

Advanced Programming with Python

Pepe García jgarciah@faculty.ie.edu

Plan for today

- Review yesterday's

Plan for today

- Review yesterday's
- Dealing with request bodies

Plan for today

- Review yesterday's
- Dealing with request bodies
- Sending files from folder



Plan for today

- Review yesterday's
- Dealing with request bodies
- Sending files from folder
- working with HTML



HTTP request bodies

So far, we've been sending data in response bodies but haven't yet seen how to receive data from requests.

Something we'll need to consider is that not all HTTP verbs allow us to set request bodies:

verb	allows body?
GET	no
HEAD	no
DELETE	no
PUT	yes
PATCH	yes
POST	yes



HTTP request bodies

Getting request body (server)

```
from flask import request

@app.route("/get-body", methods = [ "POST" ])
def get_body():
    body = request.get_json()
    return "body received!" + str(body)
```



HTTP request bodies

Using request body (client)

```
import requests

dictionary = {
    "name": "dict",
    "purpose": "none at all"
}

url = "http://localhost:5000/get-body"

requests.post(url, json=dictionary)
```



Exercise

Let's create a very simple Twitter clone! This Twitter must have the following functionalities:

- POST /tweet: submit **tweets** `{"user": "pepe", "tweet": "Hello world"}`



Exercise

Let's create a very simple Twitter clone! This Twitter must have the following functionalities:

- POST /tweet: submit **tweets** `{"user": "pepe", "tweet": "Hello world"}`
- GET /: List all tweets

Serving static files

Being able to serve static files is vital for websites. They can be images, videos, CSS templates, or anything you imagine.

Serving static files

In flask we can serve static files using the **send_from_directory** function.

```
from flask import send_from_directory
```

```
@app.route("/images/<image>.png")
```

```
def serve_image(image):
```

```
    return send_from_directory(
```

```
        "images/",
```

```
        filename="{}.png".format(image))
```



HTML files, as files of any other type can be served using the **`send_from_directory`** function.

Serving HTML

```
@app.route("/")  
def index():  
    return send_from_directory("html", filename="index.html")
```



Exercise

Let's draft a Twitter for dogs website!

Create three routes for it: a homepage, an about page, and a login page.
(we don't need any functionality on these yet, only the HTML)

<https://realpython.com/python-requests/>