



University of  
Zurich <sup>UZH</sup>

Institute of Computational Linguistics

# Programmiertechniken in der Computerlinguistik I

Herbstsemester 2015

5. Sitzung, 22. Oktober 2015

Martin Volk



University of  
Zurich <sup>UZH</sup>

Institute of Computational Linguistics

## Hashes (= Python Dictionaries)

	Examples
Define a hash (= dictionary) - with two entries	<code>my_hash = {}</code> <code>my_hash = {'uni': 'noun', 'gute': 'adj'}</code>
Enter a key – value pair	<code>my_hash['mit'] = 'praep'</code>
Check if a key is in the hash - and get the value	<code>my_hash.has_key(my_word)</code> <code>my_word in my_hash</code> <code>my_hash.get(my_word)</code>
Loop over a hash	<code>for my_word in my_hash.keys()</code> <code>for my_word in my_hash</code>



## Hashes (= Python Dictionaries)


	Examples
get all <b>keys</b> into a list	<code>my_list = my_hash.keys()</code>
get all <b>values</b> into a list	<code>my_list = my_hash.values()</code>
get the <b>number</b> of keys (= entries)	<code>my_length = len(my_hash)</code>
get all key-value <b>tuples</b> into a list	<code>my_list = my_hash.items()</code>



## Data structure: Hash (= Python Dictionary)

value	'PRON'	'is'	'magic'	'.'	'20.'	2011
key	'this'	2	'noun'	3	'day'	'year'

my\_hash                      my\_hash['noun']



**University of Zurich**  
 Institute of Computational Linguistics

### Typical Hash (for word → PoS tag)

value	'PDT'	'NN'	'IN'	'VB'	'NN'	'NN'
key	'this'	'car'	'for'	'see'	'day'	'ball'

my\_PoS\_hash


Page 5


**University of Zurich**  
 Institute of Computational Linguistics

### Hashes (= Python Dictionaries): Typical Usages

	Examples
word + PoS tag	my_hash = {'der': 'ART', 'in': 'APPR', 'Auto': 'NN'}
item + frequency	my_hash = {'der': 4, 'in': 12, 'Auto': 2} my_hash = {'ART': 7, 'NN': 12, 'APPR': 8}
word + translation	my_hash = {'der': 'the', 'in': 'in', 'Auto': 'car', 'Haus': 'house'}
string + dummy value (for quick access)	my_hash = {'der': 1, 'in': 1, 'Auto': 1}

Page 6




University of  
Zurich <sup>UZH</sup>

Institute of Computational Linguistics

## List vs. Hash

<h3>List</h3> <p><code>my_list = []</code></p> <p>stores a sequence of values <math>x_1, x_2, x_3 \dots</math></p> <p>preserves order</p> <p>quick access via number key to value, but <b>slow</b> access to a particular value</p>	<h3>Hash</h3> <p><code>my_hash = {}</code></p> <p>stores relations <math>x_1 \rightarrow y_1</math> <math>x_2 \rightarrow y_2</math></p> <p>random order !</p> <p>quick access via key to value</p>
---	---

Page 7



University of  
Zurich <sup>UZH</sup>

Institute of Computational Linguistics

## Regular Expressions Search and Substitution

	Examples
Regular expression matching	<pre>import re my_match = re.search('abc', my_text) my_match = re.search('^waa..', my_text) print my_match.group()</pre>
Get hit from within a match	<pre>my_match = re.search('ab(lw+)', my_text) print my_match.group(1)</pre>
Get multiple matches in a list	<pre>my_matches = re.findall('a\w+', my_text)</pre>
Find and substitute	<pre>my_new_text = re.sub('t', 'th', my_text)</pre>

Page 8



## Miscellaneous

	Examples
Defining functions - the main function	<code>def my_function():</code> <code>def main():</code>
Global variables (defined outside a function)	
Remove <b>leading</b> (left) “whitespace” characters	<code>my_line.lstrip()</code>
Remove <b>trailing</b> (right) “whitespace” characters	<code>my_line.rstrip()</code>

Page 9



## Programming Techniques in this Lesson

- Filtering a text and writing output to multiple files
- Person name recognition and gender statistics
- Reading information from a file and applying it to a text
- Word by word machine translation

Page 10