



Programming for Data Science: Set and Dict

Instructor: Jakramate Bootkrajang

Recap:

- Dictionary construction

```
>>> d = {}  
>>> d["key1"] = "value1"  
>>> d["key2"] = "value2"  
>>> d
```

Dictionary indexing

- To get values from dictionary

```
>>> d["key1"]  
>>> d["key2"]
```

Checking if key exists

```
>>> d = { "key1" : "value1", "key2" : "value2" }
>>> "key1" in d
>>> "key3" in d
```

Removing key/value pair

```
>>> d = { "key1" : "value1", "key2" : "value2" }  
>>> del d["key1"]  
>>> d
```

Looping with for

```
d = { "key1":1, "key2":2, "key3":1, "key4":3, "key5":1, "key6":4, "key7":2 }  
  
for k in d :  
    print("key=", k, " value=", d[k], sep="")
```

Exercise 1

- Write a function called *accept_login(users, username, password)* with three parameters:
 - users a dictionary of username keys and password values,
 - username a string for a login name
 - and password a string for a password.
- The function should return **True** if the user exists and the password is correct

Code tester

```
users = {
    "user1" : "password1",
    "user2" : "password2",
    "user3" : "password3"
}

if accept_login(users, "wronguser", "wrongpassword") :
    print("login successful!")
else :
    print("login failed...")
```


Exercise 2

- Write a function called

word_frequencies(mylist)

that accepts a list of strings called `mylist` and returns a dictionary where the keys are the words from `mylist` and the values are the number of times that word appears in `mylist`:

Code tester

```
word_list = list("aaaaabbbbccdde")
word_freq = { 'a' : 5, 'b' : 4, 'c' : 3, 'd' : 2, 'e' : 1 }

if word_frequencies(word_list) == word_freq :
    print("correct")
else :
    print("wrong")
```

Exercise 3

- Write a character counting function called *char_count(string)*

that takes a string as input and output the number of distinct characters found in the string

- Example

```
char_count("ABCD")    # print 4
```