

Session 3:

Strings and APIs

Andreas Bjerre-Nielsen

Agenda

1. Strings: manipulation, combination etc.
2. Containers - key based
3. Interacting with the web
4. Loading and saving files

Strings

Strings recap

What are strings? What do they consist of?

- Strings are sequences of characters
 - Characters can be whitespace.
- Python has two formats
 - `ascii` strings contain only English characters
 - `utf` strings contain also other European and Asian characters

String concatenation

How can I combine strings?

Strings can be added together:

```
In [ ]: s1 = 'police'  
        s2 = 'officer'  
        # s1 + s2  
        # s1 + ' ' + s2
```

String changing case

Can I alter the sentence-case of strings?

- Yes using the string methods `upper`, `lower`, `capitalize`. Example:

In []: `# s1.capitalize()`

Substrings (1)

How can I check if a substring is contained in the string?

- in/not in

```
In [ ]: 'pol' in s1
```

Substrings (2)

How can I replace a specific substring?

- replace

```
In [ ]: s1.replace('po', 'ma')
```


Substrings (3)

Can I also access a string via indices? (in the sequence of characters)

- sequence form - slicing/indexing

```
In [ ]: s1[3:]
```

Strings quiz

Which Python object do strings remind you of?

- Lists work like strings.
 - Concatention (+, *) works the same way.
 - We check if element/character is contained with `in`.
 - We can slice and use indices for.

More about strings

There are many things about strings which we have not covered:

- Methods for splitting or combining strings etc.
- String formatting (http://www.python-course.eu/python3_formatted_output.php) is exceptionally useful, e.g for making URLs, printing etc.

Containers - key based

Containers recap

What are containers? Which have we seen?

-

Dictionaries (1)

How can we make a container which is accessed by arbitrary keys?

By using a dictionary, dict. Try executing the code below:

```
In [ ]: my_dict = {'Andreas': 'Economist',  
                  'Snorre': 'Sociologist',  
                  'Ulf': 'Engineer'}  
# print(my_dict['Snorre'])
```

Dictionaries (2)

Dictionaries can also be constructed from two associated lists. These are tied together with the `zip` function. Try the following code:

```
In [ ]: keys = ['a','b']  
        values = [1,2]  
        key_value_pairs = list(zip(keys, values))  
  
        my_dict2 = dict(key_value_pairs)  
        print(my_dict2['b'])
```

Storing containers

Does there exist a file format for easy storage of containers?

Yes, the JSON file format.

- Is at the base a list or a dict.
- Looks like one:
 - `'{"a":1,"b":1}'`

Storing containers (2)

Why is JSON so useful?

- Extreme flexibility:
 - Can hold any list or dictionary of any depth which contains only float, int, str.
- Standard format that looks exactly like Python.

Interacting with the web

The web protocol

What is http and where is it used?

- http stands for HyperText Transfer Protocol.
- http is good for transmitting the data when a webpage is visited:
 - the visiting client sends request for URL or object;
 - the server returns relevant data if active.

The web protocol (2)

Should we care about http?

- In this course we don't care explicitly about http.
- We use a Python module called `requests` as a http interface.
- However... Some useful advice - you should **always**:
 - use the encrypted version, https;
 - use authenticated connection, i.e. private login, whenever possible.

Markup language (1)

What is html and where is it used?

- HyperText Markup Language
- html is a language for communicating how a webpage looks like and behaves.
 - That is, html contains: content, design, available actions.

Markup language (2)

Should we care about `html`?

- Yes, `html` is often where the interesting data can be found.
- Sometimes, we are lucky, and instead of `html` we get a JSON in return.
- Getting data from `html` will be the topic of the subsequent scraping sessions.

Web APIs (1)

So when do we get lucky, i.e. when is `html` not important?

- When we get an Application Protocol Interface, i.e. API
- What does this mean?
 - We send an API query
 - We get an API response with data back in return, typically as JSON.

Web APIs (2)

So is data free? As in free lunch?

- Most commercial APIs require authentication and have limited free usage
 - e.g. Google Maps, various weather services
- If no authentication is required the API may be delimited.
 - This means only a certain number of requests can be handled per second or per hour from a given IP address.

Web APIs (3)

So how do make the URLs?

- An API query is a URL consisting of:
 - Server URL, e.g. `https://api.github.com`
 - Endpoint path, `/users/abjer/repos`
 - Query parameters,

Web APIs in Python (1)

How do make a simple query?

```
In [ ]: server_url = 'https://api.github.com/'  
        endpoint_path = 'users/abjer/repos'  
        url = server_url + endpoint_path
```

Web APIs in Python (2)

How can we send a query with the requests module?

```
In [ ]: import requests # import the module  
        response = requests.get(url) # submit query with `get` and save response
```

Web APIs in Python (3)

How do extract something from the response?

```
In [ ]: # print(response.text[:500])
```

Web APIs in Python (4)

Can we get something more meaningful or structured?

```
In [ ]: # resp_json = resp.json()
```

Loading and saving files

How to do input-output (IO) operations in Python

Text files

How can we save a string as a text file?

```
In [ ]: my_str = 'This is important.'  
  
with open('my_file.txt', 'w') as f:  
    f.write(my_str)
```

How can we load a string from a text file?

```
In [ ]: with open('my_file.txt', 'r') as f:  
        my_str_load = f.read()  
        my_str == my_str_load
```

JSON files

How can we save a JSON file?

The trick is to convert the JSON file to a string. This can be done with `dumps` in the module `json`:

```
In [ ]: import json

with open('my_file.json', 'w') as f:
    response_json_str = json.dumps(response.json())
    f.write(response_json_str)
```

We can convert a string to JSON with `loads`.

File handling

How can we remove a file?

The module `os` can do a lot of file handling tasks:

```
In [ ]: import os  
        os.remove('my_file.json')
```

Exam projects

- You decide
- Show us tools you have used in this course
- Check out the exam post (<https://abjer.github.io/sds/post/exam/>), and the practical info (<https://abjer.github.io/sds/page/practical/>).

The end

[Return to agenda](#)