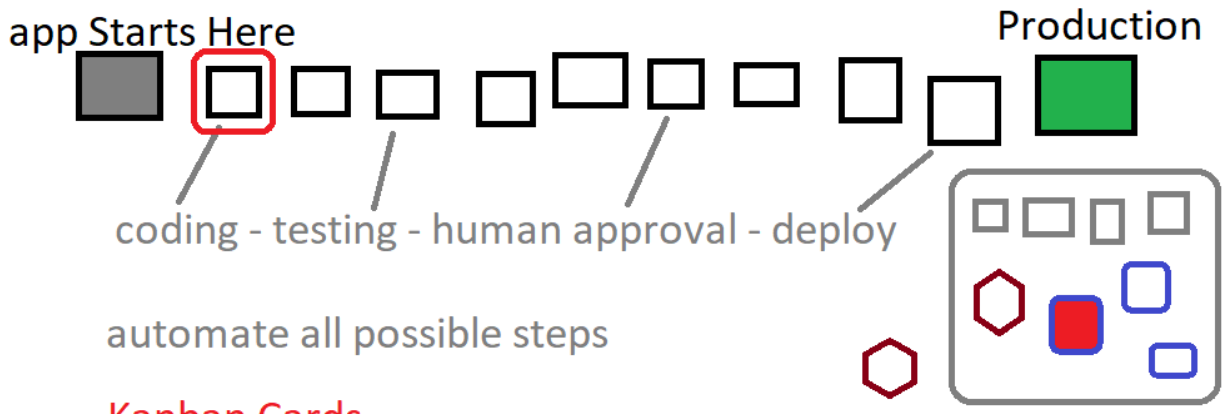
General Application Architect Program: Notes and Discussions about structures, boundaries, code

1. Presentation - UI - HTML - page structure / UX - jQuery - behavior / automate something
2. Validation - Rules the user must follow
3. Logic - how the app works
4. Content: Who cares where it comes from? SQL, Web, Files; Extract – Transform - Load

Azure DevOps - Pipelines

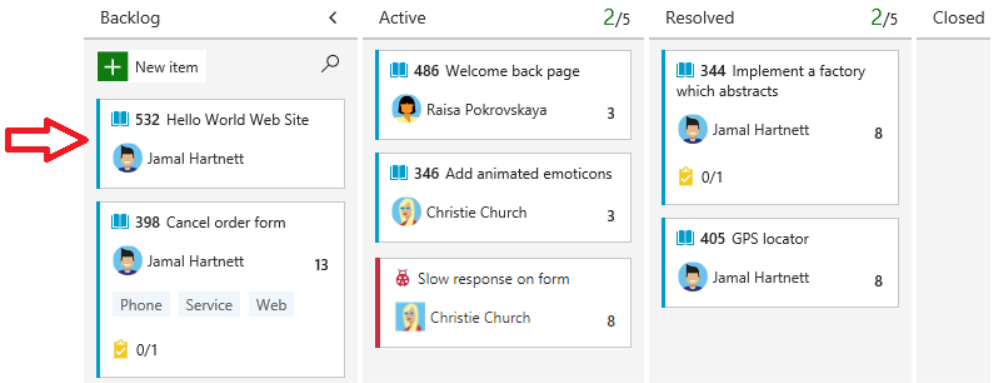
github - central code repo - version control - pipeline

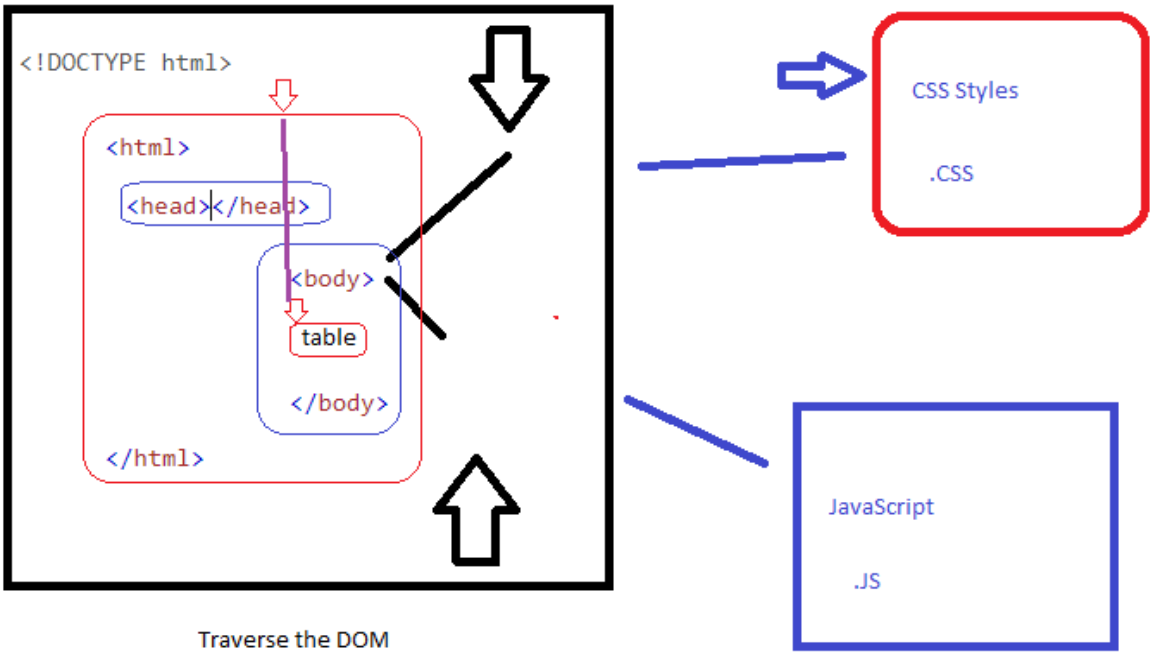
Process - CI/CD - Deployment



automate all possible steps

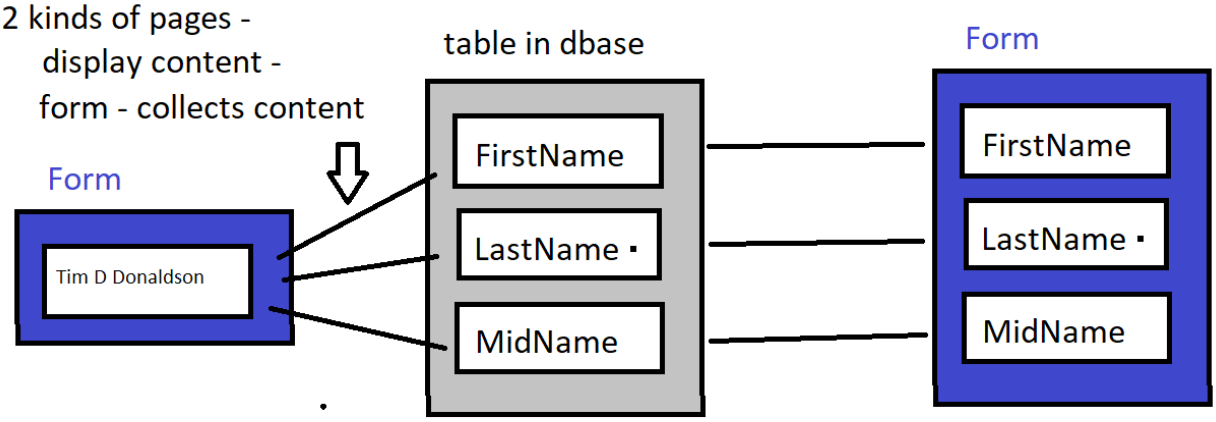
Kanban Cards





Traverse the DOM

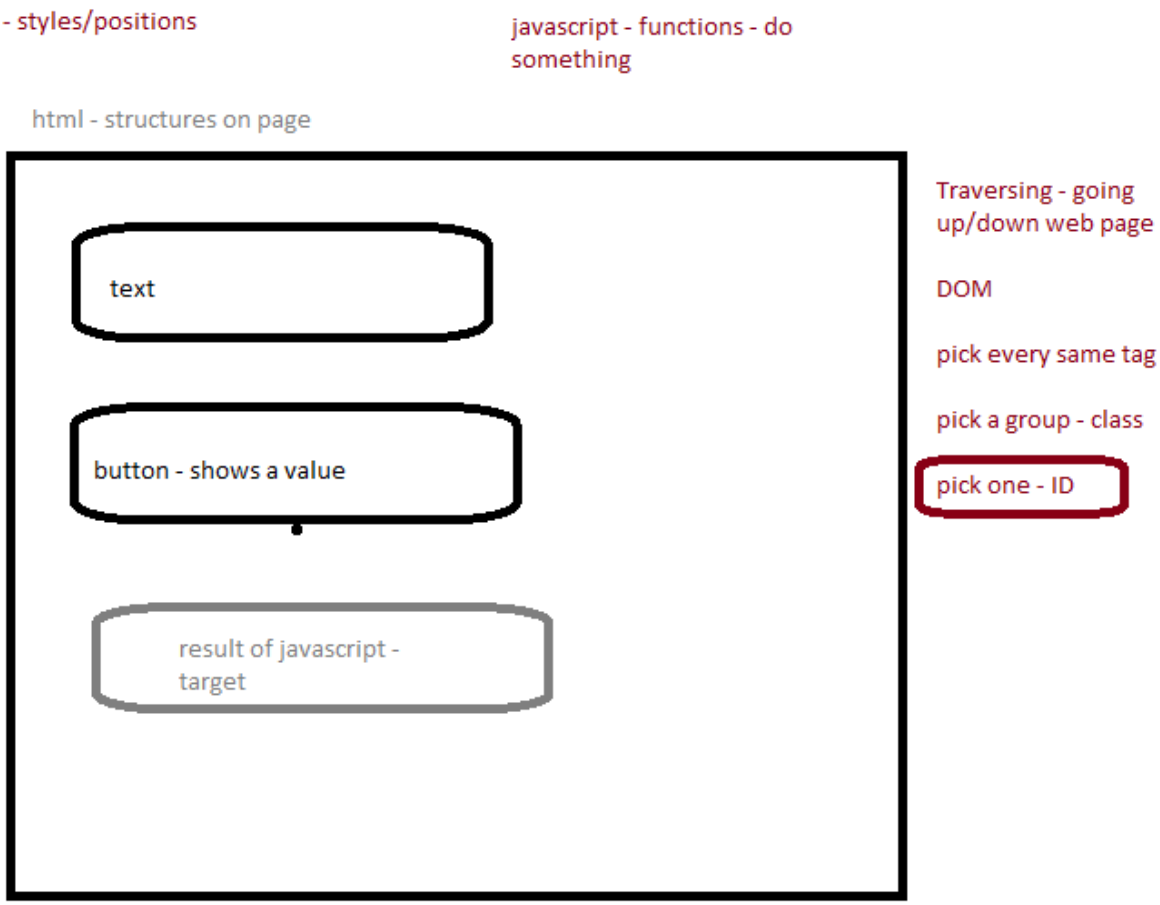
2 kinds of pages -
display content -
form - collects content

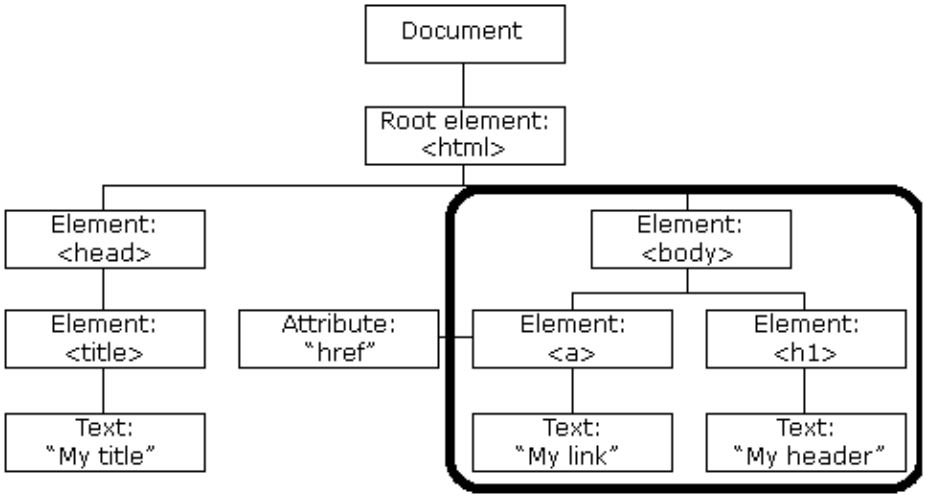


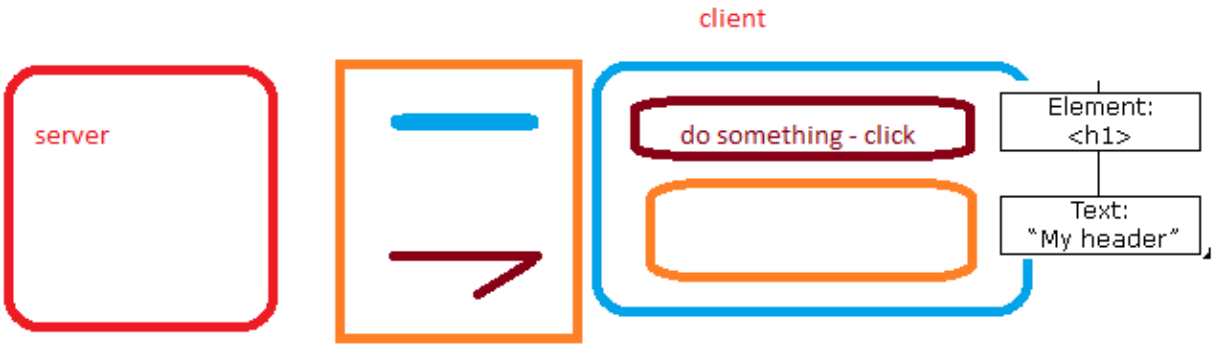
4 index positions - 1

0 1 2 3 index postion

```
<script>
var fruits = ["Banana", "Orange", "Apple", "Mango"];

var last = fruits[fruits.length-1];

document.getElementById("demo").innerHTML = last;
```







Server side - Client side - browser - JavaScript

- JavaScript Object Notation (JSON): data exchange format
- name:value pairs
- Column : Row - table, excel, csv,
- FirstName:Tim

---source that we got content from:

add it to an Array - hold more than a single value - 0 based numbering

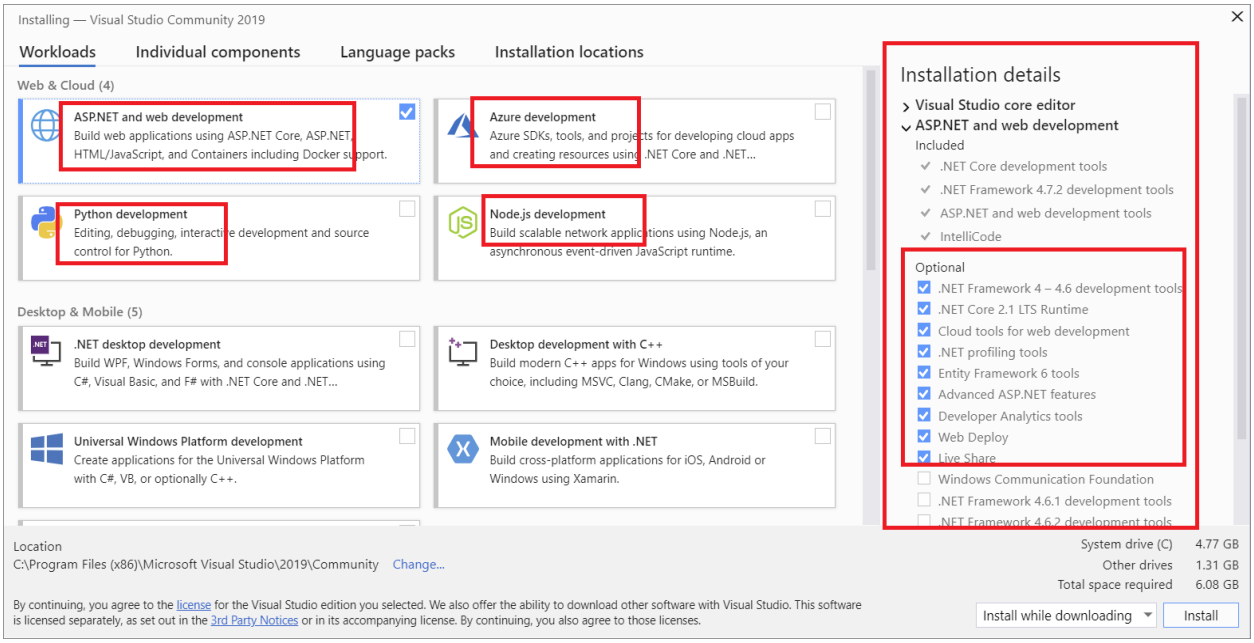
Uses an index

[0]tim

[1]Tim

[2]Josie

Visual studio Installer



List of Languages

SOC: Separation of Concerns - Put everything in separate files & Folders

GOAL: Re using code

1 + 1 = 2

x + y = z using variables

OOP - Object Oriented Programming - Relational Algebra

HTML - structure on web pages

Hypertext Markup Language

DOM - Document Object Model

Object - Properties

Person - FName, LName, Phone

CSS: Cascade Style Sheet

Position, format, style - color, font, etc

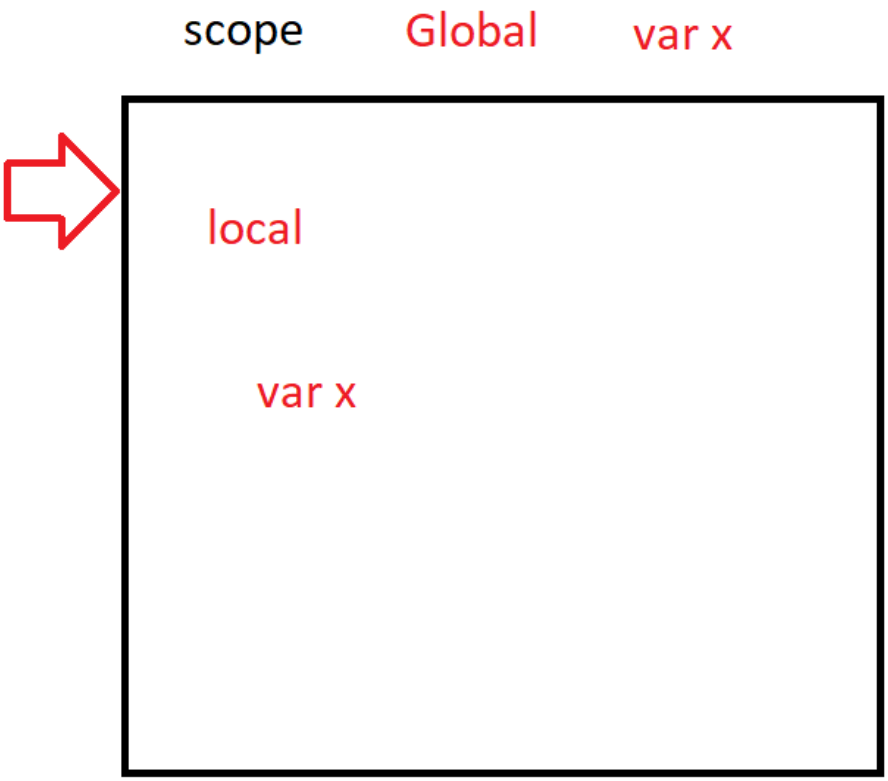
JavaScript: many flavors (JSON, AJAX, JQuery, AngularJS -)

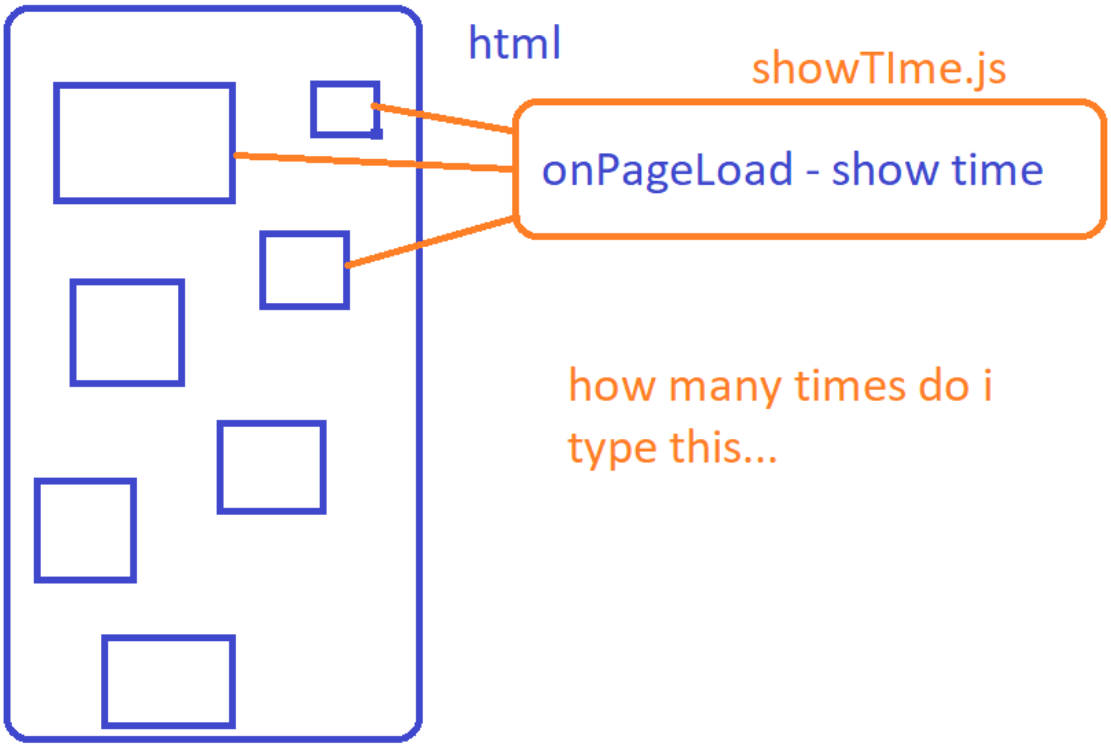
strings - a-z, A-Z, 0-9 - concatenate

numbers - 0-9 - want to do math - must be a numeric data type

implicit - happens automatically

explicit - i told to do that - CAST CONVERT



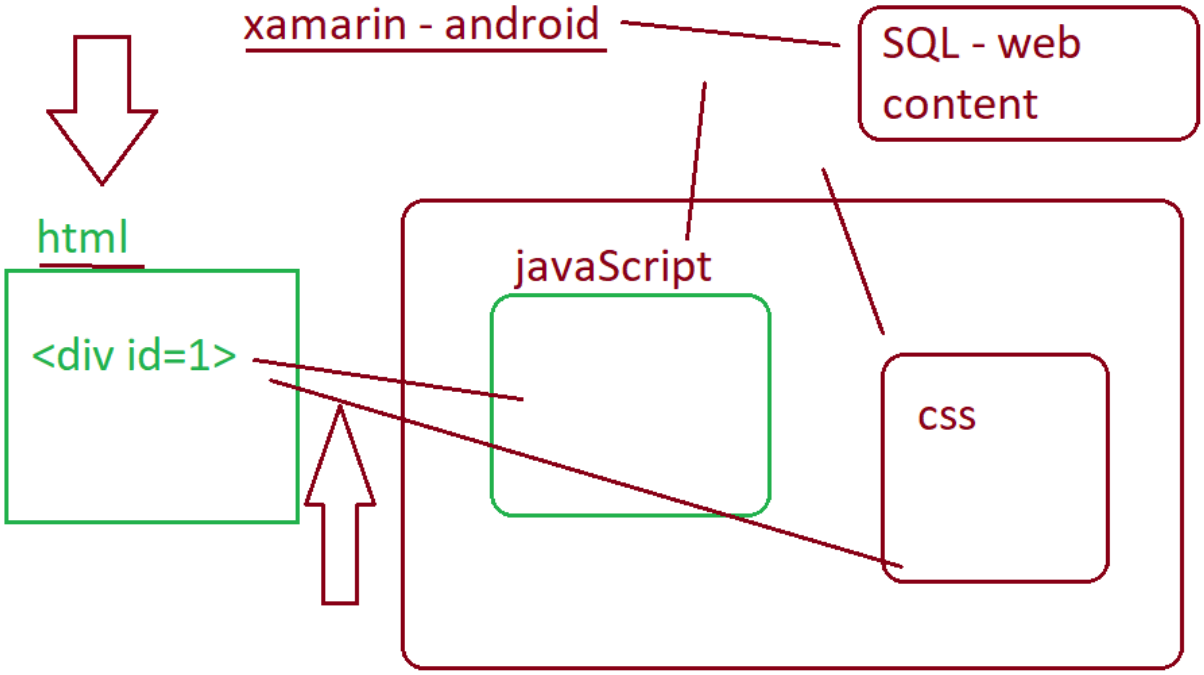


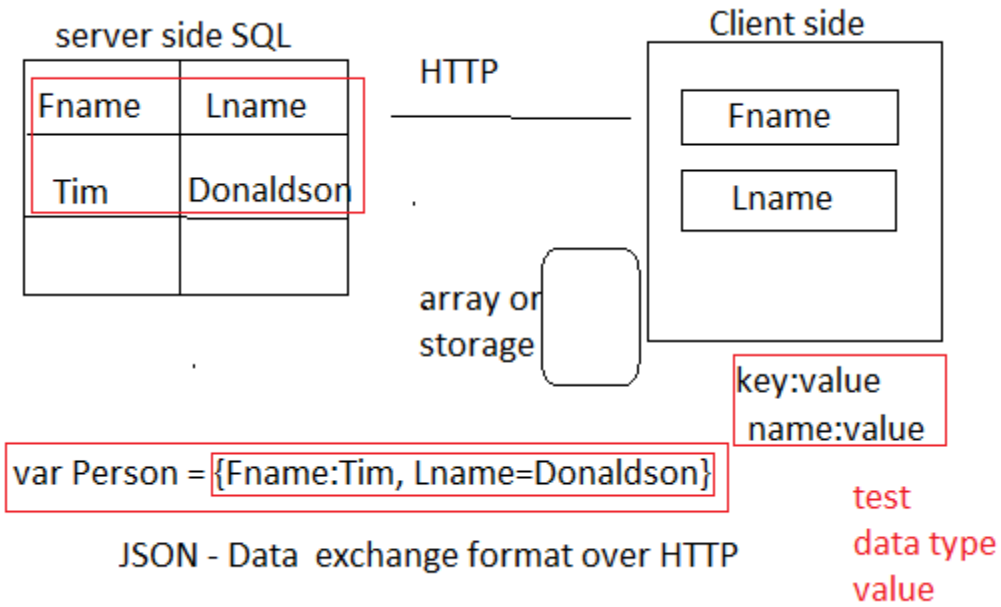
- Introduction
  - Story about helping sister with homework as my introduction to a messed up society
- Chapter 1 – Society Standards
  - Talk about what they shape you to be
  - Why those are issues and how people don't even realize

Chapter's main focus

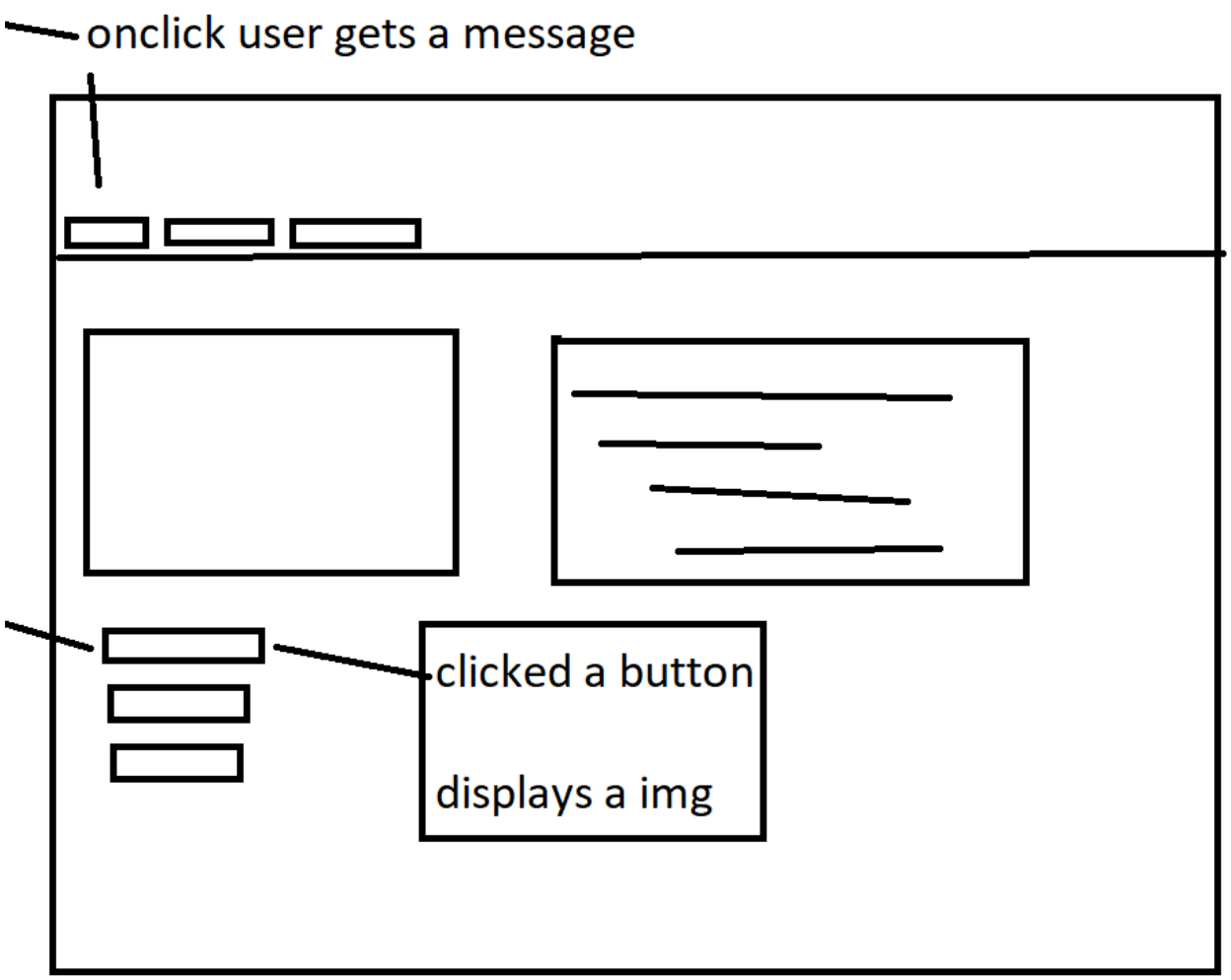
- Chapter 2 – It Starts as a Baby
  - Baby colors
  - Baby boy shirts versus baby girl shirts
  - Toys and costumes
  - College funds started as babies – they're already assuming you'll do what they did – or rather – what they *think* you should do

Beneath the chapter: specific details about how to support the chapter's focus





JSON - Data exchange format over HTTP



More Containers: Delimiters or special characters that create an electronic "box" or end code execution

()

[]

{}

;

""

''

<>

Links:

Event Listener:

https://www.w3schools.com/js/js_htmldom_eventlistener.asp

how to:

https://www.w3schools.com/howto/howto_js_accordion.asp

vs:

jQuery:

https://jqueryui.com/accordion/

Functions calling :

https://www.w3schools.com/js/js_functions.asp

Event Listener:

https://www.w3schools.com/js/tryit.asp?filename=tryjs_addeventlistener_remove

- Button shows alert - with message
- mouseover moves button with event listener
- -shows message: ha ha
- button removes listener
- user can now click the button to show alert.

object access:

https://www.w3schools.com/js/tryit.asp?filename=tryjs_object_accessors_method

Light bulb: turn on several as an array

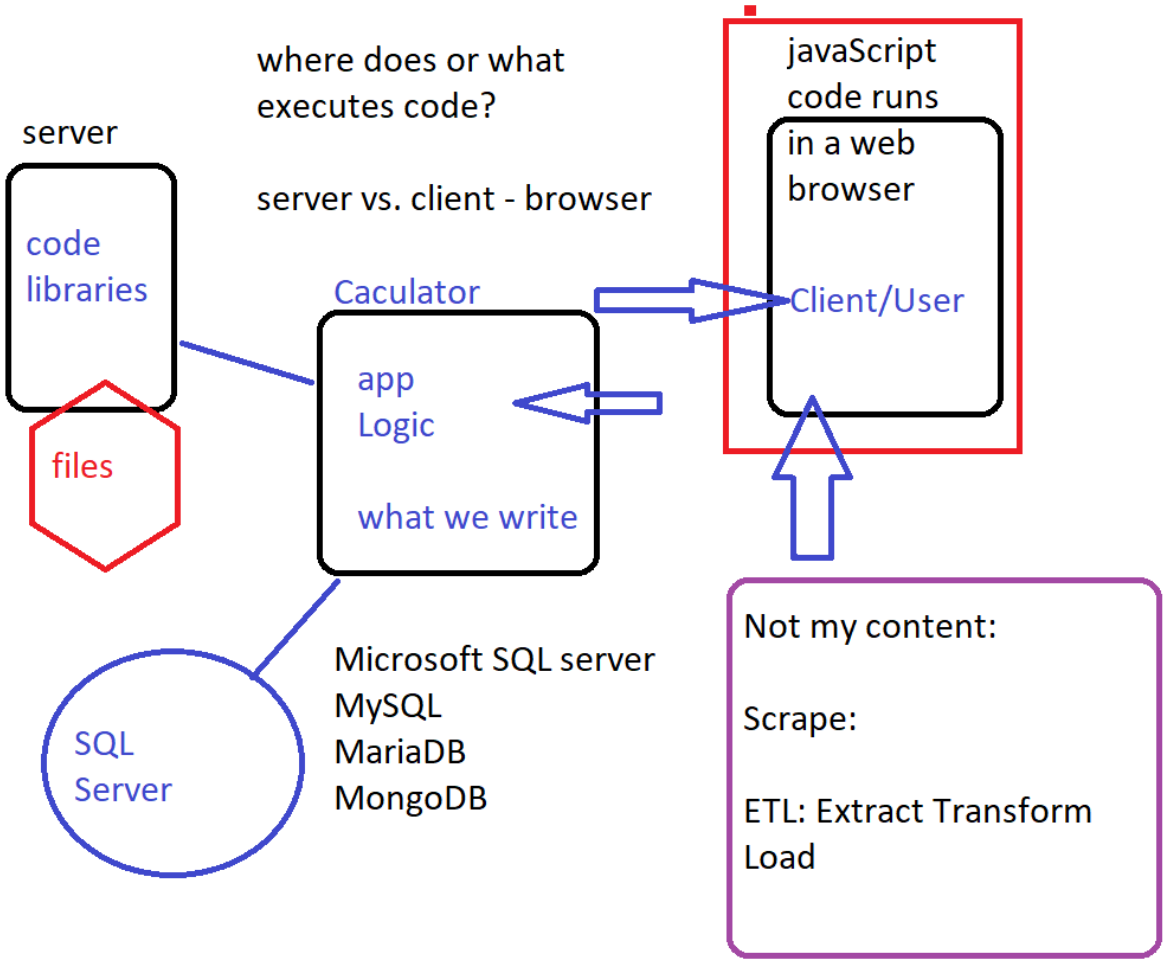
https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_lightbulb

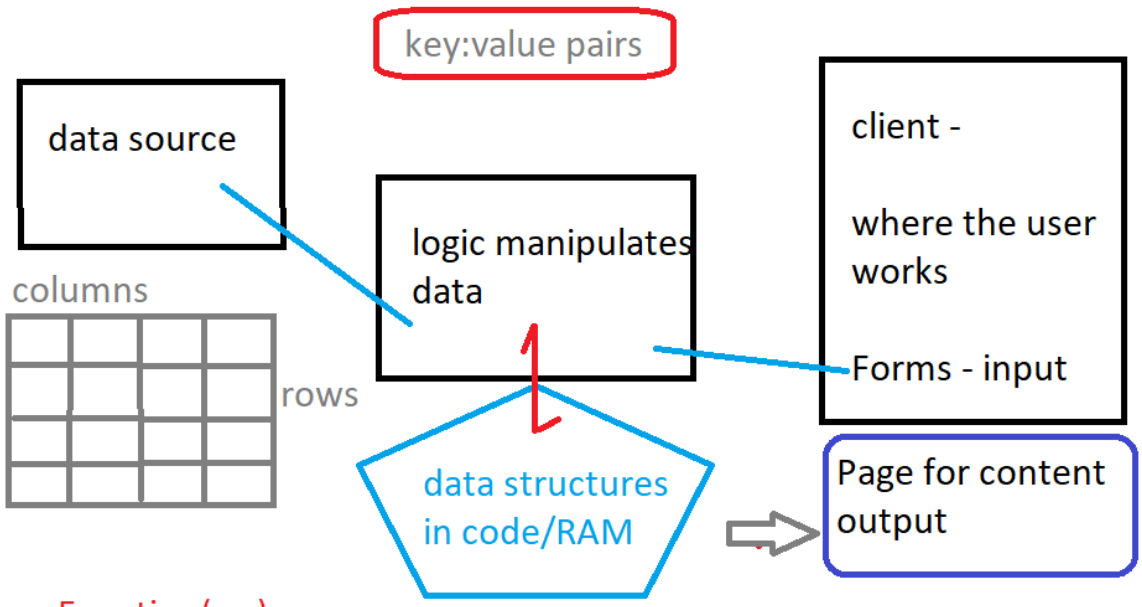
mini apps - CRUD operations

- collect a value from user - prompt box - input html form
- adding to a data structure -
- showing content back to user

end of a list

Python stuff

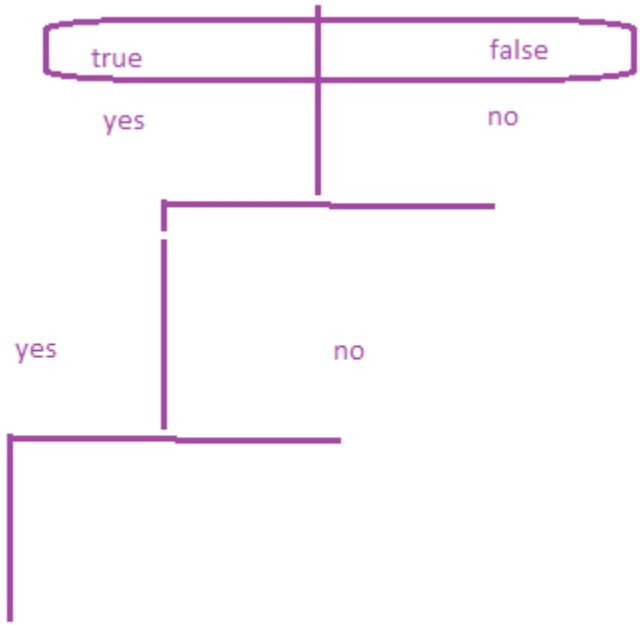




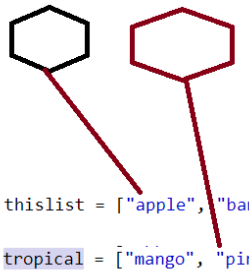
myFunction(x,y)

+ and -
1 and 0 binary code

compiling - or interpret



SOURCE 1 AND 2



```
1 thislist = ["apple", "banana", "cherry"]
2 tropical = ["mango", "pineapple", "papaya"]
3 thislist.extend(tropical)
4 print(thislist)
```

CLIENT OR USER SIDE

```
['apple', 'banana', 'cherry', 'mango', 'pineapple', 'papaya']
```

Can only do this if (?)

missing is user input

```
1 thislist = ["apple", "banana", "cherry"]   #create list
2 thislist.remove("cherry")   #delete from list
3 print(thislist)   #output to user
```

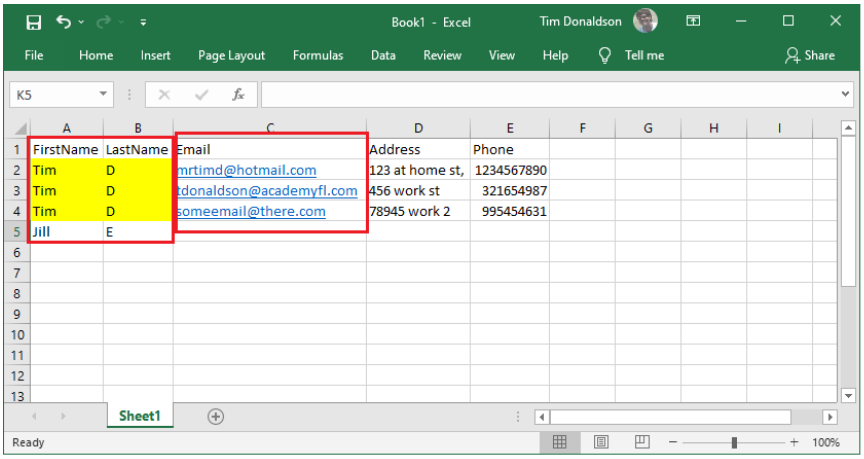
```
['apple', 'banana']
```

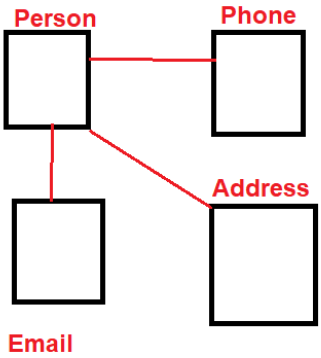
```
1 def myfunc(n):
2   return abs(n - 50)
3 thislist = []   #how do I populate this list?
4 thislist.sort(key = myfunc)
5 print(thislist)
```

`list2 = list1`,

because: `list2` will only be a *reference* to `list1`,

, and changes made in `list1` will automatically also be made in `list2`.





In real life, a car is an **object**.

A car has **properties** like weight and color, and **methods** like start and stop:

Object	Properties	Methods
	car.name = Fiat	car.start()
	car.model = 500	car.drive()
	car.weight = 850kg	car.brake()
	car.color = white	car.stop()

All cars have the same **properties**, but the property **values** differ from car to car.

All cars have the same **methods**, but the methods are performed **at different times**.

Other notes/vocabulary

Data Structures:

- Is it One value - Scalar value
- Is it a complex value - array, obj

---does it represent columns and rows somehow?

(Objects with properties)

(elements with attributes)

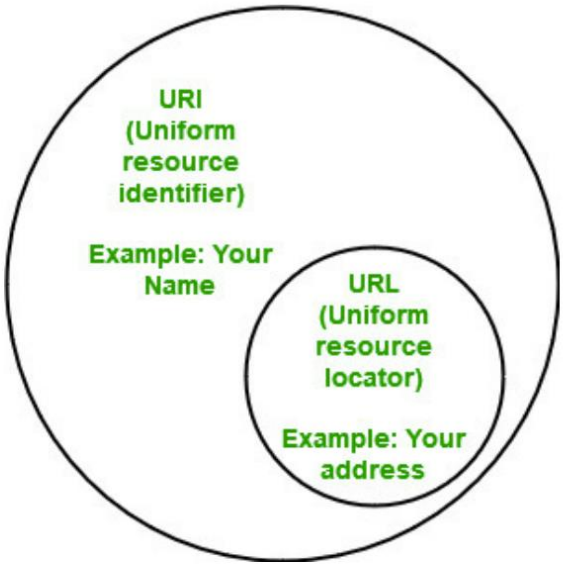
(name:value) pairs JSON

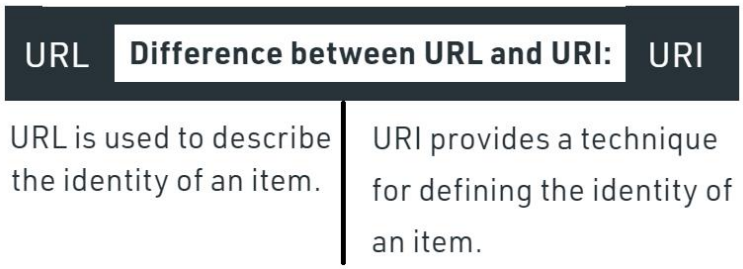
Data Types – Do math or write a sentence?

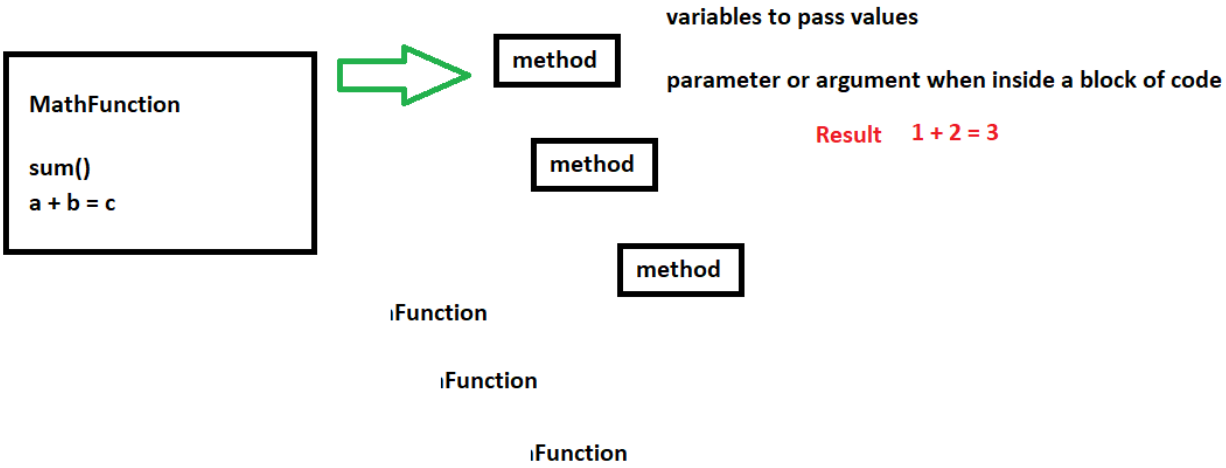
- Number - int - float - decimal - money
- String - text - nvarchar(10) - char

Images

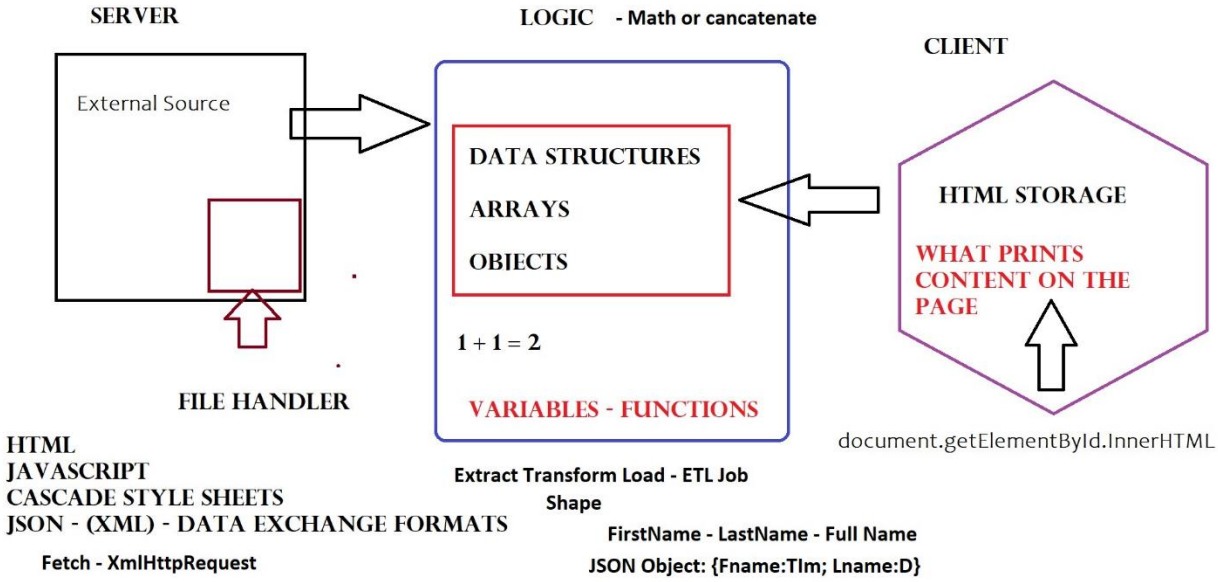
URL vs URI: [Difference between URL and URI - GeeksforGeeks]







Common Parts



- Map Source to Client as a JSON name:value pair
- Column/Row map: [Fname:Lname]
- Source: Database, file, web, other?
- JSON: name:value pair
- JavaScript Structure: [array], {object}, {JSON}
- HTML Form: GET & Querystring
- HTML Storage – File content (Fetch)

Filtering - shaping - transform -

changing data types or making a set smaller

filter()
SQL - WHERE

RULES

Take a large bucket of data and find groups or unique values

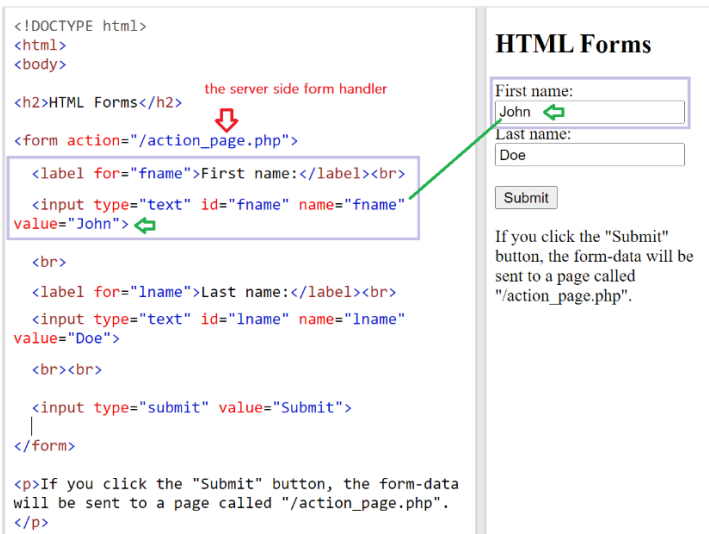
	Id	Fname	Lname
1	1	John	Doe
2			
3			

JSON Example

```
{"employees":[
  { "firstName":"John", "lastName":"Doe" },
  { "firstName":"Anna", "lastName":"Smith" },
  { "firstName":"Peter", "lastName":"Jones" }
]}
```

To a Variable

How will form data populate JS – Or send JS to page InnerHTML

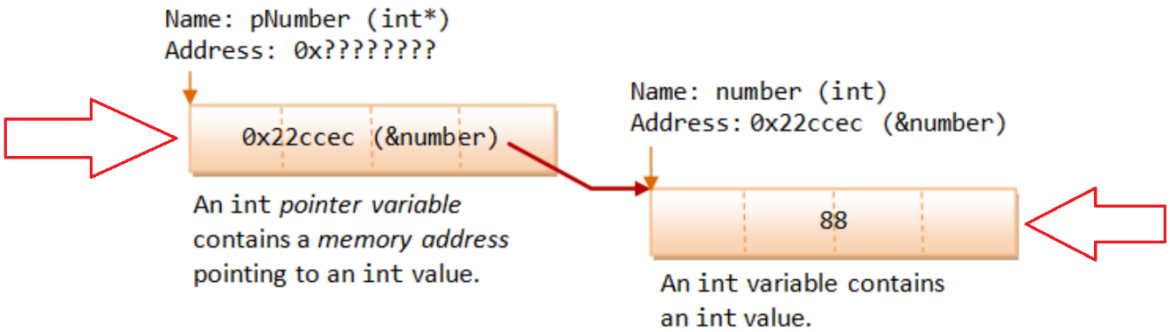



```
<!DOCTYPE html>
<html>
<body>

<h2>HTML Forms</h2>

<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John">
  <br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe">
  <br><br>
  <input type="submit" value="Submit">
</form>

<p>If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".</p>
```

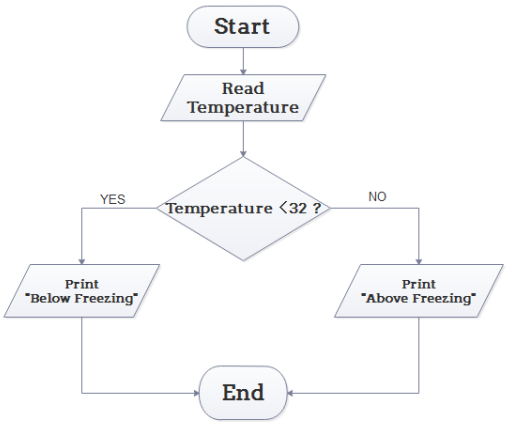
Pointer

You can use the address-of operator to get the address of a variable, and assign the address to a pointer variable. For example,

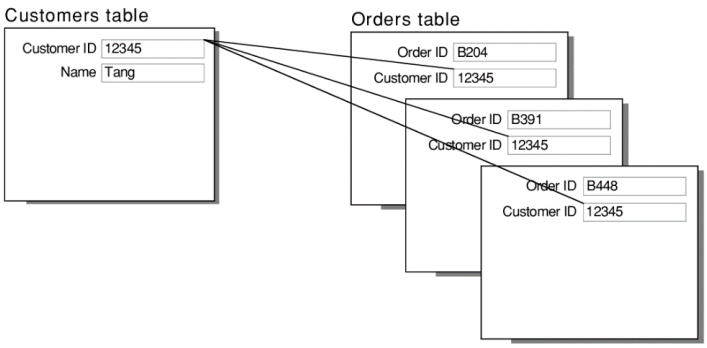
```
int number = 88;      // An int variable with a value
int * pNumber;        // Declare a pointer variable called pNumber pointing to an int (or int pointer)
pNumber = &number;    // Assign the address of the variable number to pointer pNumber

int * pAnother = &number; // Declare another int pointer and init to address of the variable number
```



Control Program Flow



More complex data structure representing multiple "tables" with 1 to Many relations



Map Columns to data structure with purpose – hierarchy – etc..

```
<h2>Customer Report</h2>

<p id="demo"></p>

<script>
let x = "";
const customer = {
  name: "John",
  age: 30,
  orders: [
    {name:"Ford", models:["Fiesta", "Focus", "Mustang"]},
    {name:"BMW", models:["320", "X3", "X5"]},
    {name:"Fiat", models:["500", "Panda"]}
  ]
}
//loop the object and output the properties

for (let i in customer.orders) {
  x += "<h2>" + customer.orders[i].name + "</h2>";
  for (let j in customer.orders[i].models) {
    x += customer.orders[i].models[j] + "<br>";
  }
}

//print statement
document.getElementById("demo").innerHTML = x;
</script>
```

customer table or spreadsheet
order table
order details

one customer customer table

3 orders orders tbl

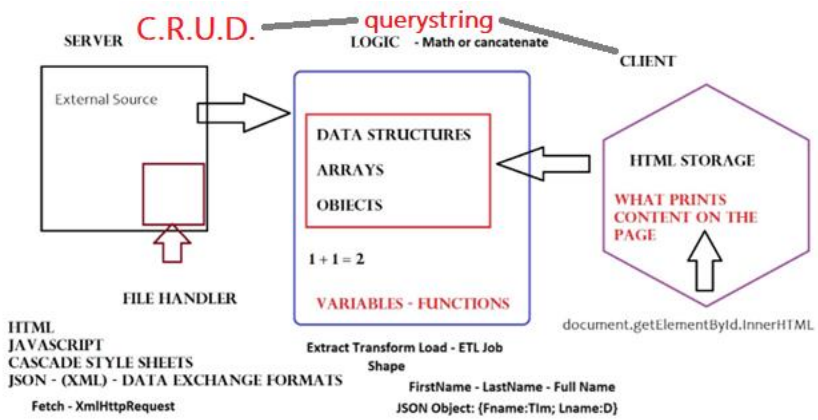
each order has multiple items order detail

customer object

orders object

object detail array

920 × 850px



Using correct HTML header tags to follow the <article> or <section> "outline"

h1

1. Things to Do in San Diego

h2

a. City

h3

i. San Diego Zoo
ii. Petco Park
iii. Balboa Park

HTML

h4

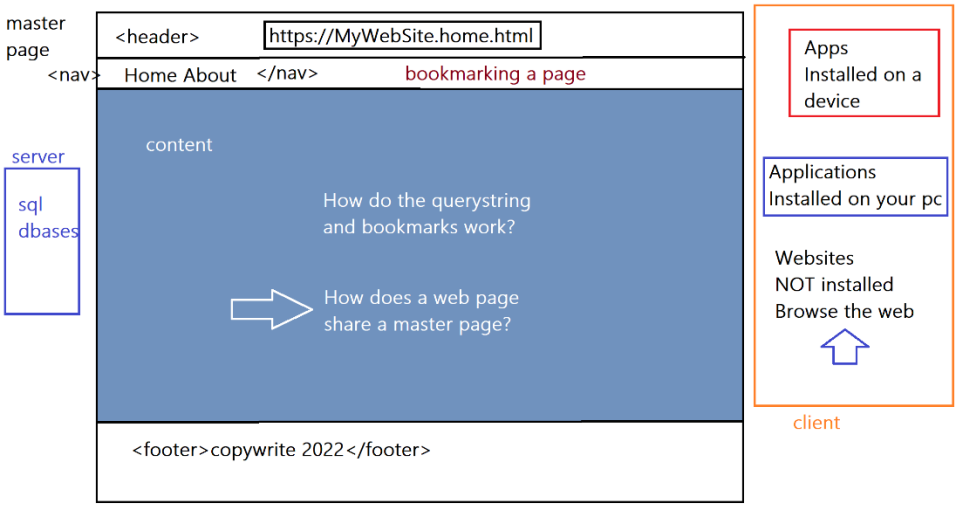
1. Art museum
2. Science museum
3. Sports museum

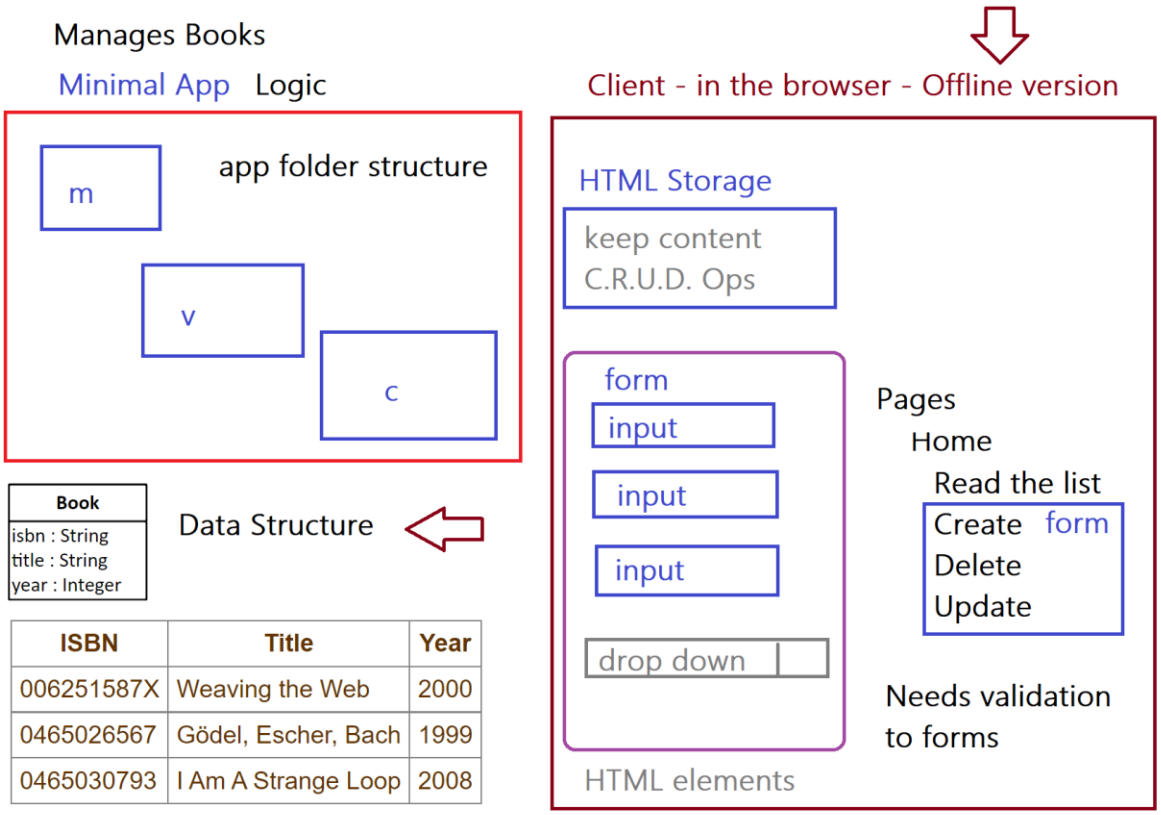
b. Beach

i. Mission Beach
ii. Ocean Beach
iii. Pacific Beach
iv. La Jolla
v. San Elijo State Beach
vi. Carlsbad State Beach

c. North County

i. Legoland

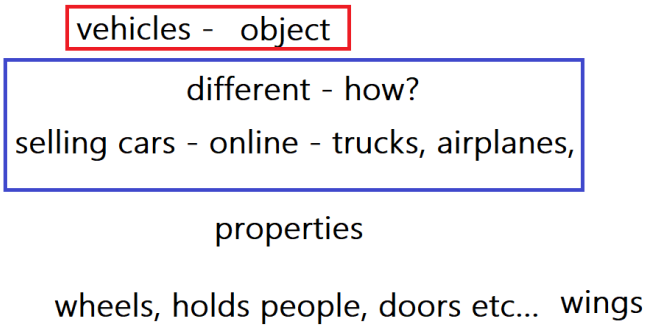


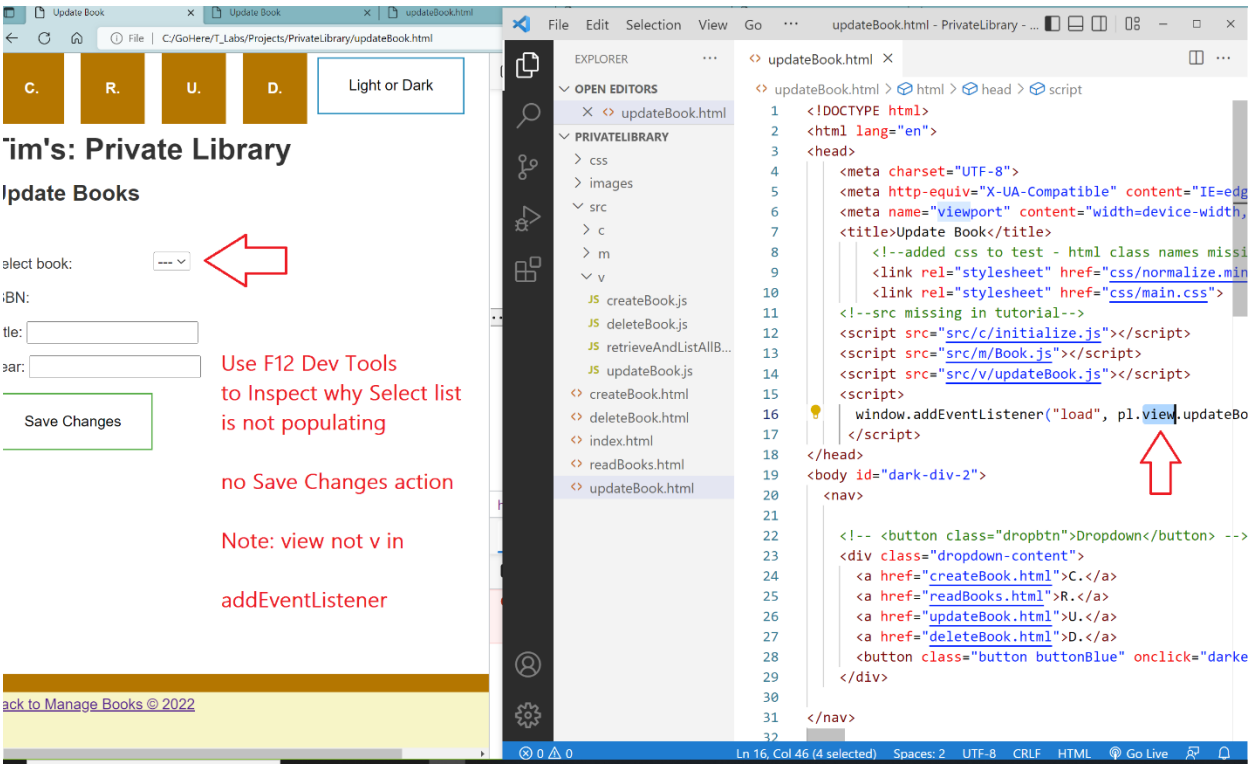


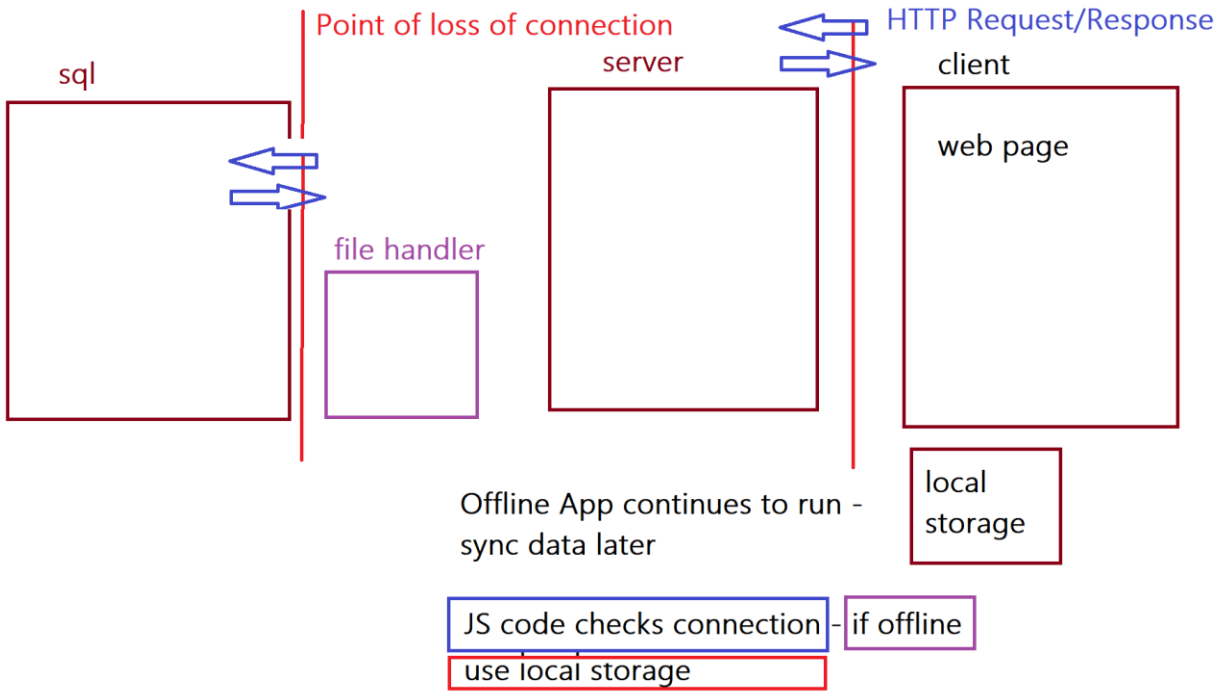
ISBN	Title	Year
006251587X	Weaving the Web	2000
0465026567	Gödel, Escher, Bach	1999
0465030793	I Am A Strange Loop	2008

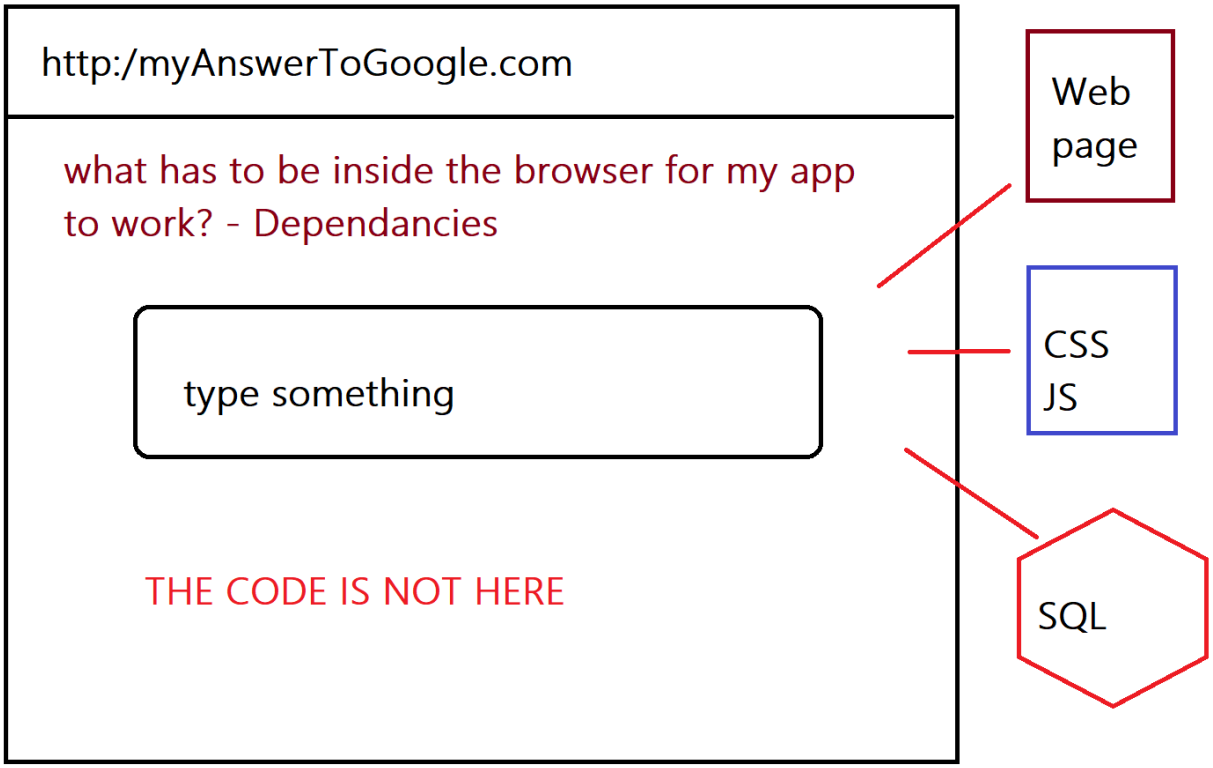
What will we do with it code: (Some anyway – What is the "Logic" or "Workflow" or "Job"?)

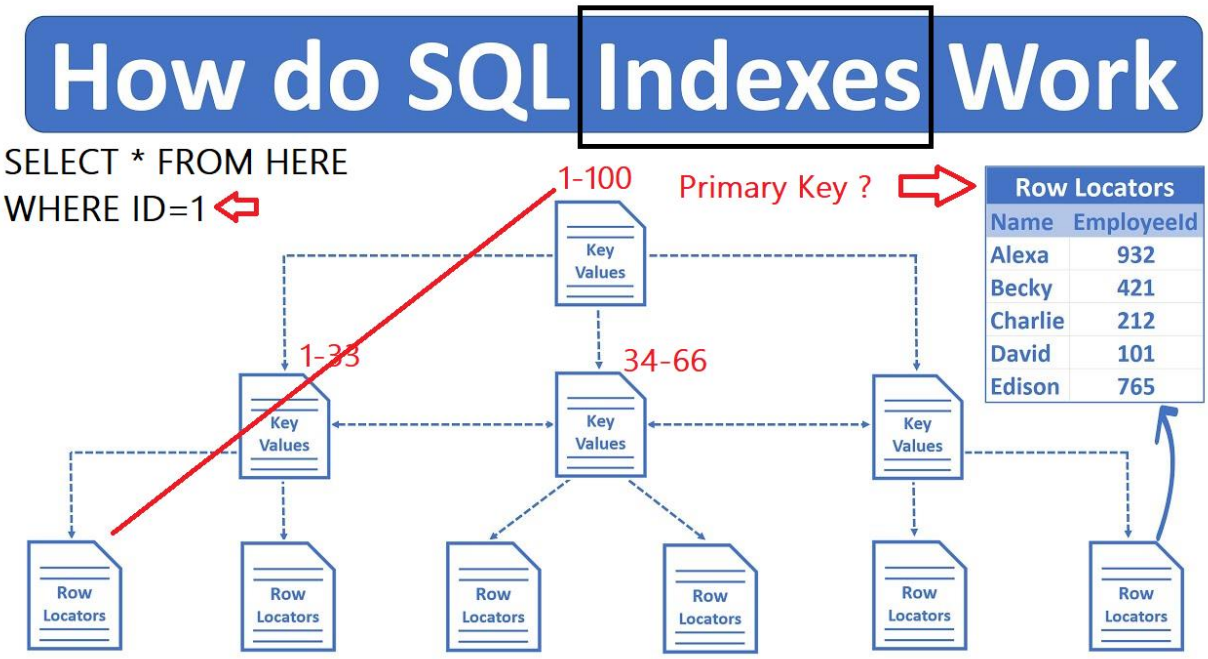
Load content (FETCH) - comparisons - testing – controlling program flow - logging to console – model data




C.
R.
U.
D.
Light or Dark
'im's: Private Library
Update Books
elect book:
BN:
tle:
ear:
Save Changes
Use F12 Dev Tools
to Inspect why Select list
is not populating
no Save Changes action
Note: view not v in
addEventListener
ack to Manage Books © 2022
updateBook.html - PrivateLibrary - ...
EXPLORER
OPEN EDITORS
updateBook.html
PRIVATELIBRARY
css
images
src
c
m
v
createBook.js
deleteBook.js
retrieveAndListAllB...
updateBook.js
createBook.html
deleteBook.html
index.html
readBooks.html
updateBook.html
updateBook.html > html > head > script
window.addEventListener("load", pl.view.updateBo


Point of loss of connection
HTTP Request/Response
sql
server
client
web page
file handler
local
storage
Offline App continues to run -
sync data later
JS code checks connection - if offline
use local storage





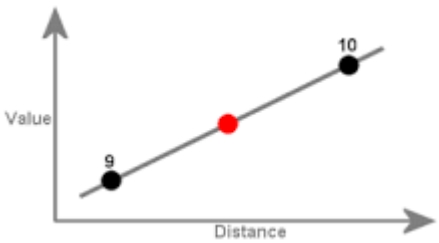
Tools:

1. Loops
2. If

3. While
4. Comparisons
5. booleans: https://www.w3schools.com/js/js_booleans.asp
6. RegEx: https://www.w3schools.com/JS/js_regexp.asp
7. maps vs. sets: JSON ?
8. Events: Thing code reacts to
9. https://www.w3schools.com/js/js_events.asp
10. JavaScript typeof (w3schools.com)
11. space

Vocabulary starts with you…

1. What is an example of interpolation?
    a. JavaScript Template Literals (w3schools.com)
    b. JavaScript string interpolation is the process of embedding an expression into part of a string. A template literal is used to embed expressions. You can add values such as variables and mathematical calculations into a string using interpolation.
    c. String Interpolation in JavaScript (dmitripavlutin.com) – good discussion examples
    d. Or;



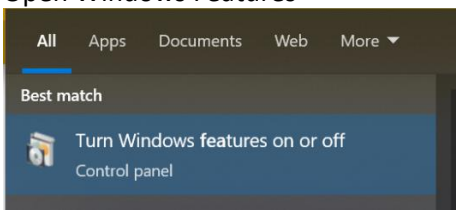
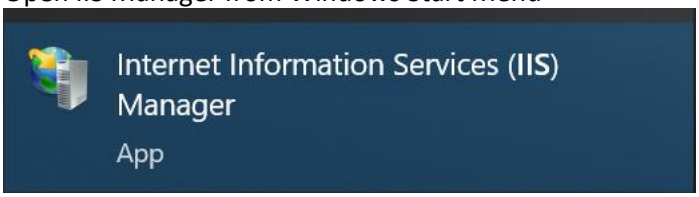
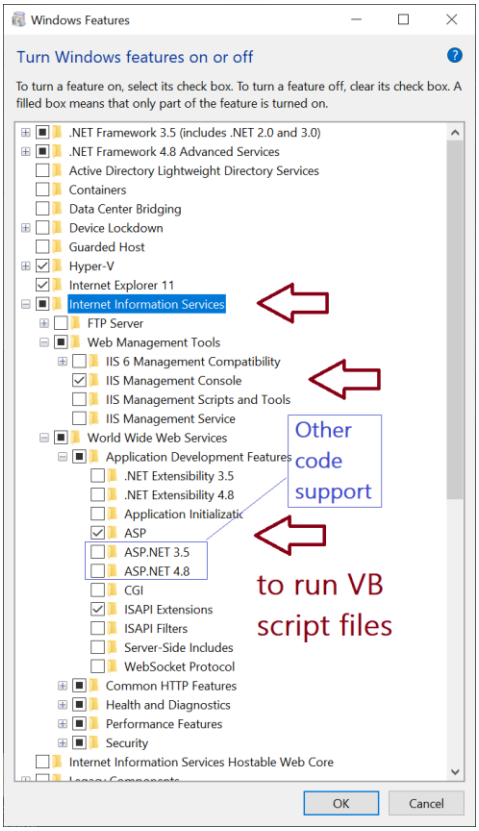
e. Interpolation is the process of estimating unknown values that fall between known values. In this example, **a straight line passes through two points of known value**. You can estimate the point of unknown value because it appears to be midway between the other two points.

2. Literals; are the constant values assigned to the constant variables. Literals represent the fixed values that cannot be modified. It also contains memory but does not have references as variables.
    a. Following the ordinary or usual meaning of the words I'm using the word in its literal, not figurative, sense.
    b. Example, const int =10; is a constant integer expression in which 10 is an integer literal.
    c. Tryit Editor v3.7 (w3schools.com)
3. Backtick key
    a. Alternatively known as acute, backtick, left quote, or an open quote, the back quote or backquote is a punctuation mark (`). It's on the same U.S. computer keyboard key as the tilde.
    b. 

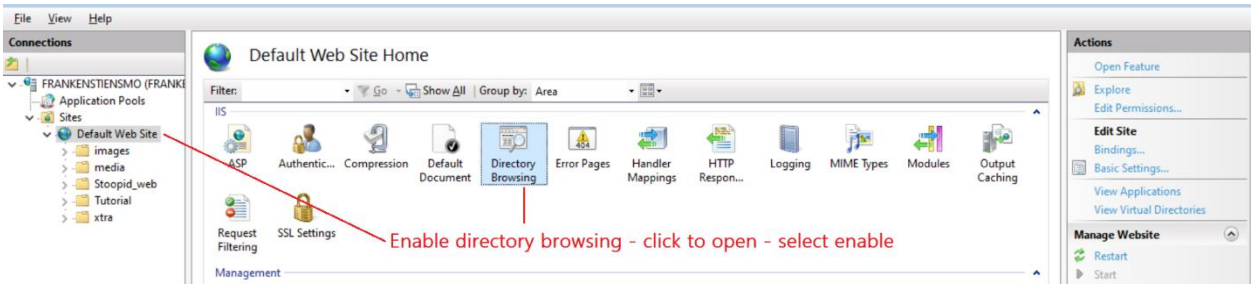
4. NaN is a JavaScript reserved word indicating that a number is not a legal number.
    a. Trying to do arithmetic with a non-numeric string will result in NaN (Not a Number):

5. space

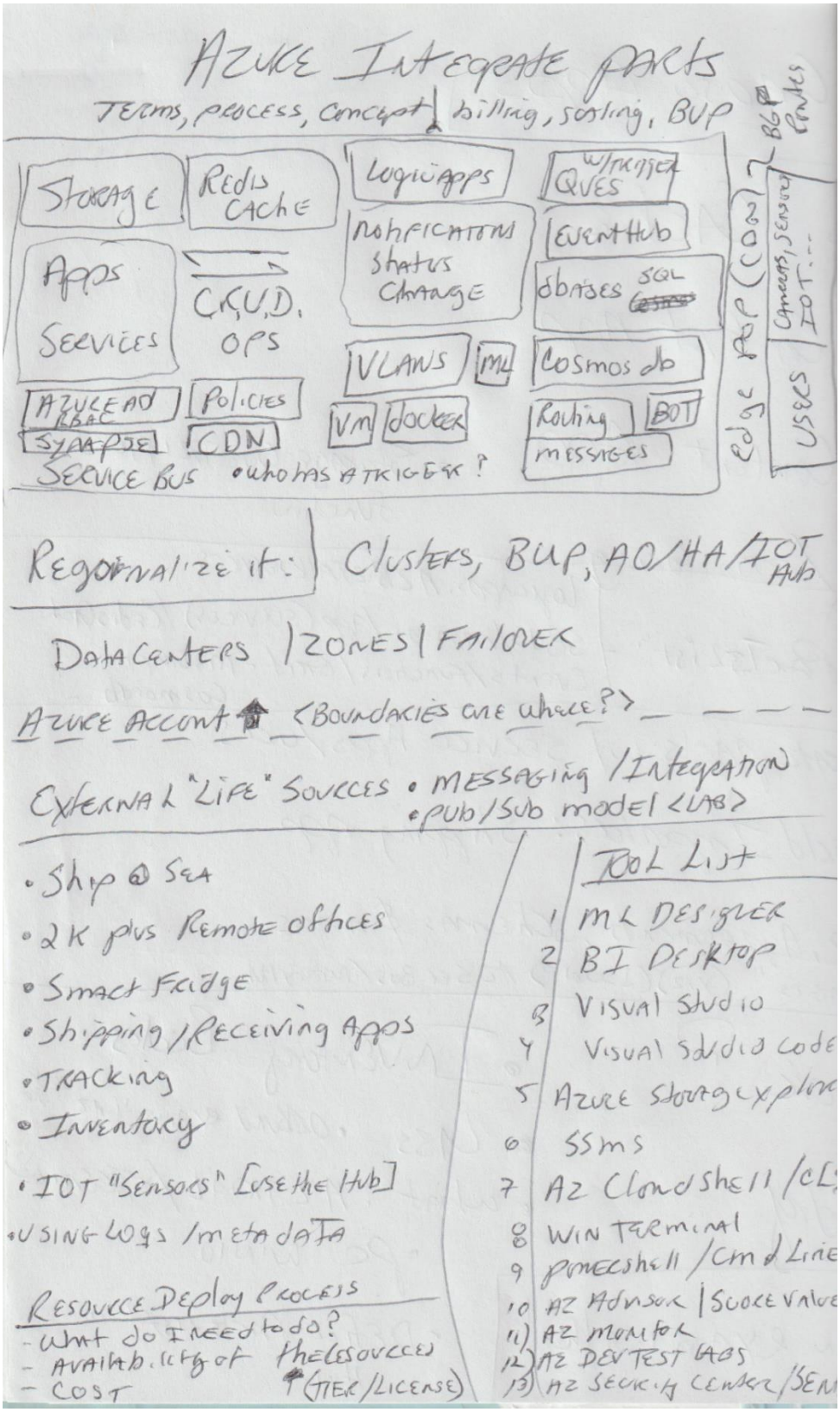
Setup Internet Information Server:

Open Windows Features	Open IIS Manager from Windows Start Menu
	
Select 	

Enable Directory Browsing – Dev only – Show localhost pages as a list



Integrate with Azure



References

1. First JS Lab:
    a. https://docs.microsoft.com/en-us/learn/modules/build-simple-website/
2. First Python Lab:
    a. https://docs.microsoft.com/en-us/learn/modules/intro-to-python/
3. ML w/Python
4. Introduction to machine learning - concepts - why do I...
    a. https://docs.microsoft.com/en-us/learn/modules/introduction-to-machine-learning/
5. Coding Tool Jupyter:
    a. https://jupyter.org/
6. VS Code/Studio
    a. https://www.bricsys.com/blog/computer-programing-a-brief-history
    b. https://teachablemachine.withgoogle.com/
    c. https://machinelearningforkids.co.uk/#!/pretrained
7. DevTools
    a. https://docs.microsoft.com/en-us/microsoft-edge/devtools-guide-chromium/landing/
8. Command line tools:
    a. Windows Terminal
    b. Cmd:
    c. cURL: https://curl.se/docs/manual.html
    d. PowerShell
    e. Azure CLI
    f. https://docs.microsoft.com/en-us/learn/paths/get-started-with-artificial-intelligence-on-azure/
9. concat vs math
10. data structures - array -what gets accessed as an array - string/html objects
11. functions
12. add in toward functions etc...
13. Date - Time - (extra formatting)
14. Querystring: JSON mapping source to target
    a. Node.js Query String Module (w3schools.com)
    b. https://www.w3schools.com/jsref/prop_loc_search.asp
    c. Tryit Editor v2.2 - Show ASP (w3schools.com) – server side example
    d. Tryit Editor v3.7 (w3schools.com) – change QS in web page link
15. Encode/Decode URI: Handling special characters in URL
    a. https://www.w3schools.com/tags/ref_urlencode.ASP
    b. https://www.w3schools.com/jsref/jsref_encodeuricomponent.asp
16. stringify()
    a. https://www.w3schools.com/jsref/jsref_stringify.asp
    b. C++ Pointers and References (ntu.edu.sg)
17. JavaScript typeof (w3schools.com)

Python

18. Intro: https://docs.microsoft.com/en-us/learn/modules/intro-to-python/

19. Get started using Python on Windows for scripting and automation:
    a. https://docs.microsoft.com/en-us/windows/python/scripting
20. Get started with Python in Visual Studio:
    a. https://docs.microsoft.com/en-us/learn/modules/python-install-vscode/
21. Get started using Python for web development on Windows:
    a. https://docs.microsoft.com/en-us/windows/python/web-frameworks
22. Visual Studio | Python documentation:
    a. https://docs.microsoft.com/en-us/visualstudio/python/?view=vs-2022space
23. space