HubSpot Hack Day

Terminal – can pull up, execute demands without using an interface. Text based commands.

Github is google drive for code.

Github – store code. Teams access it. Accessible on the web.

Git – interact with github (Microsoft word) type code. Use Git to upload those files. keep track of changes and versions.

Client – your machine when you're online. your computer. access a server through the internet

Sends request to SERVER who sends the files to the browser/client

Servers – Local or production. Facebook – production. If you are a facebook engineer then its local. Your own computer is the server. test code changes locally.

Node – a way to run code on the server using JavaScript. Cool!! Unique to Node.

Nodejs = tool to build programming projects

Express= use on top of Node to build a web project, specific for web programing.

Heroku – servers as a service. Waits for a request. Host web project. Have this app online.

Database – store data // user info // posts tweets

APIs – interact with other people's data. Request response thing. Send me this info I will send you back the data you are asking for. (MBTA, weather app, etc)

-----------------------------------------------------------------------------------------------------------------------------

Protocol : http

* Before you speak a sentence: English. Expect the url in this format, so the browser knows how to read what's coming later.

Server name: [www.buzzfeed.com](http://www.buzzfeed.com)

* Maps IP address of server

File path: “nataliebrown/low-stress-pasta-dinner”

Clients make request to servers around the world.

Browsers display the response you get back from the server.

Servers send responses back to the clients. Parse request, read it, send back the html file that the client requests

Html file

Header <h1> that’s a tag

<p> paragraph. Browser knows how to deal with it. Just google it.

There's different forms like h2 links etc. built in.

HTML JavaScript CSS = what gets you that page

There's **HTML** on every website. Bones. (Header, paragraph, how you structure the whole page)

**Javascript** is the muscle. Click a button and something happens. Log in user password do something with that password. does the action. connect something from Javascript to that button and Javascript will execute.

**CSS** – skin. The way things look the design the colors how they look on the page, bold when you hover, etc...

JS it does stuff. Onclick ----- execute this Javascript. myFunction(). At the top there are script tags. Browser knows that Javascript lives in that script tag. Replaces html with new html.

CSS makes stuff look good

Add stuff within style tags.

<style> setting body of the page.

<link rel = "stylesheet" type="text/css" href="style.css"/>

Static HTML page vs Facebook

DATA is the diff.

Create data through APIs and Databases.

APIS get data from databases that aren't your own. Using javascript. Gets info from MBTA and displays it on the page.

Databases. Store it. All the tweets. Javascript can get all the info and display it.

STEPS:

Make changes. SAVE them!

git diff : shows all the changes

git add .

git commit –m “changes made”

heroku local web

git push heroku master

git push origin master

Command c (quits out of it) or is it control