



Pattern Recognition
and Applications Lab

Zucchero Sintattico in Python

Docente: Ambra Demontis

Anno Accademico: 2020 - 2021

Corso di Laurea in Ingegneria Elettrica, Elettronica e Informatica



University of Cagliari,
Italy

Department of Electrical and
Electronic Engineering



Zucchero Sintattico in Python

In Python esistono diversi **costrutti sintattici che non hanno effetto sull'espressività del linguaggio ma che forniscono un modo per sintetizzare altre istruzioni** rendendole più veloci da leggere e scrivere.

Zucchero Sintattico in Python

Spesso capita di dover **modificare il valore di una variabile effettuando una operazione che coinvolge il suo valore.**

Ad esempio, supponete di avere una variabile contatore chiamata *cont* e di voler incrementare il suo valore di 1.

Dovreste scrivere:

```
cont = cont + 1
```

Zuccherò Sintattico in Python

Questo tipo di operazioni possono essere abbreviate utilizzando degli operatori ad-hoc.

Ad esempio, l'istruzione `cont = cont + 1`

Può essere scritta in modo più sintetico utilizzando l'operatore `+=`

`cont += 1`

Questo tipo di operatori, prima, esegue l'operazione utilizzando il valore della variabile come primo operando, poi, assegna il risultato alla variabile.

Zuccherò Sintattico in Python

Operatore	Sintassi	Significato
<code>+=</code>	<code>a += b</code>	<code>a = a + b</code>
<code>-=</code>	<code>a -= b</code>	<code>a = a - b</code>
<code>*=</code>	<code>a *= b</code>	<code>a = a * b</code>
<code>/=</code>	<code>a /= b</code>	<code>a = a / b</code>
<code>%=</code>	<code>a %= b</code>	<code>a = a % b</code>
<code>//=</code>	<code>a //= b</code>	<code>a = a // b</code>
<code>**=</code>	<code>a **= b</code>	<code>a = a ** b</code>

Zuccherio Sintattico in Python

Un'altro operatore utile per “sintetizzare” il codice che un programmatore deve scrivere è l'operatore **in**, utilizzato con la seguente sintassi:

<elemento> in <oggetto_iterabile>

Questo operatore può essere utilizzato con le strutture dati built-in che abbiamo visto durante il corso (tuple, liste, stringhe, dizionari) e con tutti gli oggetti iterabili definiti dal programmatore.

L'espressione scritta sopra, restituisce: True, se <elemento> è uguale ad uno qualsiasi degli oggetti restituiti dalla funzione next applicata all'oggetto iterabile, False altrimenti.

Zucchero Sintattico in Python

```
lista = [1,2,3]
```

```
print(3 in lista)
```

Stampa:

True

```
print(50 in lista)
```

Stampa:

False

Zucchero Sintattico in Python

```
dizionario = {"nome": "Anna", "cognome": "Bianchi"}
```

```
print("nome" in dizionario)
```

Stampa:

True

```
print("Anna" in dizionario)
```

Stampa:

False

Zuccherio Sintattico in Python

```
stringa = "LPO - with Python 3"
```

```
print("L" in stringa)
```

Stampa:

True

```
print("l" in stringa)
```

Stampa:

False

Zuccherio Sintattico in Python

Proviamo ora ad utilizzarlo con un oggetto iterabile creato dal programmatore.

Consideriamo la classe *CNuovaLista*, definita nella lezione riguardo agli oggetti iteratori e iterabili.

Questa classe utilizza oggetti iteratori che restituiscono a due a due gli elementi della lista fino a che ci sono almeno due elementi da restituire.

Zuccherò Sintattico in Python

```
l = [1,2,3,4,5]
```

```
oggetto_nuova_lista = CNuovaLista(l)
```

```
for elems in oggetto_nuova_lista:  
    print(elems)
```

Stampa:

```
[1, 2]
```

```
[3, 4]
```

Zuccherio Sintattico in Python

Proviamo ad utilizzare l'operatore `in` con l'oggetto iterabile di tipo `CNuovaLista` creato nella slide precedente.

```
print([1,2] in oggetto_nuova_lista)
```

Stampa:

True

```
print([2,3] in oggetto_nuova_lista)
```

Stampa:

False