



Attendance:
tiny.cc/event-attendance





App Dev League

Day 2: Web Dev



Agenda

1. Installation
2. Web Requests
3. Flask Intro
4. Project



Installation

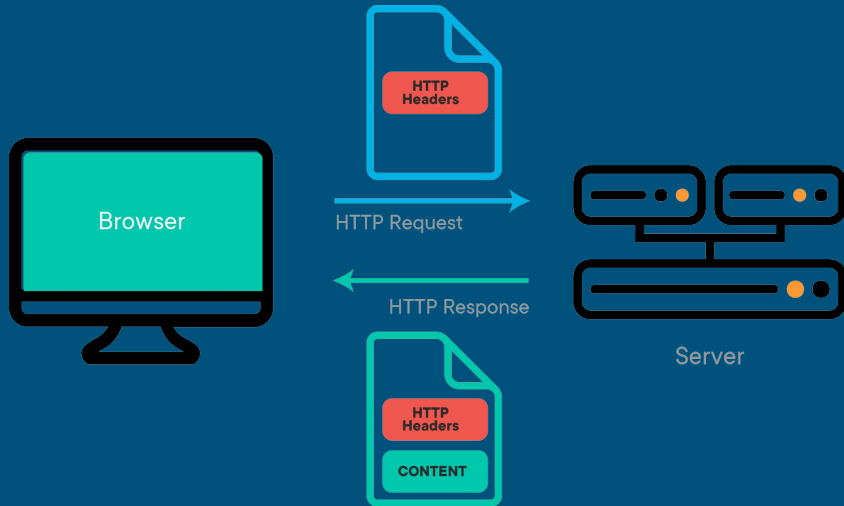


Web Requests



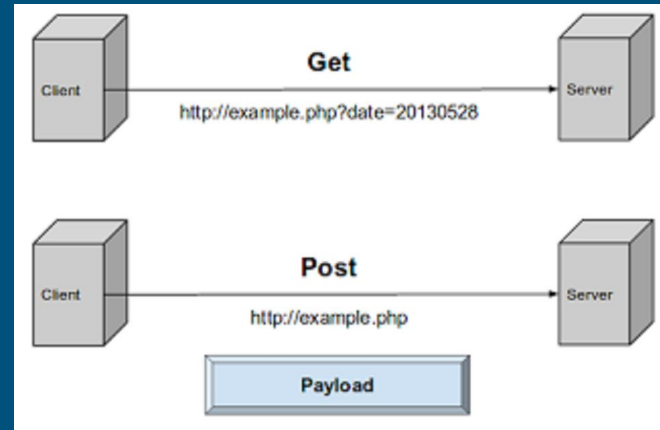
What are Web Requests?

- Browser sends HTTP Request (either GET or POST)
- Server replies with HTTP Response including the content

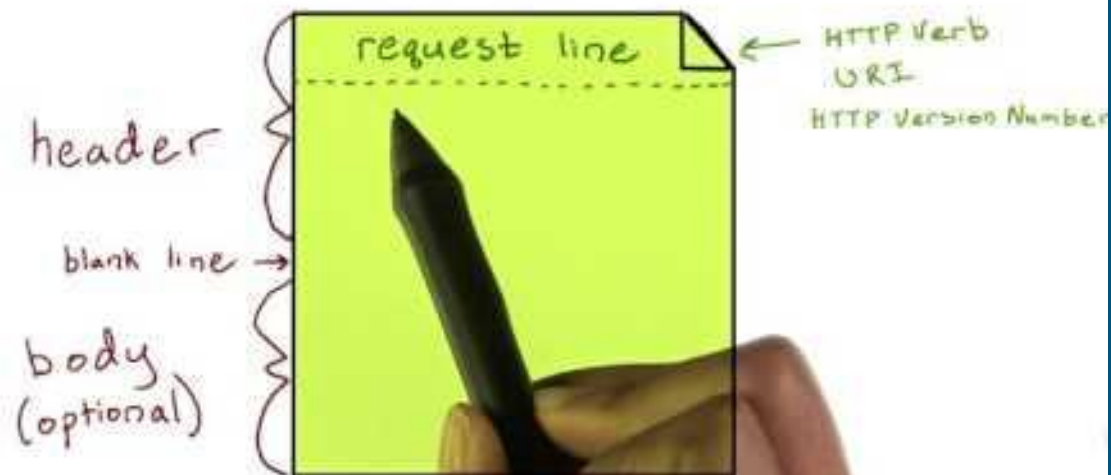


GET vs. POST Requests

- GET request: GETs webpage data (HTML, CSS, JavaScript files) from a website link/URL
- POST request: POSTs/sends data to the server (for example, sending messages) using both a website link/URL and a payload



HTTP Requests



Flask Intro

3



Flask Routes

- Open up server to GET and POST requests
- `@app.route` decorator with the path after the url
- Optional `methods` parameter to specify which methods

```
@app.route('/', methods=['GET', 'POST'])  
def home():  
    return render_template('index.html')
```

Flask Routes

- Then, you define a function using the “def” keyword, the name can be anything but it should describe something about that page (for example “home” or “about”)
- Within the function, a certain value is returned. This is the HTTP Response that is sent back to the browser.

Here, we use the “render_template” function to render an HTML file.

```
@app.route('/', methods=['GET', 'POST'])  
def home():  
    return render_template('index.html')
```

Advanced Routes

- Routes can also contain parameters, they can be specified using “<parameter_name>” and then passed as a parameter into the function
- Then, you can specify a parameter to the `render_template` function and access the value using `{{parameter_name}}` inside the HTML file

```
@app.route('/user/<username>')  
def generate_user(username):  
    return render_template('hello_world.html',  
                           user=username)
```

```
templates/hello_world.html x  
1  <!DOCTYPE HTML>  
2  
3  <html>  
4      <head>  
5          <title>Hello World</title>  
6      <body>  
7          <h1>Hello {{user}}.</h1>  
8      </body>  
9  </html>
```

POST Requests

- Pages may support both GET and POST requests
- You can check the type of method in `request.method`
- `request.form.get(field_name)` to get the data from an HTML “<form>” element through the “name” attribute
- There are other request data fields besides `request.form`, but we will not be covering them today

```
def homepage():  
    if request.method == 'POST':  
        message = request.form.get("message")  
        return render_template('sent.html',  
                               message=message)  
    else:  
        return render_template('index.html')
```

Project



Project

- We will be using the bootstrap contact form we built yesterday.
- We will integrate the contact form with Flask
- A GET request will be used to display the contact form
- A POST request will be used to send the messages

THANKS!

ANY QUESTIONS?

You can find more info @

- ◇ <https://www.appdevleague.org>
- ◇ <https://linktr.ee/AppDevLeague>

