son solo algunos ejemplos de los muchos métodos disponibles para manipular cadenas en **Python**.

Ya estamos llegando al final de [nuestro viaje](#) y estás más cerca de **conseguir tu ticket to code y convertirte en especialista en datos**. Para lograrlo, solo tienes que realizar el Henry Challenge y elegir en qué fecha quieras comenar la Carrera de Data Science.

*Para eso, te compartimos algunos secretos que facilitarán tu preparación para este examen técnico, ¿dónde? en [nuestra parada final](#)??*

*¿Quieres acceder a nuestro Curso de Introducción a Python? Ingresas [aquí](#) y da tus primeros pasos en tecnología?*

### Etiquetas

- [Data Science](#)
- [Henry](#)

### Recomendado para ti



[Desarrollo Web](#)

[Programación en la era de la inteligencia artificial: ¿Cuál es el futuro de los desarrolladores y...](#)

[Analizamos el rol de los devs.](#)

[hace un mes • 8 min de lectura](#)