

THE INTERNET

Outline

- History of Internet
- Define client and server
- What is HTTP Request
- URLs
- Request-Response Cycle
- Request in Detail, incl HTTP Verbs
- Response in Detail
- Postman Demo - used throughout

History of Internet

- Internet is network of computers connected by cables - network of many networks
- Networks connected through cables, & through underwater fiber optic network cables across continents
- Sharks are known for biting through cables
- Invented by military through DARPA
(Department of Advanced Research Projects Agency) and connected in 1969

History of Internet

- World Wide Web is service that exists on top of the internet
- Created in 1989 by Sir Tim Berners-Lee and colleagues at CERN (European Organization for Nuclear Research)
- www has webpages that:
 - Connect to each other via URLs and
 - Communicate via HTTP(S) protocol
 - Render content in HTML

Define Client and Server

- Client: Program or Browser, Displays Web Content
- Server: Sends Data to Client
- How They Communicate: via HTTP
- Point of all this:
 - Understand (high level) relationship between client and server
 - Understand what the client is sending to and expecting back from the server
 - Make informed decisions when you start building the server (for the next mod)

What is HTTP Request

- HTTP (Hypertext Transfer Protocol)
- Language of web
- Describes how webpages and files are sent over internet
- Can locate server to send request to via URL/URI
- Request consists of:
 - URL
 - Verb
 - header

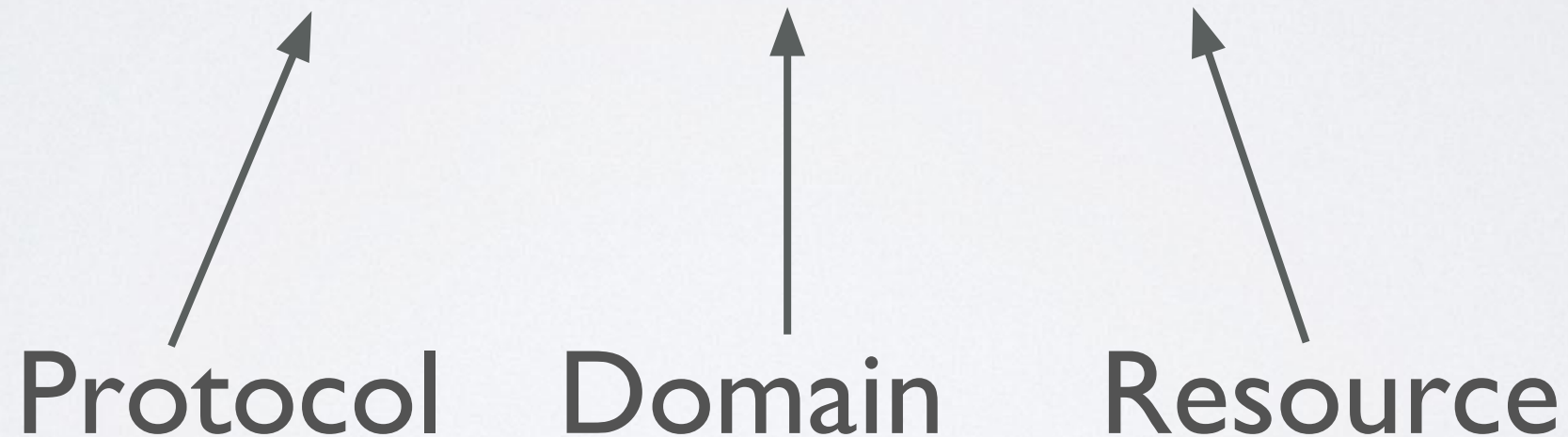
Taking a Look in Postman

Examining <https://swap.io/about> in Postman

- Postman used to examine communication between client and server
- Look at the code for this page
 - Don't have to understand it all, but you will in time!
- This is an example of a GET Request
 - Which will also be explained shortly

URL Breakdown

<https://swapi.co/about>

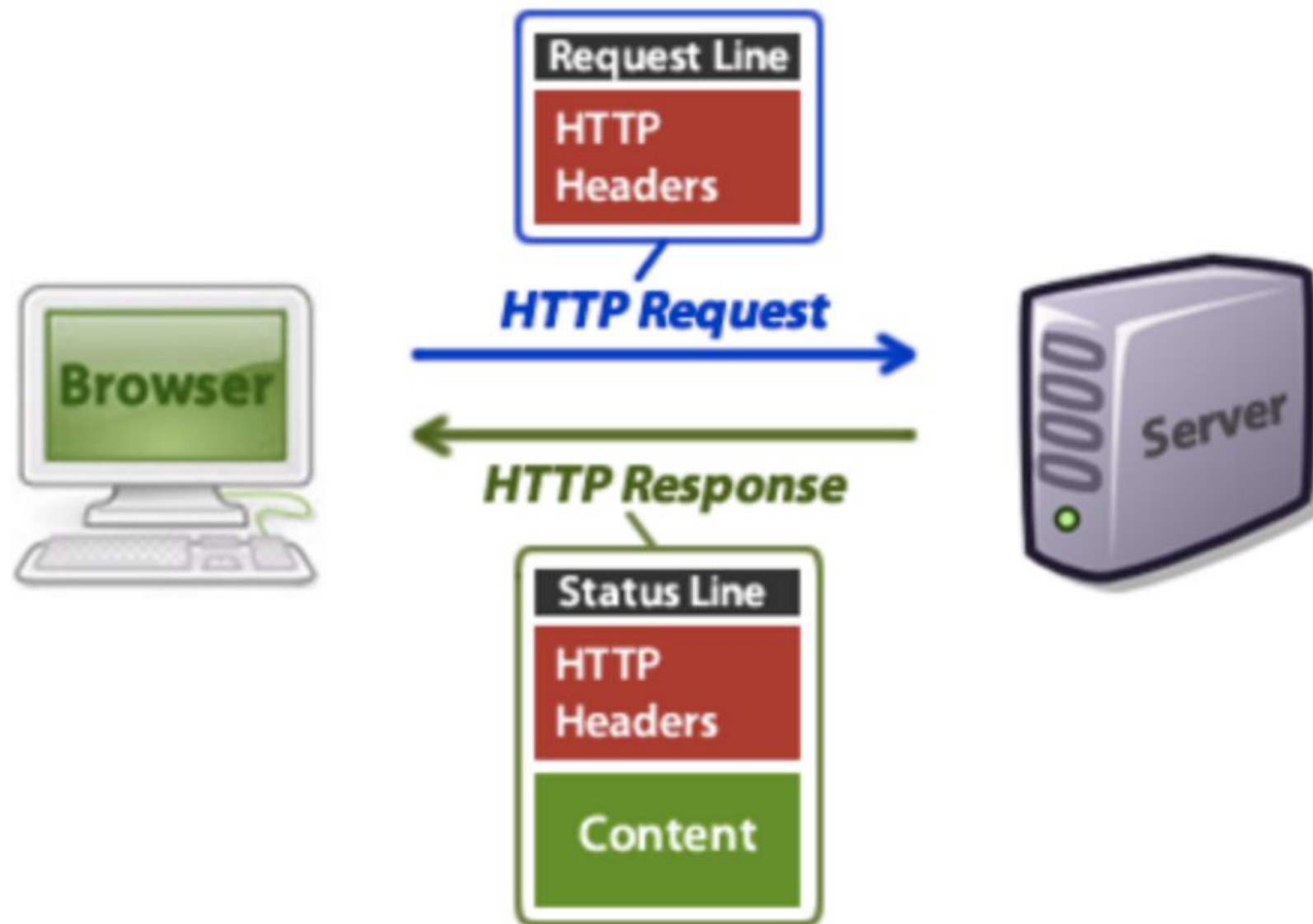


URL Breakdown: Search

<https://swapi.co/api/people/?search=r2>



Request Response Cycle

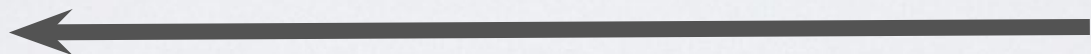
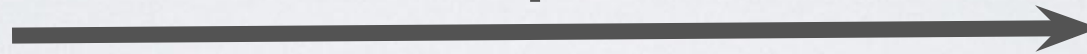


Source: net.tutsplus.com

Request Response Cycle



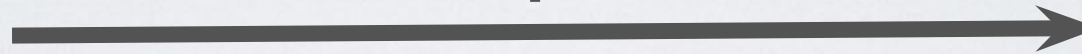
1. Browsers Make a
Request



2. Server Responds with
Content

Request Response Cycle

I. Browsers Make a Request



Request URL: http://localhost:3000/

Request Method: GET

Status Code:  200 OK

▼ Request Headers [view parsed](#)

GET / HTTP/1.1

Host: localhost:3000

Connection: keep-alive

Cache-Control: max-age=0

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_2) AppleWebKit/535.2 (KHTML, like Gecko) Chrome/15.0.874.106 Safari/535.2

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Encoding: gzip,deflate,sdch

Accept-Language: en-US,en;q=0.8

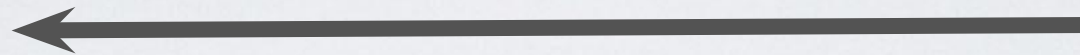
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3

Cookie: _mixtape_sample_session=BAh7CEkiD3Nlc3Npb25faWQ0gZFRkkiJTcyMzYxZmFjMzlhNzgxNzkxYTEzNmVmYTdkNDM1NzdlBjsAVEkiEF9jc3JmX3Rva2VuBjsARkkiMUxJQ0tiRkdQSzAzTEk5OGNSY05SKzUzN2FFdkYvZGpaYWNMcVpZVRHcUE9BjsARkkiCmZsYXNoBjsARm86JUFjdGlvbkrpc3BhdGNo0jpGbGFzaDo6Rmxhc2hIYXNoCToKQHVzZWRv0ghTZXQ0GpAaGFzaHsG0gtub3RpY2VU0gxAY2xvc2VkrjoNQGZsYXNoZXN7BjsKSSImTWl4dGFwZSB3YXMgc3VjY2Vzc2Z1bGx5IGNyZWZ0ZWQuBjsARjoJQG5vdzA%3D--0195ea67620a5d8ca936750ec70e238f19dd624b

If-Modified-Since: Sun, 06 Nov 2011 18:48:25 GMT

Request Response Cycle

2. Server Responds with Content

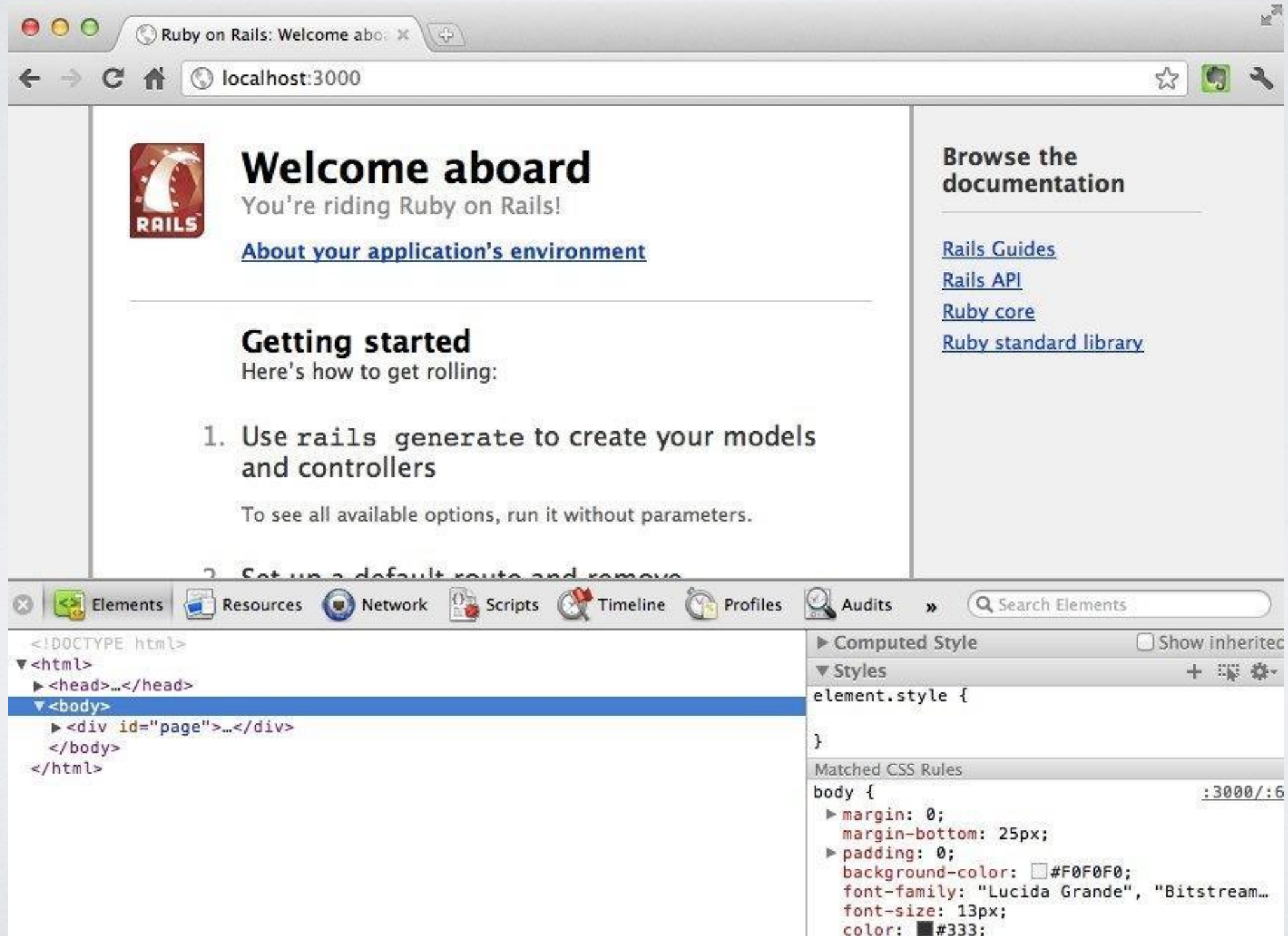


```
HTTP/1.1 200 OK
Last-Modified: Sun, 06 Nov 2011 18:48:25 GMT
Content-Type: text/html
Content-Length: 5907
Server: WEBrick/1.3.1 (Ruby/1.9.2/2011-07-09)
Date: Sun, 06 Nov 2011 18:50:01 GMT
Connection: Keep-Alive
```

```
<!DOCTYPE html>
<html>
  <head>
    <title>Ruby on Rails: Welcome aboard</title>
    <style type="text/css" media="screen">
      body {
        margin: 0;
        margin-bottom: 25px;
        padding: 0;
        background-color: #f0f0f0;
        font-family: "Lucida Grande", "Bitstream Vera Sans", "Verdana";
        font-size: 13px;
        color: #333;
      }
    </style>
  </head>
  <body>
```

Request Response Cycle

Servers send back HTML, browsers render Content



How Browser Renders Content

Looking at Postman again for <https://swap.io/about>

- What response do you get?
- What does it look like?

Rendered Content

- HTML defines **structure**
- CSS defines how page should look (**style**)
- JavaScript
 - Defines how page will behave once loaded (**behavior**)
 - How user interacts with page
- Can we identify what is the HTML, CSS, and JavaScript on this page?

HTTP Request in Detail

Try in Postman:

- <https://swapi.co/api/planets/1/> (that's a number 1)

View header in Chrome:

- View -> Developer -> JavaScript Console -> Network tab
- Can you see the HTTP verb “GET” in the header?

Review: CRUD

- What is CRUD?

Request Methods / Verbs

- Follow CRUD Model
- HTTP / CRUD:
 - GET : Read
 - POST: Create
 - PATCH: Update
 - DELETE: Destroy

When might we do?

- When might we use these different methods?
 - ie Facebook posts
- What request method are we using in last Postman request?

HTTP Response in Detail

- HTTP Response consists of :
 - Header (with response code)
 - Body
 - Data (like JSON) or
 - HTML/CSS
- Response Codes:
 - 100's: informational
 - 200's: success
 - 300's: redirects
 - 400's: error
 - 404 - file not found

HTTP Response in Detail

Try in Postman:

- <https://swapi.co/api/planets/1/> (that's a number 1)
- What does the body (JSON) look like?

View header in Chrome:

- View -> Developer -> JavaScript Console -> Network tab
 - What HTTP verb is in header?
 - What is the response code in header?

Quiz

Slack to me your answers: @Vui Nguyen

- What are the HTTP Request verbs?
- How do they correlate to CRUD?
- Label the parts of a URL:
 - <https://swapi.co/api/people/?search=r2>