Python les-materialen Yilmaz Mustafa, Instructeur Java/Python

# Dictionaries (Woordenboeken)

We hebben geleerd over *sequencen* in Python, maar nu gaan we een versnelling hoger schakelen (switchen) en leren over *mappings toewijzingen)* in Python. Als u bekend bent met andere talen, kunt u deze dictionaries zien als ‘hash-tabellen’.

Dit gedeelte dient als een korte inleiding tot woordenboeken en bestaat uit:

1.) Een dictionary bouwen  
 2.) Objecten openen vanuit een dictionary  
 3.) Geneste dictionaries  
 4.) Basis-dictionary-methoden

Dus wat zijn mappings? Mappings zijn een verzameling objecten die worden opgeslagen door een *key (sleutel)*, in tegenstelling tot een reeks/sequence die objecten opslaat op hun relatieve positie. Dit is een belangrijk onderscheid, aangezien toewijzingen de volgorde niet behouden omdat ze objecten hebben die zijn gedefinieerd door een sleutel.

Een Python-dictionary bestaat uit een key (sleutel) en vervolgens een bijbehorende value (waarde). Die waarde kan bijna elk Python-object zijn.

## Een dictionary/woordenboek aanmaken

Laten we eens kijken hoe we woordenboeken kunnen aanmaken om een beter begrip te krijgen van hoe ze werken!

# Maak een dictionary/woordenboek met {} en : om een sleutel en een waarde aan te duiden  
my\_dict = {'key1':'value1','key2':'value2'}

# Roep values (waarden) met hun key (sleutel)  
my\_dict['key2']

'value2'

Het is belangrijk op te merken dat woordenboeken zeer flexibel zijn in de gegevenstypen die ze kunnen bevatten. Bijvoorbeeld:

my\_dict = {'key1':123,'key2':[12,23,33],'key3':['item0','item1','item2']}

# Laten we items uit het woordenboek accessen  
my\_dict['key3']

['item0', 'item1', 'item2']

# Kan een index aanroepen op die value (waarde)  
my\_dict['key3'][0]

'item0'

# Kan dan zelfs methoden op die value (waarde) aanroepen  
my\_dict['key3'][0].upper()

'ITEM0'

We kunnen ook invloed hebben op de waarden van een sleutel. Bijvoorbeeld:

my\_dict['key1']

123

# Trek 123 af van de value (waarde)  
my\_dict['key1'] = my\_dict['key1'] - 123

#Controleren  
my\_dict['key1']

0

Een snelle opmerking, Python heeft een ingebouwde methode om zelf af te trekken of op te tellen (of vermenigvuldigen of delen). We hadden ook += of -= kunnen gebruiken voor de bovenstaande verklaring. Bijvoorbeeld:

# Stel het object gelijk aan zichzelf min (-) 123  
my\_dict['key1'] -= 123  
my\_dict['key1']

-123

We kunnen ook sleutels maken op basis van toewijzing. Als we bijvoorbeeld zouden beginnen met een leeg woordenboek, zouden we er voortdurend aan kunnen toevoegen:

# Maak een nieuw dictionary/woordenboek aan  
d = {}

# Maak een nieuwe sleutel via toewijzing (assignment) aan  
d['animal'] = 'Dog'

# Kan dit met elk object doen  
d['answer'] = 42

#Tonen  
d

{'animal': 'Dog', 'answer': 42}

## Geneste dictionaries/woordenboeken

Hopelijk begin je te zien hoe krachtig Python is met zijn flexibiliteit om objecten te nesten en methoden erop aan te roepen. Laten we een woordenboek bekijken dat in een woordenboek is genest:

# Woordenboek genest in een woordenboek genest in een woordenboek  
d = {'key1':{'nestkey':{'subnestkey':'value'}}}

Wauw! Dat is nogal een begin van woordenboeken! Laten we eens kijken hoe we die value (waarde) kunnen grijpen:

# Blijf het key (sleutel) aan te roepen  
d['key1']['nestkey']['subnestkey']

'value'

## Een paar woordenboekmethoden

Er zijn een paar methoden waarop we een woordenboek kunnen gebruiken. Laten we een korte introductie geven tot een paar van hen:

# Maak een typisch woordenboek aan  
d = {'key1':1,'key2':2,'key3':3}

# Methode om een lijst van alle sleutels terug te geven (retourneren)  
d.keys()

dict\_keys(['key1', 'key2', 'key3'])

# Methode om alle waarden op te halen  
d.values()

dict\_values([1, 2, 3])

# Methode om tupels van alle items te retourneren (we zullen binnenkort meer leren over tupels)  
d.items()

dict\_items([('key1', 1), ('key2', 2), ('key3', 3)])

Hopelijk heb je nu een goed basisbegrip voor het maken van dictionaries/woordenboeken. Er is hier nog veel meer om op in te gaan, maar we zullen de woordenboeken later opnieuw bekijken. Na dit gedeelte hoeft u alleen maar te weten hoe u een woordenboek kunt maken en hoe u er waarden uit kunt halen.