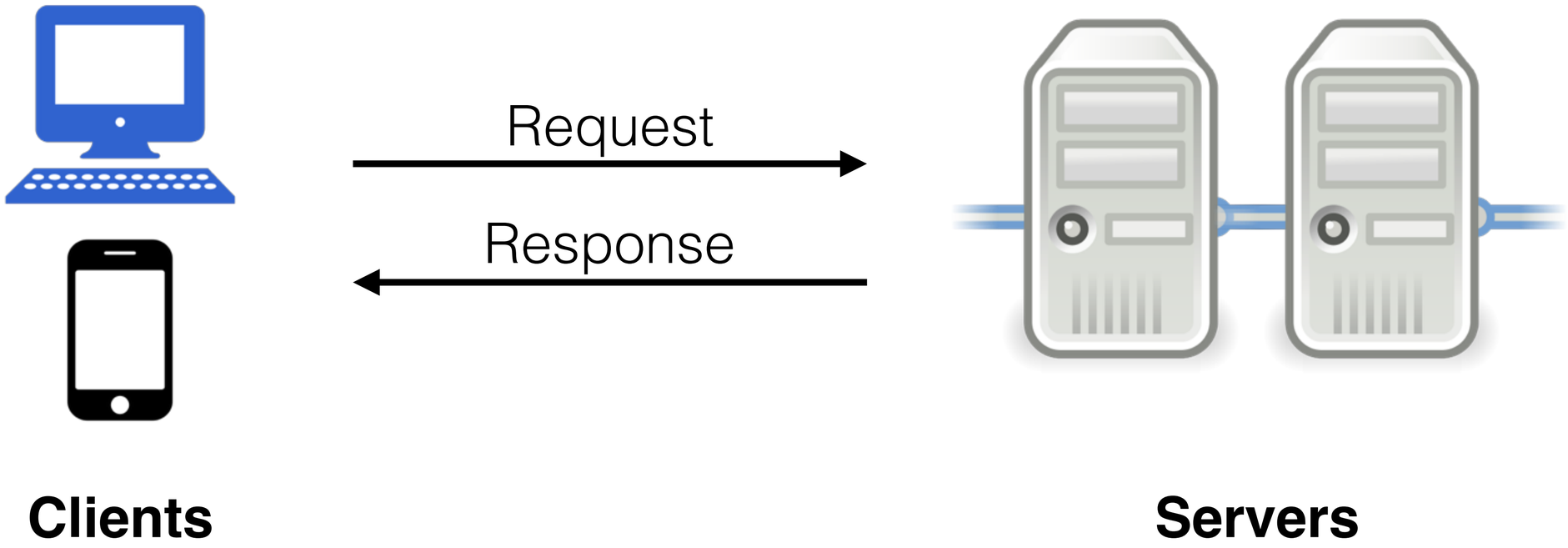
Introduction to Web

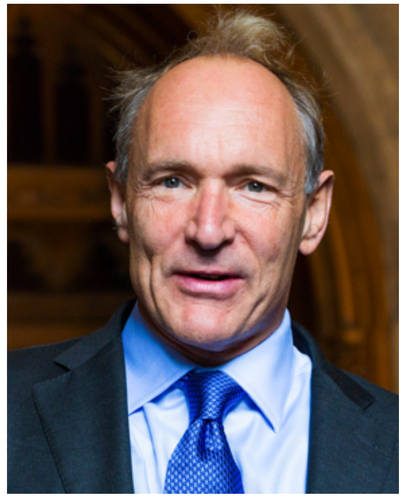
Zane Ma
University of Illinois
CS 461 / ECE 422 - Spring 2018

What is the Web?

Application layer on top of TCP/IP that follows a ***client-server model***



History of the Web



Designed by Tim Berners Lee to exchange text / papers between physicists at CERN

Basic text pages with uni-directional links

Embedded images - content from multiple servers loaded on a single client

Dynamic elements - executable JavaScript

Hardware access - camera, microphone, filesystem

No initial security considerations - bolted on!

Threat Model

Client and Server are programs that respond to input from each other

Malicious requests/responses can obtain privileged information, perform unauthorized actions, or run arbitrary code on the server

1. Malicious Client
   - Steal user records from database - Equifax hack
2. Malicious Server
   - Install malware (keylogger, botnet), violate user privacy
3. Man-in-the-Middle (applies to all network protocols, not just web)

HTTP

Hyper**t**ext **T**ransfer **P**rotocol - request/response mechanism (RFC 7230)

Web pages are identified by a global Uniform Resource Locator (URL)

http://courses.engr.illinois.edu:80/cs461/sp2018?user=admin#grading

Protocol **Host** **Port** **Path** **Query** **Fragment**

HTTP Request Methods

GET – Requests data from a resource

- GET youtube.com/videos/32410 - returns video # 32410

POST – Submits data to be processed to a URL

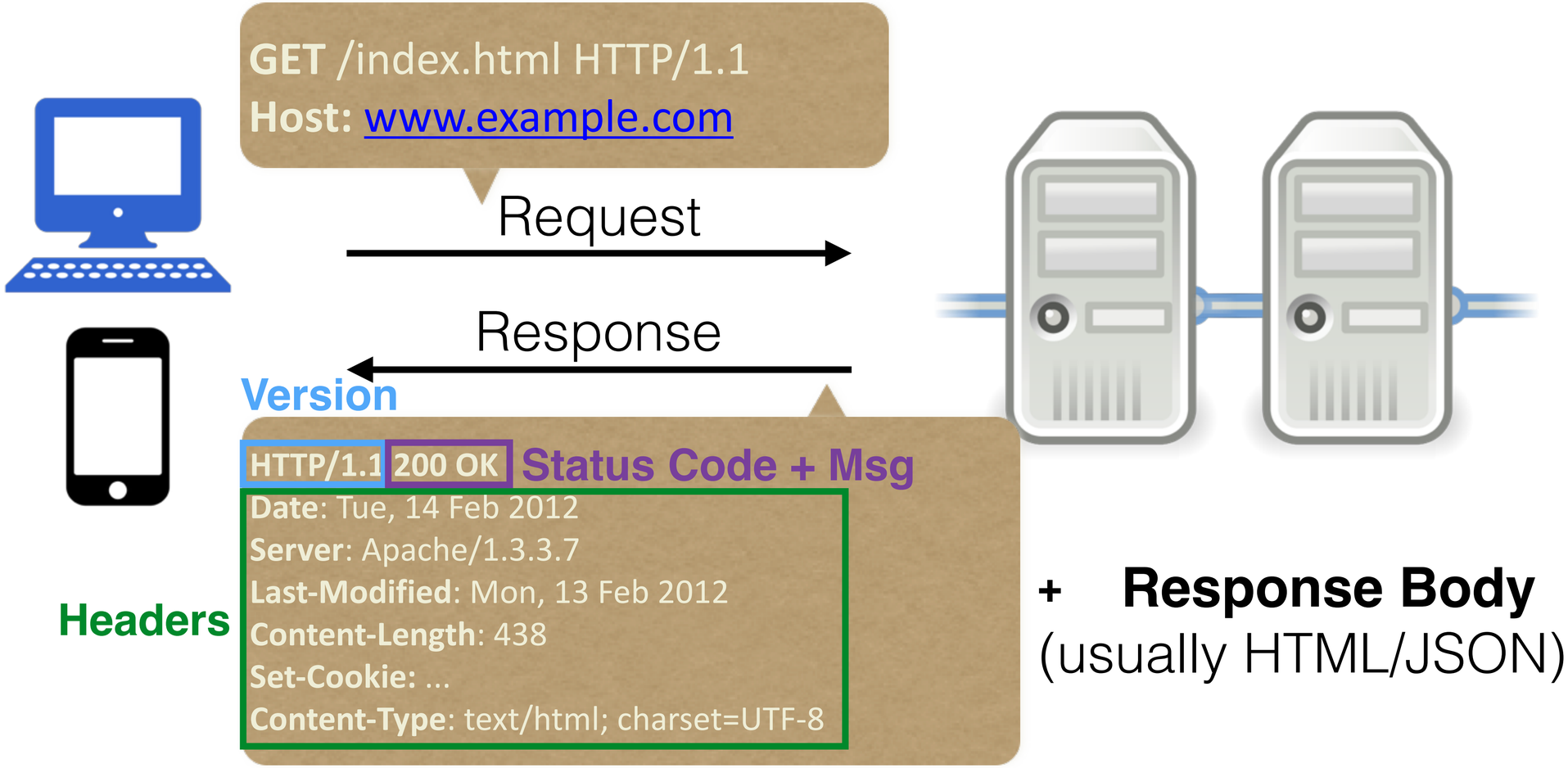
- POST youtube.com/videos - upload a video

DELETE – Delete data resource URL

- DELETE youtube.com/video/32410 - delete a video # 32410

PUT, HEAD, CONNECT, OPTIONS, TRACE, PATCH

HTTP

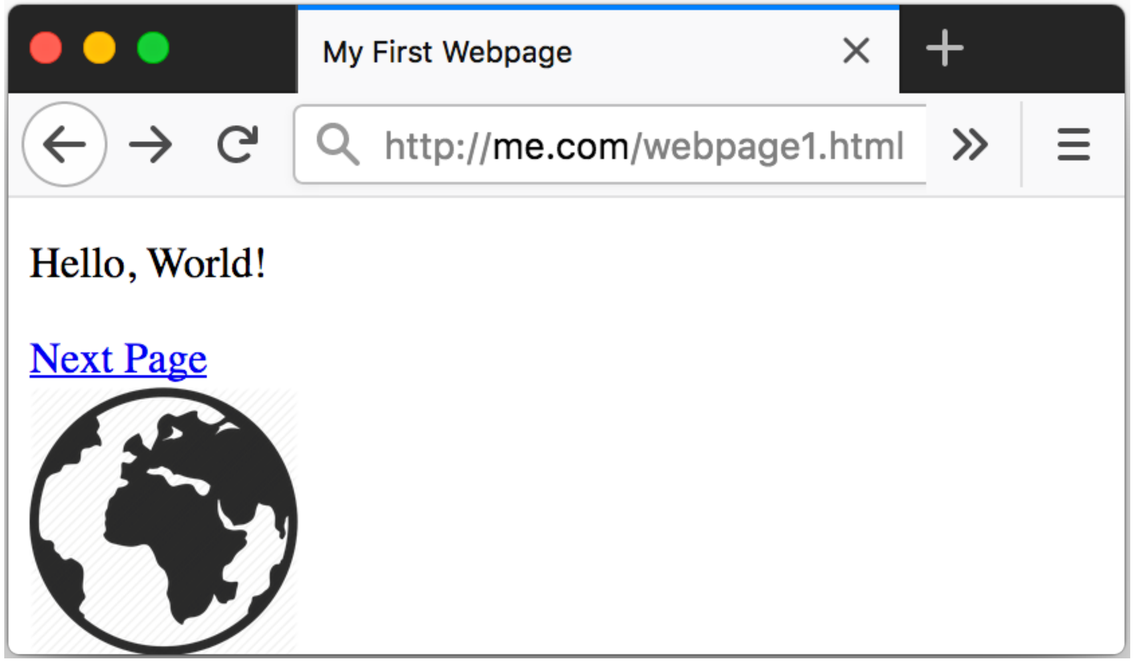

GET /index.html HTTP/1.1
Host: www.example.com
Request
Response
Version
HTTP/1.1
200 OK
Status Code + Msg
Date: Tue, 14 Feb 2012
Server: Apache/1.3.3.7
Last-Modified: Mon, 13 Feb 2012
Content-Length: 438
Set-Cookie: ...
Content-Type: text/html; charset=UTF-8
Headers
+ Response Body
(usually HTML/JSON)

HTML

Hyper**t**ext **M**arkup **L**anguage

Format for specifying web page layout + dependencies

```html
<!DOCTYPE html>
<html>
    <head>
        <title>My First Webpage</title>
    </head>
    <body>
        <p>Hello, World!</p>
        <a href="/webpage2.html">Next Page</a>
        <br>
        <img src="http://wiki.com/Earth_globe.png">
    </body>
</html>
```

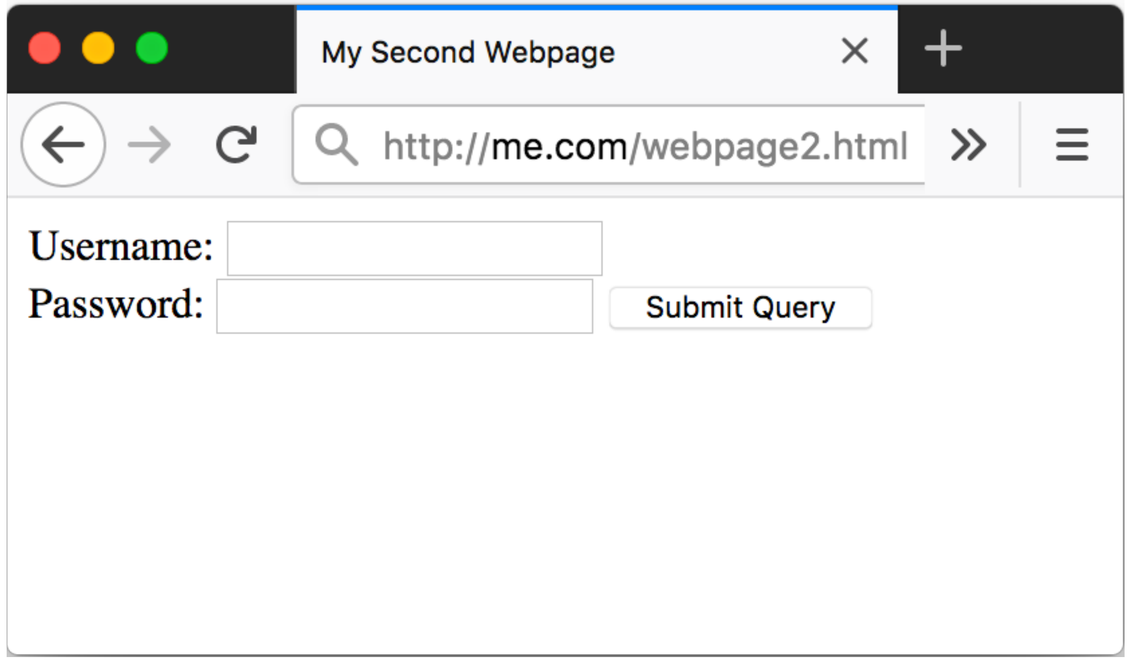


HTML

Not just for retrieval of static content - data submission

User input through <form> and <input> tags

```html
<!DOCTYPE html>
<html>
    <head>
        <title>My Second Webpage</title>
    </head>
    <body>
        <form action="/login" method="POST">
            Username:
            <input type="text" name="username">
            <br>
            Password:
            <input type="password" name="password">
            <input type="submit" name="submit">
        </form>
    </body>
</html>
```

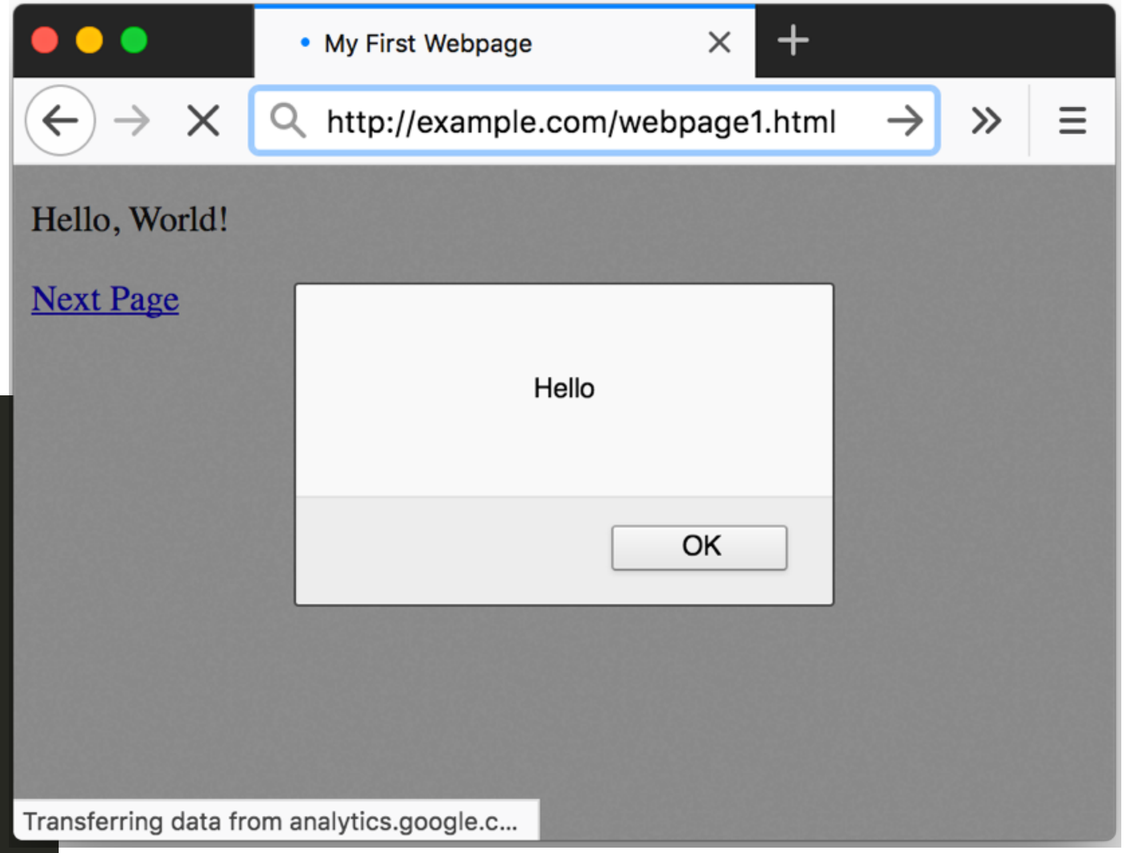


Dynamic HTML

JavaScript = Turing-complete language that allows for dynamic webpages

Flash, Java also used

```html
<!DOCTYPE html>
<html>
    <head>
        <title>My First Webpage</title>
    </head>
    <body>
        <p>Hello, World!</p>
        <a href="/webpage2.html">Next Page</a>
    </body>
    <script type="text/javascript">
            alert("Hello");
    </script>
    <script type="text/javascript" src="http://analytics.google.com/">
    </script>
</html>
```



JavaScript

Powerful browser programming language that can:

- Alter page contents
- Track events (mouse click, motion, keystrokes)
- Access hardware (camera, microphone, location, filesystem)
- Read / set cookies
- Issue web requests

Despite its name, not related to Java!

JavaScript

Code enclosed within <script> … </script> tags

Event handlers can be embedded in HTML

```
<img src="picture.gif" onMouseOver="alert('Leave the picture alone!')">
```

Built-in functions can change content of window

```
window.open("http://illinois.edu");
```

Click-jacking attack

```
<a onMouseUp="window.open('http://www.evilsite.com')"
  href="http://www.trustedsite.com/">Trust me!</a>
```

jQuery

Popular library that simplifies most aspects of JavaScript

```
<input id="button1" type="button" value="Click Me!"/>
```

Click Me!

JavaScript

```
var button = document.getElementById("button1");
button.addEventListener('click', function() {
    alert("Hello");
});
```

jQuery

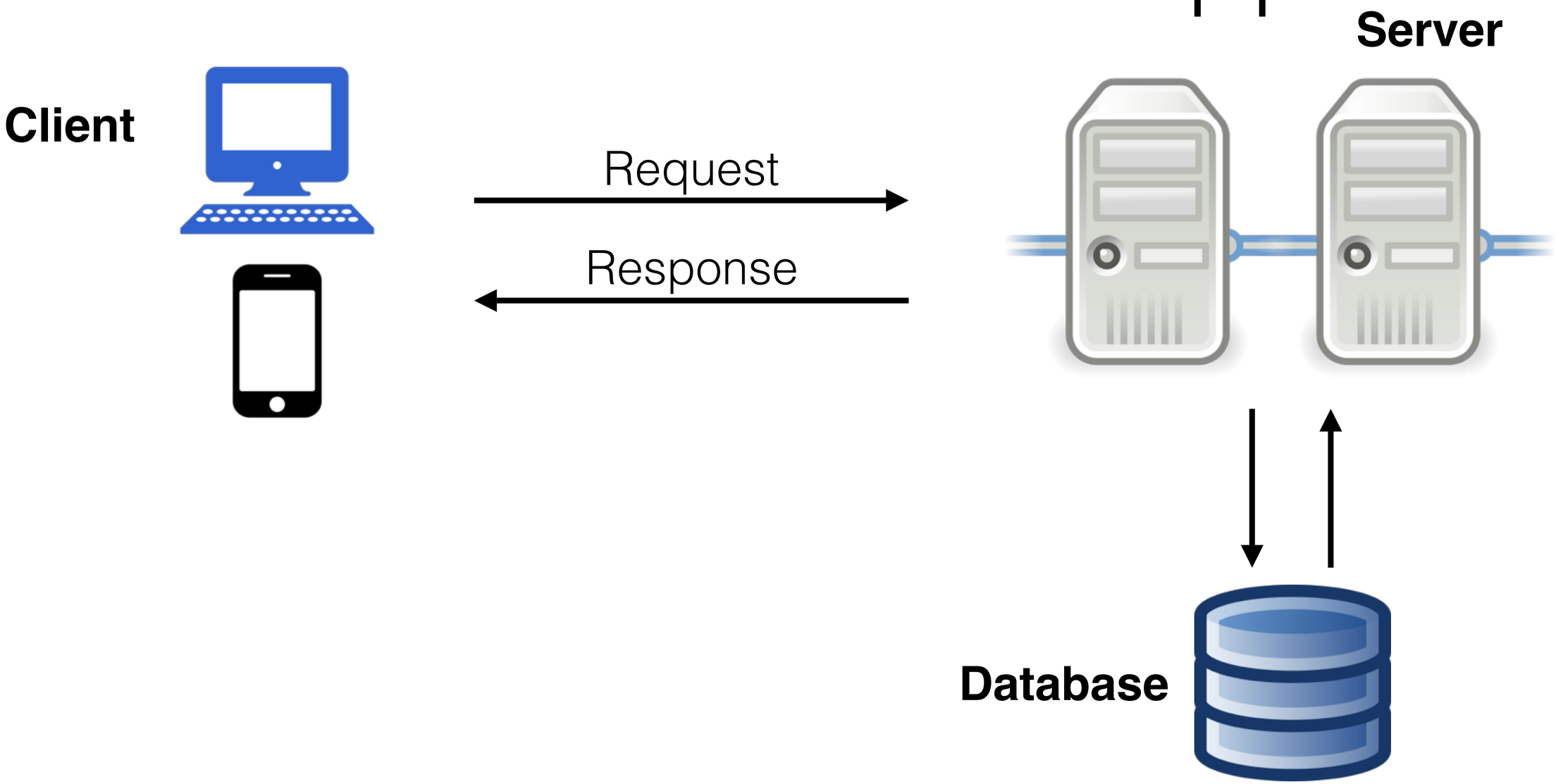
```
$('#button1').click(function(){
    alert("Hello");
});
```

$() == jquery()

prefixes: # for id, . for class

jQuery also handles browser discrepancies!

Three-Tiered Web App



Server

Servers respond to HTTP requests with HTML, JS, CSS, etc. files

- Static: JS, CSS, images
- Dynamic: HTML - e.g. Facebook feed, online bank profile

Example: PHP: Hypertext Processor - can be done in any language

```
<!DOCTYPE html>
<html>
    <body>
    Welcome! Your IP address is:
    <?php echo $_SERVER['REMOTE_ADDR'] ?>
</body>
</html>
```

PHP

```
<!DOCTYPE html>
<html>
    <body>
    Welcome! Your IP address is:
    130.126.255.93
</body>
</html>
```

Databases

Servers often need to store data (i.e. when POST request is made)

- Usernames, passwords, PII, etc.

Use a database that can create, read, update and delete records (CRUD)

Structured Query Language (SQL) is used to interact with many popular database systems - many dialects for different implementations

SQL

Used for relational database systems

Data is split into tables - columns are fields, rows are individual records

Users Table

Entry	Name	DOB	SSN	COB
1	John Smith	1/2/13	389765904	USA
2	Jane Smith	4/13/11	657893046	USA
3	Martin Sommer	6/7/80	578899888	Germany
4	Ana Trujillo	6/8/10	585939023	Spain

SQL

Users Table

Entry	Name	DOB	SSN	COB
1	John Smith	1/2/13	389765904	USA
2	Jane Smith	4/13/11	657893046	USA
3	Martin Sommer	6/7/80	578899888	Germany
4	Ana Trujillo	6/8/10	585939023	Spain

SELECT * FROM Users;

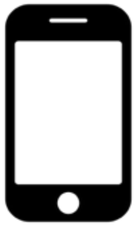
SELECT * FROM Users WHERE COB='USA';

SELECT Name FROM Users WHERE SSN > 600000000;

SQL Keywords are case-insensitive

Three-Tiered Web App

Client



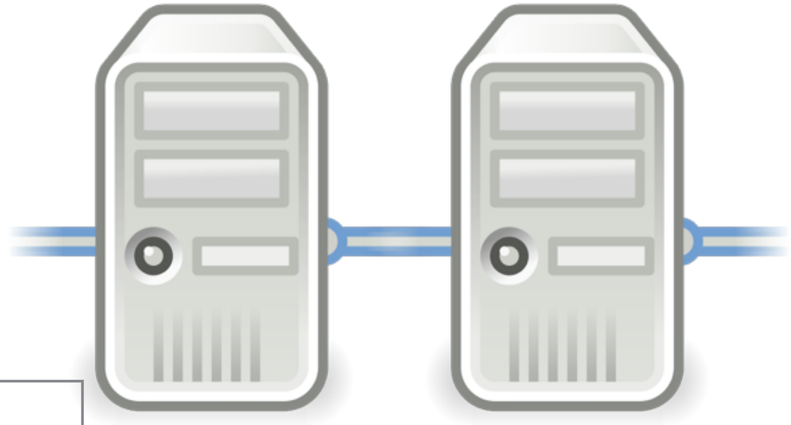
GET /user/1 HTTP/1.1
Host: example.com

Request →

← Response

HTTP/1.1 200 OK
<html><body>Hello, John Smith!</body></html>

Server



SELECT * FROM Users WHERE Entry = 1;

John Smith,
1/2/1913,
389765904,
USA

Database