



El futuro digital
es de todos

MinTIC

Ciclo 1

Fundamentos de programación con Python

Sesión 14: Uso de Strings

Programa Ciencias de la Computación e Inteligencia Artificial
Escuela de Ciencias Exactas e Ingeniería
Universidad Sergio Arboleda
Bogotá



UNIVERSIDAD
SERGIO ARBOLEDA





Agenda

1. Introducción
2. Indexación
3. Operaciones con Strings
4. ASCII
5. UNICODE
6. Métodos de Strings
7. Ejercicios





1. Introducción

- Las cadenas (o strings) son un tipo de datos compuestos por secuencias de caracteres que representan texto. Estas cadenas de texto son de tipo *str* y se delimitan mediante el uso de comillas simples o dobles.





1. Introducción

- En el caso que queramos usar comillas (o un apóstrofo) dentro de una cadena tenemos distintas opciones. La más simple es encerrar nuestra cadena mediante un tipo de comillas (simples o dobles) y usar el otro tipo dentro de la cadena. Otra opción es usar en todo momento el mismo tipo de comillas, pero usando la barra invertida (\) como carácter de escape en las comillas del interior de la cadena para indicar que esos caracteres forman parte de la cadena.





1. Introducción

- El intérprete de Python tiene incorporadas distintas funciones relacionadas con las cadenas. Una de ellas es la función `len()`, que nos indica el número de elementos de un objeto. En el caso de que el objeto sea una cadena nos indica el número de caracteres que la componen.





2. Indexación de Strings

Cada uno de los caracteres de una cadena (incluidos los espacios) tiene asignado un índice. Este índice nos permite seleccionar su carácter asociado haciendo referencia a él entre corchetes ([]) en el nombre de la variable que almacena la cadena. Si consideremos el orden de izquierda a derecha, el índice empieza en 0 para el primer carácter, etc. También se puede considerar el orden de derecha a izquierda, en cuyo caso al último carácter le corresponde el índice -1, al penúltimo -2 y así sucesivamente





3. Operaciones con String

Una operación que podemos realizar es la concatenación que consiste en unir distintas cadenas mediante el uso del signo más (+).

También podemos concatenar con el signo de multiplicación (*), que en este caso significa adjuntarle un determinado número de copias a la cadena.





3. Operaciones con String

➤ Comparación:

- Se usan los operadores convencionales (<, <=, >, >=, ==, !=) para comparar cadenas usando el orden lexicográfico.
- En el orden lexicográfico, se comparan de izquierda a derecha uno a uno los caracteres, mientras sean iguales.
- En el caso que no sean iguales, si el carácter de la primera cadena es menor que el de la segunda a la primer cadena se le considera menor, pero si es mayor, a la primer cadena se le considera mayor.
- Si todos los caracteres son iguales, se considera que las cadenas son iguales





4. ASCII

- ❖ Código Estadounidense Estándar para el Intercambio de Información (American Standard Code for Information Interchange)
- En su versión original usa 7 bits, definiendo 128 caracteres.
- En la versión extendida usa 8 bits (esto es 1 byte), definiendo 256 caracteres.
- Es la base de los archivos de texto plano (o sin formato).
- Es el esquema base para la escritura de programas en casi todos los
- lenguajes de programación (incluido Python).





5. UNICODE

Esquema de codificación cuyo objetivo es dar a cada carácter usado por cada uno de los lenguajes humanos su propio código, es decir, permitir la “internacionalización” de la computación.

- **UTF – 8:** definido por ocho (8) bits (un byte). Toma como base el ASCII, ANSI de Windows y el ISO – 8859 – 1. Muy usado en HTML.
- **UTF – 16:** definido por 16 bits (2 bytes). Usa una representación de longitud variable que permite su optimización en procesos de codificación a texto (usando un subconjunto de (ASCII o UTF – 8).
- **UTF-32:** definido por 32 bits (4 bytes). Es el m´as simple pues usa una representación de longitud fija





6. Métodos de String

- Longitud (len): longitud de una cadena
- Subcadena (slice): obtiene una cadena de otra cadena
- Contando (count): cuenta cuantos caracteres hay en una cadena
- Buscando (find, rfind): obtiene la primera y ultima ocurrencia en una cadena
- Mayúscula y minúsculas: s.lower() s.upper()
- Removiendo caracteres (strip, lstrip, rstrip)
- Dividiendo cadenas (split)
- Reemplazando (replace)





7. Ejercicios

- Realizar un programa que comprueba si una cadena leída por teclado comienza por una subcadena introducida por teclado.
- Pide una cadena y un carácter por teclado (valida que sea un carácter) y muestra cuantas veces aparece el carácter en la cadena.
- Suponiendo que hemos introducido una cadena por teclado que representa una frase (palabras separadas por espacios), realiza un programa que cuente cuantas palabras tiene.
- Pide una cadena y dos caracteres por teclado (valida que sea un carácter), sustituye la aparición del primer carácter en la cadena por el segundo carácter.





Preguntas

