# HTML Notes

* <article> defines the content as an article
* <aside> moves content to the side that isn’t essential for the article
* <audio> plays sound on the page
* <bdi> bi-directional isolation changes direction of text
* <canvas> draws shapes
* <figcaption> caption for a picture
* <figure> separates images
* <footer> bottom of the page
* <header> top of the page
* <main> most important part of the tag
* <mark> highlight
* <meter> shows progress
* <nav> navigation
* <output> Calculator
* <picture> photographic content
* <progress> Loading bar
* <section> separates this section of the
* <source>

# CSS Notes

* Stylesheet
* Style tag
* Body tag: style attribute
* Selector{ properties: value;}
* Class name
* Html tag
* ID = #
* class^=”” ~ Class that begins with “”
* Clear floats
* :last-child
* :root
* :visited
* :read-only
* :focus
* :valid
* :invalid
* class$= ~ ends with
* <meta http-quiv=”X-UA-Compatible” content= “ie=edge”>
* <meta name= “viewport”content=”width=device-with, initial scale=1.0”>
* @media only screen and (max-width: 1024px){

Body {}

}

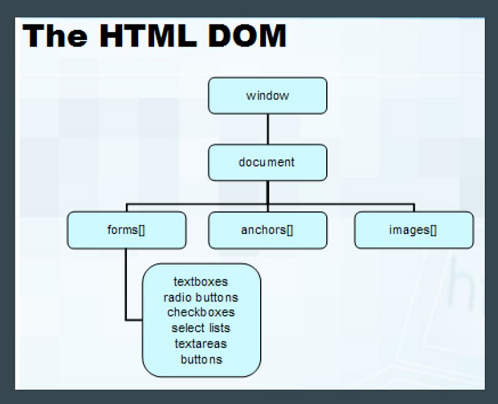
* @media only and (max-with: 500px){

body {}

}

* Largest to smallest media query
* Ctrl+/ = commented out
* HTML: Hyper Text Markup Language
* <th> ~ Table Header
* <td > ~ Table Data
* <tr> ~Table Row
* <form action=“” method= “”>
* method= “get” ~ Puts info in browser bar
* method= “post” ~ Puts info to server
* All inputs need labels
* <select name= “” id= “”> ~list of options
* <option value= “”>
* Selected ~ puts as placeholder
* <input type="checkbox">
* <input type="radio" name="" id="">
* Checkboxes and radios need value
* <fieldset>
* <legend>
* <input type="color" name="favcolor">
* <input type="date" name="" id="">
* <input type="datetime-local" name="" id="">
* <input type="number" name="" id="" min="" step="" max= “”>

# JavaScript Notes

* JavaScript
  + LiveScript
  + JScript
  + ECMAScript
* Eich created Netscape in 1995
* LiveScript → JavaScript
* command (“######”);
* Typeof
* Different datatypes push things together
* 64 bits
* ` ${ }! ` = Template literal
* function doSomething() {}
* Variables in functions are arguments
* function do(x, y) {}
  + do(“”, “”)
* if (this == that) { doThis}
* else if (this == thatThing ) { doThis}
* else { doThis}
* ‘>’ ~ greater than
* ‘<’ ~ less than
* ‘<=’ ~ less than or equal to
* ‘>=’ ~ greater than or equal to
* ‘==’ ~ compares value
* ‘===’ ~ compares values and data type
* ‘&&’ ~ both sides must be true
* ‘||’ ~ only one side needs to be true
* ‘:’ ~else if
* switch (expression) {
  + case n:
    - break;
  + case m:
    - break;
  + default:
    - Break;
* Returns have a value and breaks don’t
* Document Object Model
  + DOM changes the way things are displayed
* DOM can be used to give hover interactivity to mobile devices.
* DOM can help me have a hamburger drop-down menu
* Web apps are powered by JavaScript
* window.innerWidth
* window.innerHeight
* Nodes ~ HTML elements on the page that can be changed
* document.all
* document.contentType
* appendChild ~adds an element
* removeChild
* element.attributes
* removeAttributes
* .style ~add css styles
* .querySelectorAll(‘’)
* .getElementsByTagName(‘’)
* .getElementById(‘’)
* .getElementsByClassName(‘’)
* .className

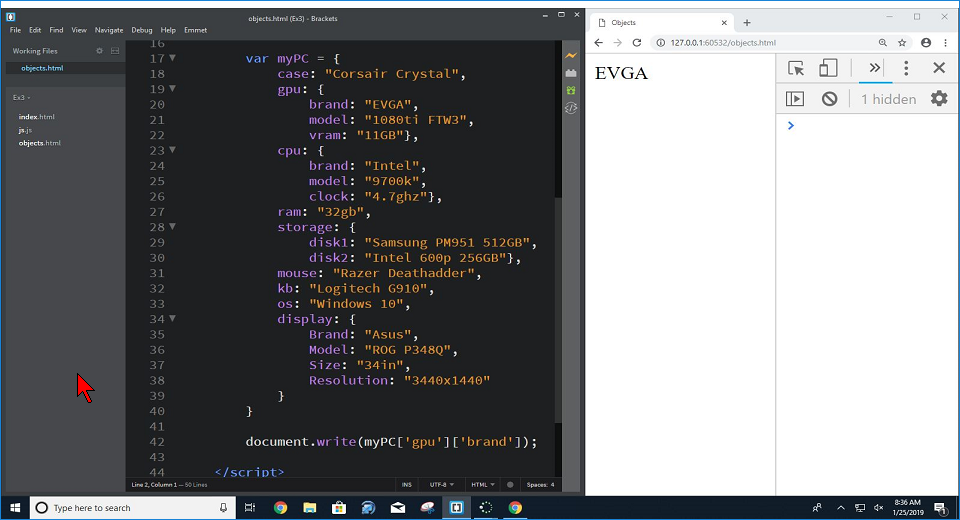
## **FTP Protocol**

* The client talks to the server
* IP Address= 10.2.114.67
* UserName: webappam
* PassWord: l3tm31n

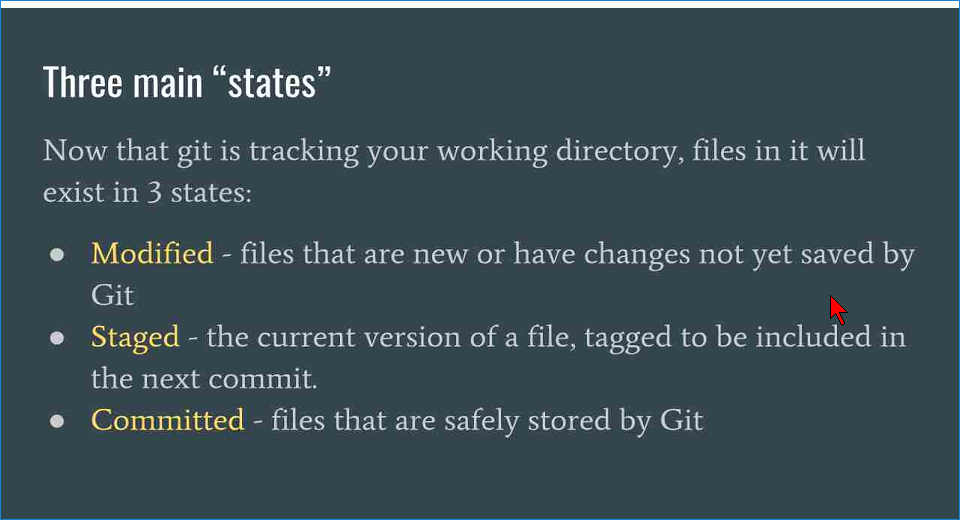
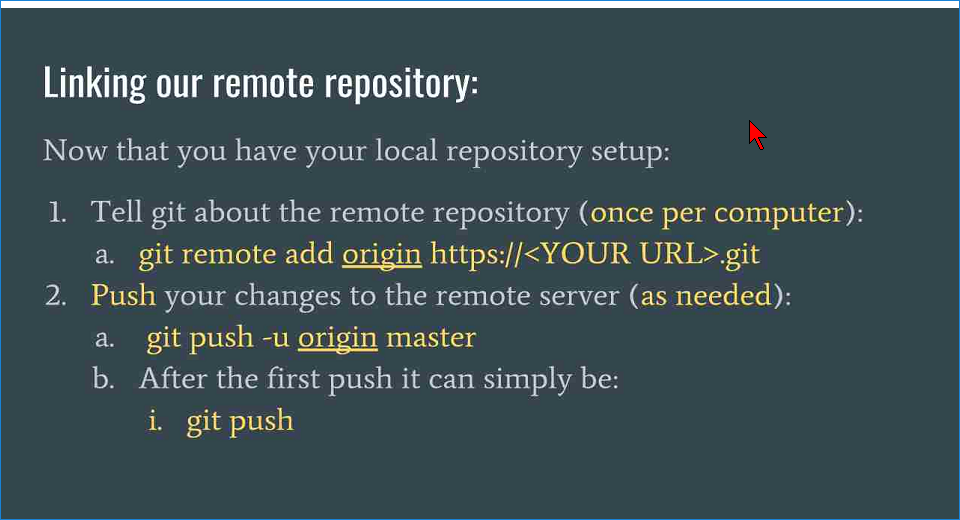
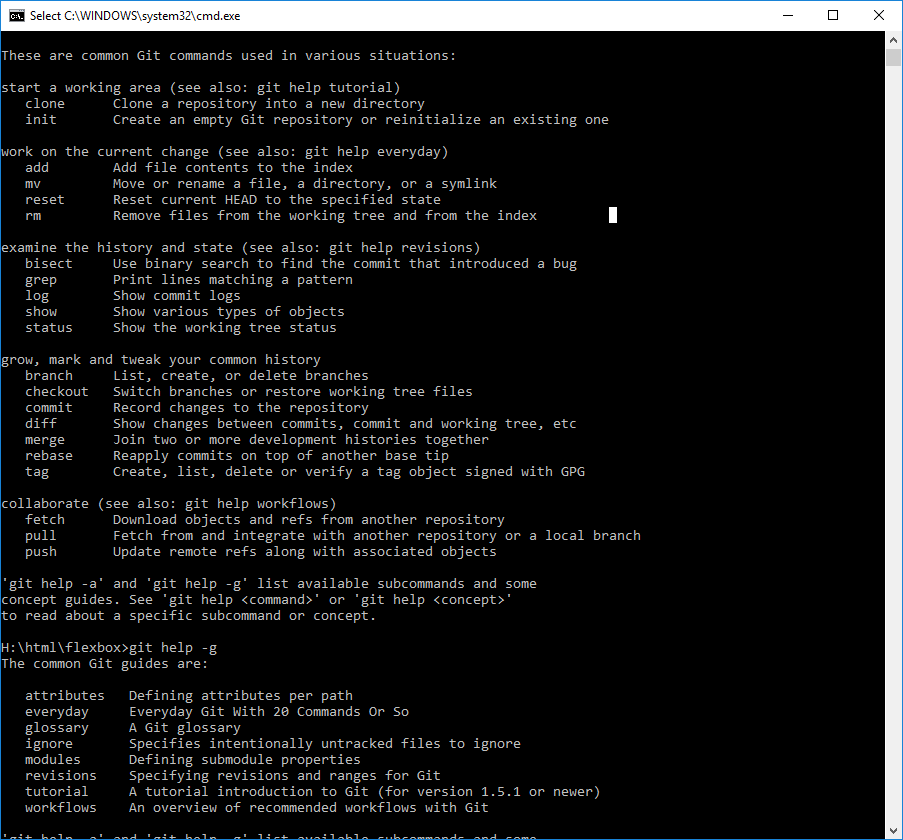
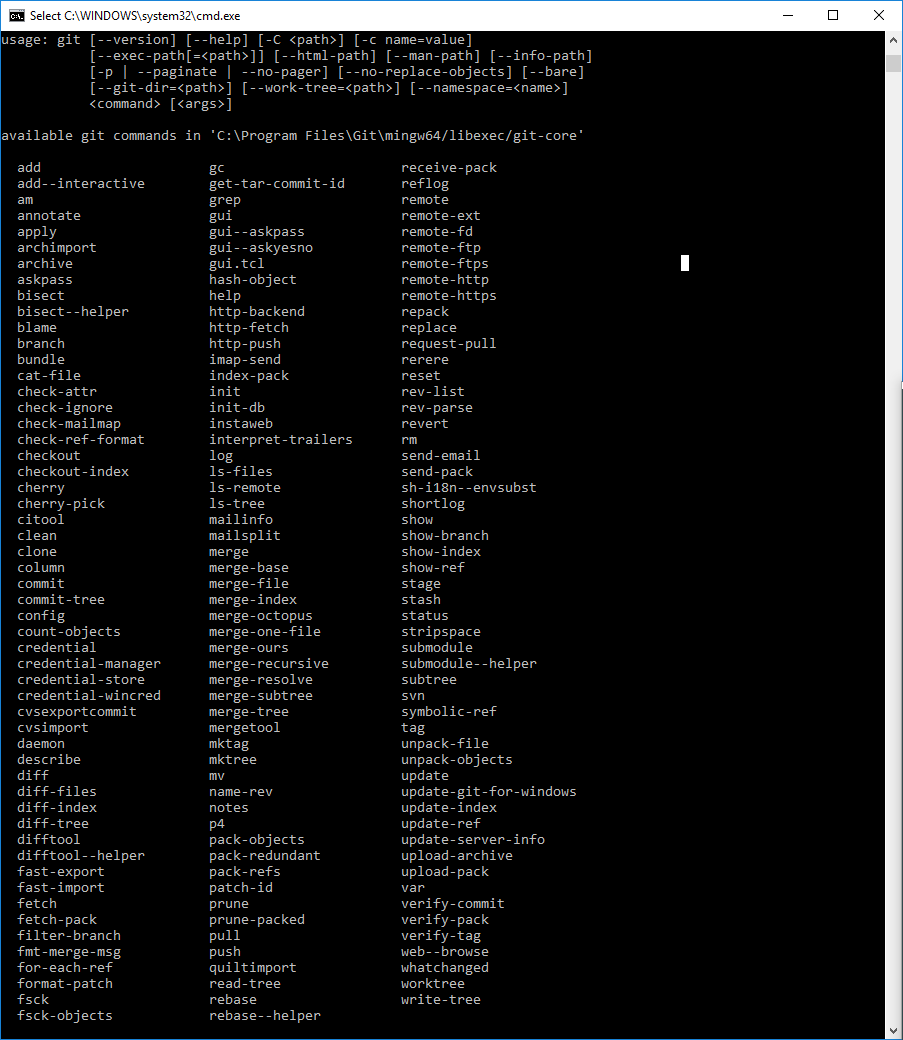
## Arrays

* var ## = [“#”, “#”, “#”, “#”, “#”, “#”];
* Var len = ##.length;
* for(let i = 0; i < len; i++) {

document.write(##[i] + “ ”)};

* ##[0-∞];
* 

## Git

* Coding checkpoints
* 
* 
* Git init
* Git status
* Git add
* Git status
* Git commit -m “”
* Git log
* 
* Git commit -m “”
* Git remote add origin hhtps://
* Git push -u origin .
* Git branch .
* Git checkout .
* Git help
* 
* 
* 4(I got this)
* This was a nice refresher course. We can share repositories on GitHub and collaborate on the same files.