**VSCode Extensions**

Recommended

1. esbenp.prettier-vscode
2. formulahendry.auto-close-tag
3. hex-ci.stylelint-plus
4. dbaeumer.vscode-eslint
5. naumovs.color-highlight
6. DigitalBrainstem.javascript-ejs-support

Optional

1. ritwickdey.LiveServer
2. erikphansen.vscode-toggle-column-selection
3. file-icons

[www.codepen.io](http://www.codepen.io)

this is link where we can see live version of your web side in this we can write code and see same time how it’s look like.

We have header tag and can use <h1> </h1> to till h6 and it goes smaller with number increment

All this kind of documents are available on **MDN** and **w3school**, **devcode.io**

To give same we use brake tag <br> and doesn’t need to close this like header just add where we want space. This is call self-closing tag

We also have horizontal row and its use to give line and must not have end tag same like <br>

<hr> also support other property(attribute) like size = “” or color, noshade , width.

<hr>

<h1>The Adventures of <br> Sherlock Holmes </h1>

<br>

<h3> by</h3>

<br>

<h2> Arthur Conan Doyle </h2>

<hr>

To comment in html we have to use <! - - your msg - ->

We need boiler plate or kind of templet new we have init plugin so we just need to type html and it will give that.

All this code and sorcut working fine on aotm IDE

<https://github.com/nwinkler/atom-keyboard-shortcuts>

This is the link were you can find shortcut for atom for all 3 os

<https://docs.emmet.io/cheat-sheet/>

to know all doc type

meta for give other information like un coding basically we used utf-8

<!- - This is my msg to you - -!>

<center>

<hr size="3" noshadow>

<h1>The Adventures of <br> Sherlock Holmes </h1>

<br>

<h3> by</h3>

<br>

<h2> Arthur Conan Doyle </h2>

<hr size="3" noshadow>

</center>

We can add paragraph by use <p> </p>

For italic use <em> </em> don’t use <i> and for bold don’t use <b> use <strong>

For list.

We use unordered list it used like

**<ul>**

**<li>** This one point**</li>**

<li> This second point</li>

**</ul>**

**To get number we have to <ol> </ol> it will give you numbers**

Consider we want roman number or rank should start with 7 we have to defined that property in opening tags

<ol type= “i”> # for roman number rest all same for li

<ol start=”7”> # numbers start from 7

**Images In HTML**

This is online image cropping tool

<https://crop-circle.imageonline.co/>

<https://www.befunky.com/create/resize-image/>

<img src=”pathOfImage or can be url” alt=”text that we want to show ”>

<img src="img/amardip.png" alt="profile image"> # in this we have added image in img folder

**Hyper link**

We use Anker tag **<a href = link> text that you want </a>** just type a and it will auto complete.

Also you can give your local html page name so that work as link

**<a href = hobbies.html > text that you want </a>**

**Tables**

Tr = table row in side that we have **td** for table data means coalman

We have <thead> for table header in we can create **tr and td** and it’s show in bold <th>

<tbody> for table body

<th> same as <thead> but can be formatted

<table>

<tr>

</tr>

</table>

<table>

<thead>

<tr>

<th> date </th> # this for header and bold text

<th>works</th>

</tr>

<tbody>

<tr>

<td>2013 to 2017</td>

<td>Max as Sr.IT</td>

</tr>

<tr>

<td>2018 to 2021</td>

<td> licious</td>

</tr>

</tbody>

</thead>

</table>

Now we working with layout normally we do this with css but let’s try with pour html

Now we want our text details in front of our image so for that we have to use table and in which we have separate column

Inside table we can use celspasing = 20 get space of 20

**Form**:

It work well with JS but for we see

First type form and it will auto complete

Than we can add **input** and will give text input

For button input just changes input text to **submit in type**

In input we have many type like color or email, file, date and time, checkbox, password

In this we have 2 part label and input in side input you can give default value.

<form **action="index.html**" **method="post">** # in action menas when we submit it will take you on index.html we ca use <mailto:youremail@gmail.com>

# **method="post”** can be use **msg as wht input we get**

<label><strong>username</strong></label>

<input type="text" name="" value=" ">

<input type="submit" name="" value="submit">

<input type="button" name="" value="my button">

<label> wloud you like to except it </label>

<input type="checkbox" name="" value="">

<input type="datetime-local" name="" value="">

<input type="file" name="" value="">

<input type="image" name="" value="">

<input type="range" name="" value="">

<input type="password" name="" value="">

<input type="color" name="" value="">

</form>

For multiline text input we have to use textarea

<textarea name="name" rows="8" cols="80"></textarea>

CSS (cascading style sheet)

We can add our css code in side our html tag like changing a background color mostly we use style key word and give other attribute.

style=”background-color: blue;”

we can use hexadecimal instead of name blue

but this is not looking good to find best color we have web site name **colorhunt.co** where we can get a hex code for color. Ex #EAF6F6

but changing in html make many error and can be difficult so we can make internal css so we don’t need to insert css code in each html tag in this we smartly add css code in body but this is not good so we write css code that applicable for entire html page

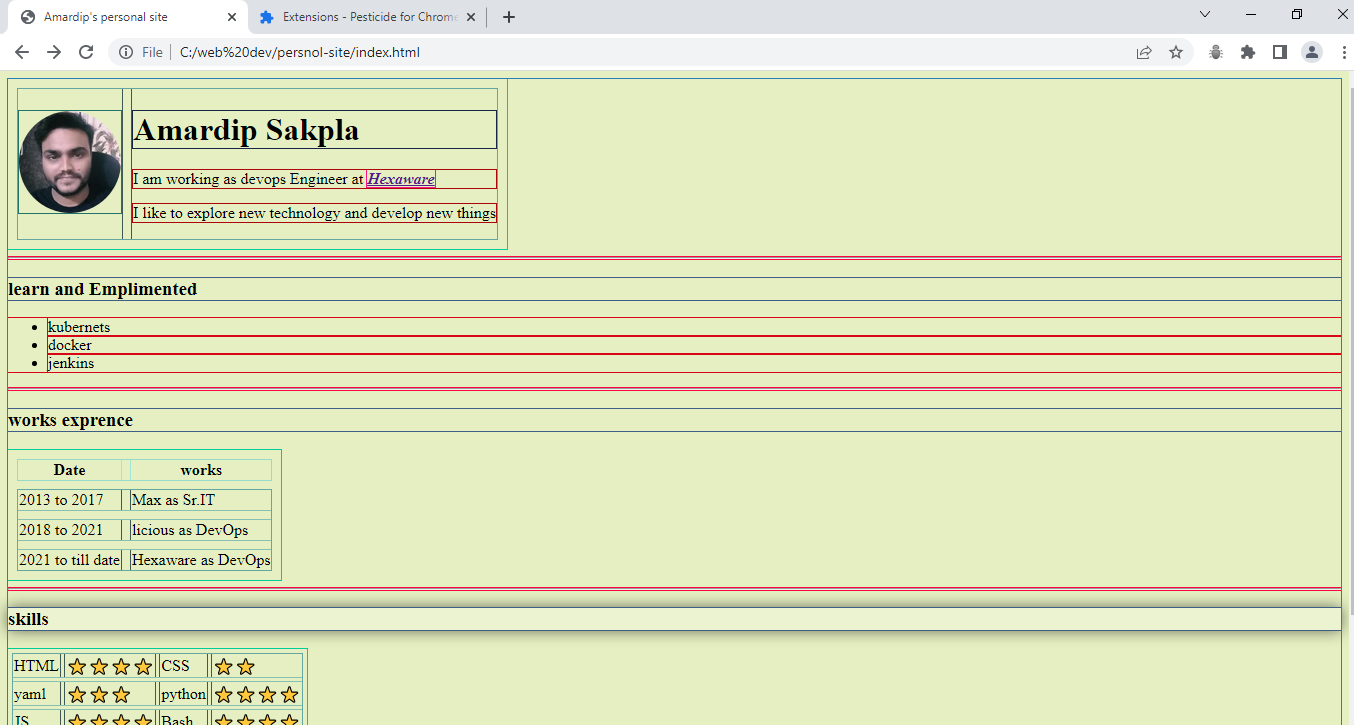
we have to create a style tag right after title before body and before head and inside this style tag give a name of tag which is applicable for this css code in our case body { } inside this braises we have to use css code **background-color: blue;** that’s it.

Note: few element has default css so when we make change in that tag it will never show you changes

This is extension show you how css alien page <https://chrome.google.com/webstore/detail/pesticide-for-chrome/bakpbgckdnepkmkeaiomhmfcnejndkbi>

When you install this you have to go to details of extension have to find Allow access to file URLs

Now when you on side you can view the alignment like below



As we see border has 4 side top left and right and left so we can only use one for use like dot

<style>

body{

background-color: #E5EFC1;

}

hr{

border-style: none;

border-top-style: dotted;

border-color: white;

border-width: 5px;

height: 0px;

width: 10%

}

</style>

**External CSS**

For this we have to create one folder name as css and inside that create styles.css

In side this file we have code without <style> tag and then and in index.html we have to add link just type link and it will auto complete give your style.css path that we can give like css/style.css

<link rel="stylesheet" href="css/styles.css"> # you can past this code in any html and your css code will be applicable for that also.

Debugging:

We can use chrom menu > more tools developer’s tool in this we have a console there you can see error.

Css: syntax

Tag\_name { property\_name: value; } or selector { property\_name: value; }

h1 { color: read;}

can have more than one role in css

h1 { color: read; font-size: 200px}

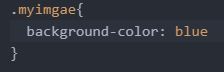
In this case we have a images and we want style different for each image so we add class and give any name to that class in your html

In css we have to call that class instead tag and add .infront of it

HTML



CSS



We can use one or more classes to apply more than one property like my image and circular image

HTML:

<img class="myimgae **circular** " src="img/amardip.png" alt="profile image">

CSS:

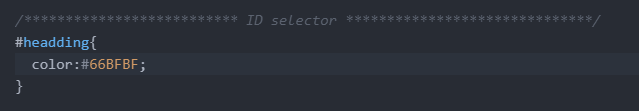
. **circular** {

Border-redius: 100%

}

ID selector





As like class we have id selector also

It’s seems like ID and class are working same but this is not a case

In class we can use same name repatriate but in ID you can’t use same name again and again

We can use one or more classes name to apply more than one property and this not working in ID

To change the state of element when we hover on that

Selector: hover {

Property: value;

}

Ex:

img:hover{

background-color: gold;

}

New project:

In this project we are going create a site which look very professional

We see there is icon on title of your web site called as favicon

www.favicon.cc is

if your updated favicon is not showing you have add ?v=2 at end of your icon name

Div: use to group the element together here we group h1 and p

<div class="">

<h1>This is Amardip</h1>

<p> Let's fun</p>

</div>

When we use div we see there is default margin and we have to remove that so we make a change in css code.

div{

background-color: blue;

}

body{

margin: 0

}

h1{

margin-top: 0

}

When we use tags right after each other it will take that in next line and but there is exception for this and that is

<Span>

<Img>

<a>

In other hand block element never allow to take other element right or left

And we can change this property by css code that is

display: inline; or display: block and display: inline-block;

we can hidden element like

display: none; or visibility: hidden;

position:

**static** is default one

**absolute :** in this if we more left it will actually move to left

**relative**: this related to previous position of element. if we jet relative 30 means it will move 30 from his original location.

**Coordinates**: that tis top bottom, left, right like this. In this it move opposite like we say left 20 means it will move right 20

**Fixed**: it will stay there when your scroll also like your tab bar

Margin: **auto** that help you to get the center and we have to give left, top, right and down

Font-family: there is 6 type of font family inside that we can use different font

Font-family: forntname, fontfamily\_name;

font-family: verdana, sans-serif;

but it’s very as per os to achieved it have this site

<https://www.cssfontstack.com/> not more useful

<https://fonts.googel.com/>

once you select the font their u get link for font and you have to use this HTML below to title

and in css copy for from same side

<https://www.flaticon.com/> where we can get icon or images for website

or gihpy.com for GIF

we have font-size: it take values in px but u can give in % also

100% = 16px if we want font 90px means we have to do 90\*16 =562.6

Fornt-size:562.5

Other way using em instead of px

1 **em** = 100% means 16px

**rem** is good thing which never get effected from hieraticalforn

<https://css3buttongenerator.com/>

this is the link to create buttin code you can copy it in ccs button class

bootstrap:

This is for front library

[www.codeply.com](http://www.codeply.com)

this is for paly ground where we can add any library

or get

getbootstrap.com

where we can download example

installing bootstrap:

go to getbootstrap.com and get link

<https://getbootstrap.com/docs/5.1/getting-started/introduction/>

and copy below code and past in your html

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

<https://dribbble.com/> here you can find many website sample.

Sneakpeekit.com

This is the link where you can download wirefrike mean scaltan

<https://balsamiq.cloud/> create wirefrike mean scaltan online

for navbar or link bar we have a van element in html <nav>

<nav> is working same as div

<nav>

<ul>

<li>

contact

</li>

</ul>

</nav>

Above is the normal code for html we have add boot start lass we can refer documentation

From getbootstrap.com there you can find many key work you can use any of them in same class

For collapse function to work on your browser may be need java script also

So we can add javascript code below to our bootstrap installation code

And all three lines

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/popper.js@1.12.9/dist/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmYl" crossorigin="anonymous"></script>

Now you see collapse is working

And this is the boot strap class added html

<nav class="navbar navbar-expand-lg">

<ul class="navbar-nav">

<li class= "nav-item">

<a class="nav-link" href="">Contact</a>

</li>

<li class= "nav-item">

<a class="nav-link" href="">Price</a>

</li>

<li class= "nav-item">

<a class="nav-link" href="">Download</a>

</li>

</ul>

</nav>

We can change the color of navbar is by below property

navbar-dark bg-dark

for margin in starting

m= margin

s= starting

auto = this to fix as possible as left

<nav class="navbar navbar-expand-md navbar-dark bg-dark">

<a class="navbar-brand" href="">tindog</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">

<ul class="navbar-nav ms-auto">

<li class= "nav-item">

<a class="nav-link" href="">Contact</a>

</li>

<li class= "nav-item">

<a class="nav-link" href="">Price</a>

</li>

<li class= "nav-item">

<a class="nav-link" href="">Download</a>

</li>

</ul>

</div>

</nav>

**new project web site with css**

**https://github.com/londonappbrewery/TinDog-Start**

In the coming lessons, we'll be learning about Bootstrap while we build and design our TinDog website.

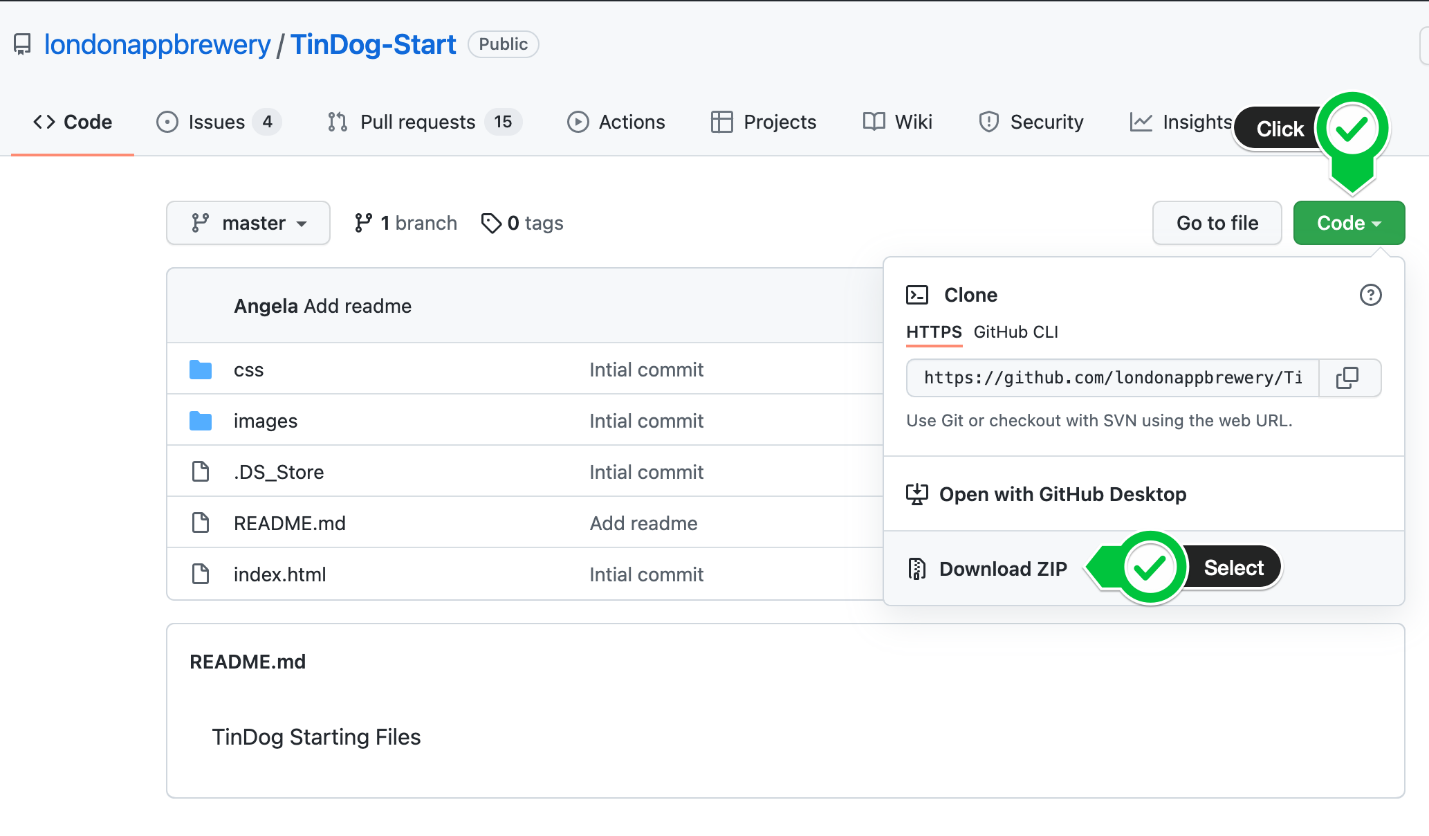
I've created a skeleton project with all the graphics and some skeleton HTML for you to work with.

1. Head over here:

https://github.com/londonappbrewery/TinDog-Start

2. Click on the Code dropdown then click on Download ZIP

3. Extract the ZIP file and move it to your Desktop.



N.B.Throughout the course, we'll often be using skeleton files to save you time typing out all the HTML and text you see on screen. If you want to have extra practice creating the HTML elements, then simply follow the above steps then double click on the index.html file to see it rendered in the browser then you can use this as a starting point to create your own index.html file from scratch.

When we open code we find css and image folder also boiler templet for html

Dynamic grid view : when we see product on amazon on laptop it has 4 line and when we see mobile here is only 2 line of products.

So in web development responsive means dynamic as per device. Change the view as per mobile, tablet, laptop.

In css class=”col-6” # it’s column 6 use to get only 50% your screen the logic bind in very row we have 12 column 6 is half and 3 means 25%

<body>

<div class="row">

<div class="col-6" style="background-color: red; bordar:1px solid;">

My-column

</div>

</div>

</body>

But above code is not responsive so we use **col-md-6** class in internal div so it get responsive to see difference we add 2 div .

<body>

<div class="row">

<div class="**col-md-6**" style="background-color: red; bordar:1px solid;">

My-column l-1

</div>

<div class="col-md-6" style="background-color: red; bordar:1px solid;">

My-column-2

</div>

</div>

See in specific size of device we can make it specific number of column and that we can achieved by add to that size tyle like

<div class="row">

<div class="**col-lg-3 col-md-4 col-sm-6**" style="background-color: red; bordar: 1px solid">

col

</div>

<div class="col-lg-3 col-md-4 col-sm-6" style="background-color: red; bordar: 1px solid">

col

</div>

<div class="col-lg-3 col-md-4 col-sm-6" style="background-color: red; bordar: 1px solid">

col

</div>

<div class="col-lg-3 col-md-4 col-sm-6" style="background-color: red; bordar: 1px solid">

col

</div>

</div>

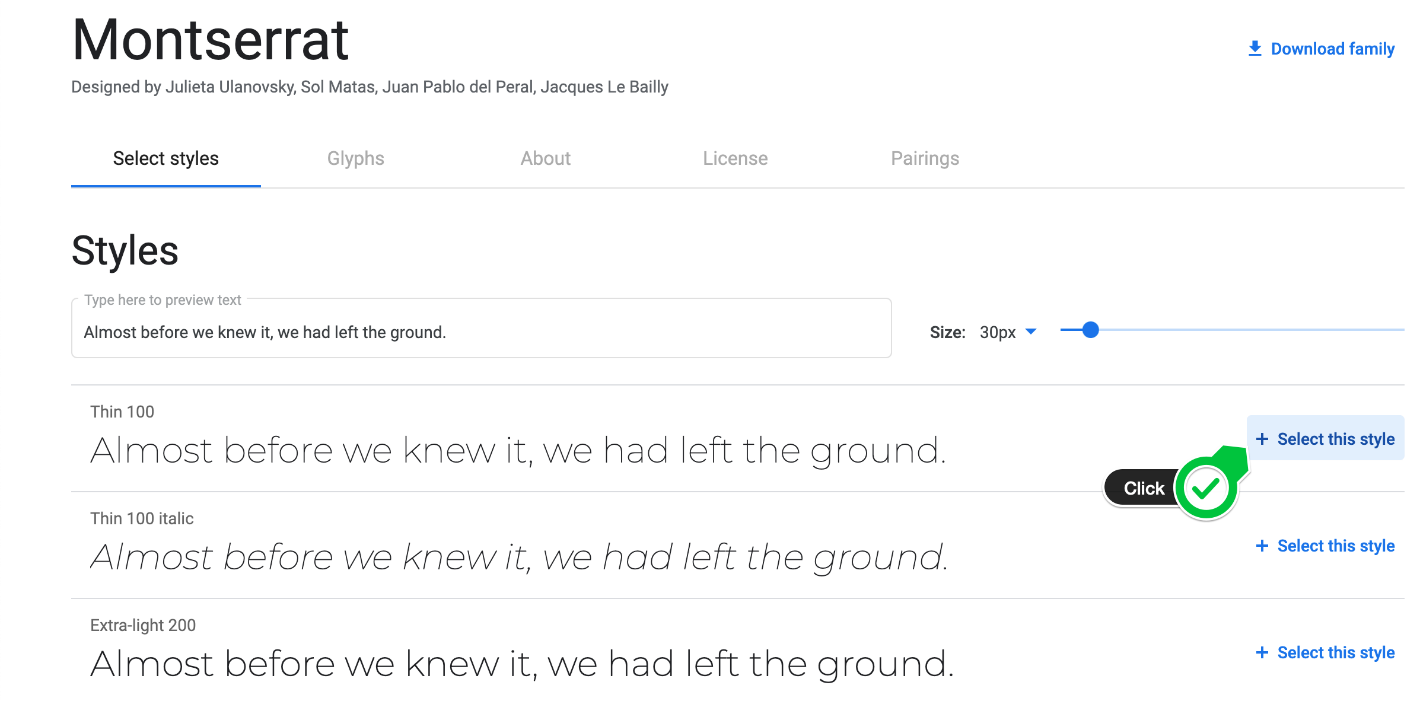
**Getting Montserrat Black and other Font Weights**

**Tl;DR If you have problems getting a Montserrat-Black to work check back to this note for the solution.**

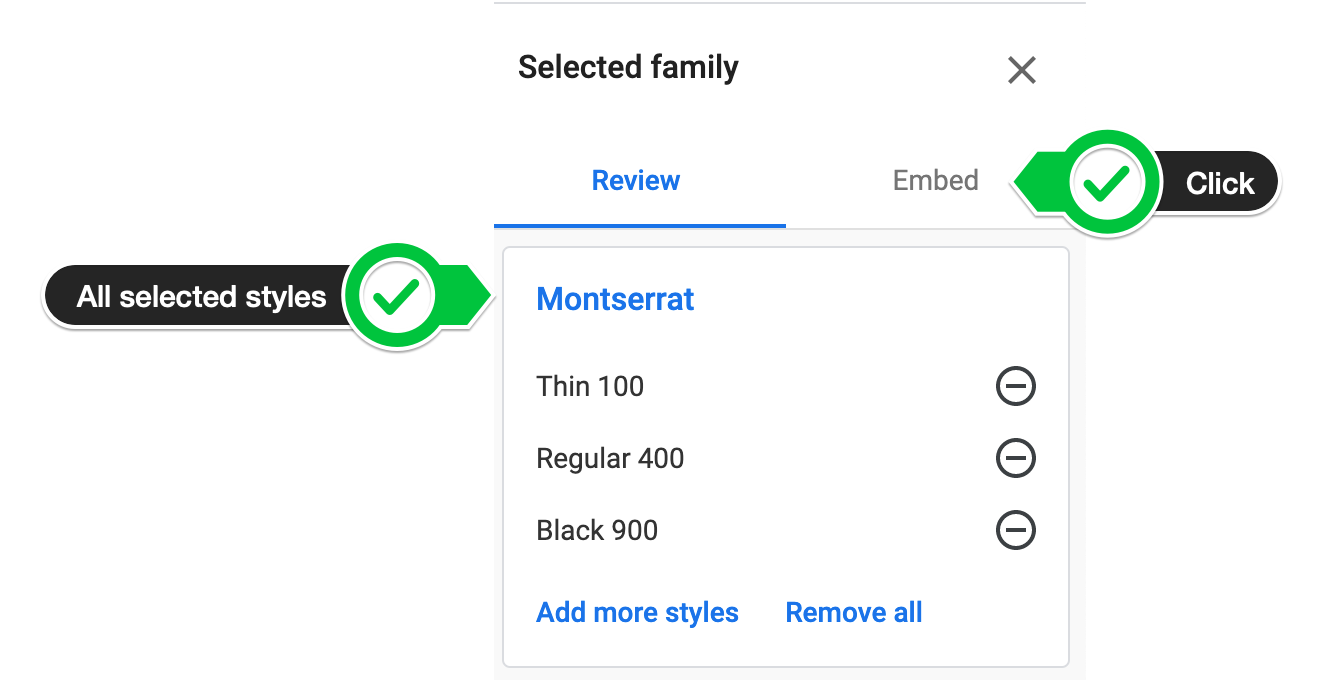
Hey guys,

In the next lesson, I'll be introducing you to Google fonts. Some students have had issues getting the bolder Montserrat black to show up because Google has updated their website UI. In order to get the different font weights of the Google font, you need to **include** it when you select it in Google fonts.

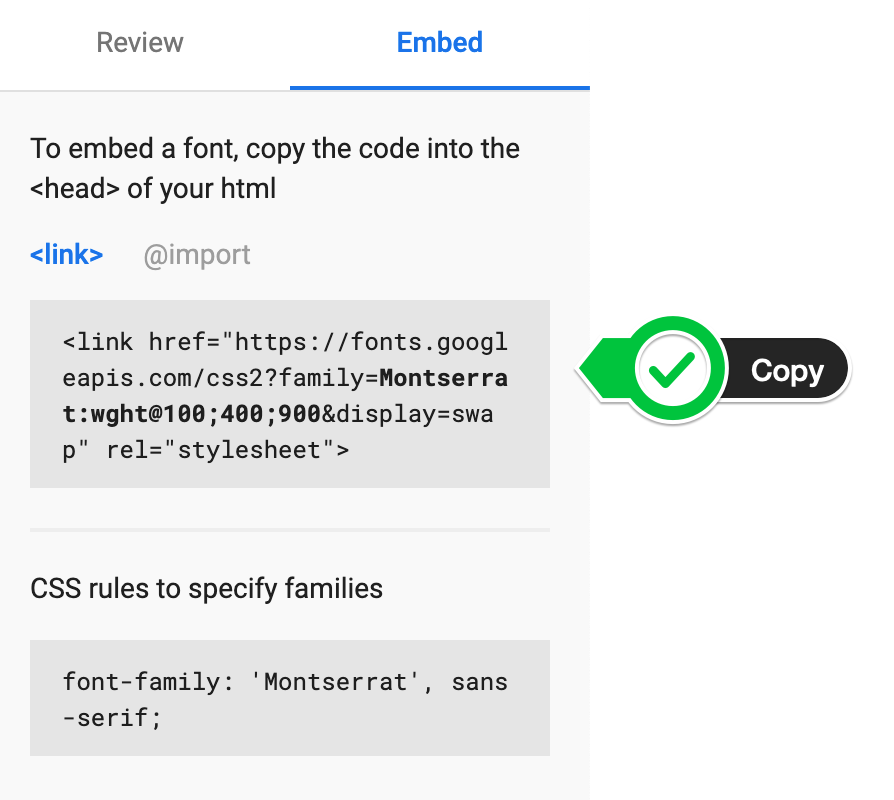
So when you are on the font page, make sure you select the 100, 400 and 900 font weights, e.g.



Next, look at the right-hand pane and make sure that you have all the desired font weights and then click on the Embed tab.



Now you should be able to copy the stylesheet link for the font and use it in your CSS.



So the final code should look like this:

1. <link href="https://fonts.googleapis.com/css2?family=Montserrat:wght@100;300;400;500;900&family=Ubuntu:wght@300;400;700&display=swap" rel="stylesheet">
2. h1 { font-family: 'Montserrat, sans-serif'; font-size: 3rem; line-height: 1.5; font-weight: 900;}

note: font size always defined in rem

Hey guys,

A quick note about the next lecture. If you find that your custom styles are not working, make sure that you change the order of link tags in the header.

CSS styles are applied in the order they are linked in your HTML code.

So if you had two stylesheets e.g. styles1.css and styles2.css which both target the same element e.g.

styles1.css

body {

background-color: red;

}

styles2.css

body {

background-color: blue;

}

If inside the head section if your HTML, you list your links as this:

<link rel="stylesheet" href="styles1.css">

<link rel="stylesheet" href="styles2.css">

The resulting page will be blue.

But if you listed your links like this:

<link rel="stylesheet" href="styles2.css">

<link rel="stylesheet" href="styles1.css">

The resulting page will be red.

Essentially both styles are being applied, but the one that's visible at the end is the one applied last.

So following that logic, if your custom styles are not overriding the bootstrap styles, all you need to do is move the link to your custom stylesheet to a line after the bootstrap CDN link:

<link href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" rel="stylesheet">

<link rel="stylesheet" href="css/styles.css">

This means that you first load the default bootstrap styles, then you can override some of those components with your own custom CSS.

Unlike CSS and JavaScript, HTML code is executed from top to bottom so the order of your code matters.

All the best,

Your instructor, Angela

**Container:**

Save way we can use container in just change the div class to **container** or **container-fluid**

**You can get icon or img for your buttons and other resources from this link**

<https://fontawesome.com/>

from this link you have add cdn in our html code as like bootstrap and the you can copy code for that particular icon

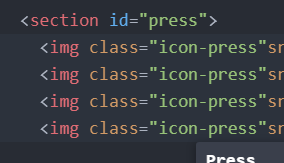
<script src="https://kit.fontawesome.com/324f5b9f5a.js" crossorigin="anonymous"></script>

Note: write the msg and hold **CTRL + /** it will **comment out**

We can use id if class is not working to select element in css

When we make **border-radius: 100%** it look round(circle)

If we have **Sublime-Style-column-selection** then we can use **, + alt** and drag andyou can typetype that add in all line



Carousel = this is the slide show

In this we have many property like

data-ride="false" data-interval="1000"

| **Name** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| interval | number | 5000 | The amount of time to delay between automatically cycling an item. If false, carousel will not automatically cycle. |
| keyboard | boolean | true | Whether the carousel should react to keyboard events. |
| pause | string | boolean | "hover" | If set to "hover", pauses the cycling of the carousel on mouseenter and resumes the cycling of the carousel on mouseleave. If set to false, hovering over the carousel won't pause it.  On touch-enabled devices, when set to "hover", cycling will pause on touchend (once the user finished interacting with the carousel) for two intervals, before automatically resuming. Note that this is in addition to the above mouse behavior. |
| ride | string | false | Autoplays the carousel after the user manually cycles the first item. If "carousel", autoplays the carousel on load. |
| wrap | boolean | true | Whether the carousel should cycle continuously or have hard stops. |

When you want to slide manually by clicking at that time

For first dive item keep active and other more active class or all show it same screen.

<div id="testmonialcarousely" class="carousel slide" data-ride="false" >

<div class="carousel-inner">

<**div class=" carousel-item active**" >

<h2 id="h2">I no longer have to sniff other dogs for love. I've found the hottest Corgi on TinDog. Woof.</h2>

<img class="testimonials-img" id="testimonials-img" src="images/dog-img.jpg" alt="dog-profile">

<em>Pebbles, New York</em>

</div>

<**div class="carousel-item ">**

<h2 id="h2" class="testimonial-text">My dog used to be so lonely, but with TinDog's help, they've found the love of their life. I think.</h2>

<img id="testimonials-img" class="testimonial-image" src="images/lady-img.jpg" alt="lady-profile">

<em>Beverly, Illinois</em>

</div>

</div>

<a class="carousel-control-prev" href="#testmonialcarousely" role="button" data-slide="prev">

<span class="carousel-control-prev-icon" ></span>

<span

</a>

<a class="carousel-control-next" href="#testmonialcarousely" role="button" data-slide="next">

<span class="carousel-control-next-icon" ></span>

</a>

</div>

We can get the different as same like getbootrap.com and <https://bootsnipp.com/>

Here you can get html as well as css whit bootstrap.

We have ready to use price table on getbootrap site you can edit that and use for your web side

But we can create your own. Like this

<body>

<div class="card-deck">

<div class="card">

<div class="card-header">

card-heade

</div>

<div class="card-body">

card-body

</div>

</div>

</div>

</body>

And then you can garnish it and finally it look like below all code is push on repo of web-dev

<div class="row">

<div id="pricing-column" class="col-lg-4 col-md-6">

<div class="card">

<div class="card-header">

<h3>Chihuahua</h3>

</div>

<div class="card-body">

<h2>Free</h2>

<p>5 Matches Per Day</p>

<p>10 Messages Per Day</p>

<p>Unlimited App Usage</p>

<button type="button" class="btn btn-lg btn-block btn-outline-dark">Sign Up Free </button>

</div>

</div>

</div>

<div id="pricing-column" class=" col-lg-4 col-md-6">

<div class="card">

<div class="card-header">

<h3>Labrador</h3>

</div>

<div class="card-body">

<h2>$49 / mo</h2>

<p>Unlimited Matches</p>

<p>Unlimited Messages</p>

<p>Unlimited App Usage</p>

<button type="button" class="btn btn-lg btn-block btn-dark">Sign Up</button>

</div>

</div>

</div>

<div id="pricing-column" class=" col-lg-4">

<div class="card">

<div class="card-header">

<h3>Mastiff</h3>

</div>

<div class="card-body">

<h2>$99 / mo</h2>

<p>Pirority Listing</p>

<p>Unlimited Matches</p>

<p>Unlimited Messages</p>

<p>Unlimited App Usage</p>

<button type="button" class="btn btn-lg btn-block btn-dark">Sign Up</button>

</div>

</div>

</div>

</div>

Zindex this is more advance concept which help to cut view of one or more element.

Here is the example html and css code for div

<body>

<div class="red">

HI

</div>

<div class="yellow">

hi

</div>

<div class="blue">

Hi

</div>

</body>

div{

height:100px;

width: 100px;

border:1px solid;

**position: absolute;**

}

.red{

background-color:red;

**z-index: 1; <!- - this is take front 🡪**

}

.yellow{

background-color:yellow;

**left:20px;**

**top:20px;**

}

.blue{

background-color:blue;

**left:40px;**

**top:40px;**

**z-index: -1; <!- - this is take background if we use - 1 🡪**

}

If we remove z-index element our layout is totally different the accepted values are

1 = front

-1 = background

0 = current position

css media query : this feature helps us adapt view as per small device also.

The tag can be used like

@media <type> <feature>

Below code make a color change of of h1 when max-width: 900px and less

html

<h1>

hello word

</h1>

css:

@media (max-width: 900px){

h1{

color:red;

}

}

We have added nav bar but we never attach any action on that button so we are going attached section Id so it will render like that.

And we just have add that in href=”#jdksfkd”

Same is working url level you see when give #thatId with url you redirect to that perticluer page

Now we are try DRY do not repeat your self method:

h1, h2, h3, h4 {

}

In this way we hitting same property for all tag separated by ,

We can use **id and class** combination to hit specific element this also hirarical order means your class should be the part of your id . here separate by space

#mynavbar .container-fluid{

}

In above case we are only hitting to container-fluid which in inside #myvabr id

When there is no any space in 2 select it go without haraci means it look for to select in one element

<h1 class=”one” id=”two” >

hello word

</h1>

.one#two{

Color red

}

Web design:

Color theory:

Red: love, energy, intensity that’s why we see most car add with red color BG

Yellow: joy, intellect, attention,

Green: freshness, growth, safety, many glossary company use this. Or any food

Blue: stability, serenity, trust.

Purple: royal, wealthy, femininity.

If we are using 2 color take two side color means look very same. Good for logo, body, and nav bar

And take exactly opposite color so that can pops, don’t do for text and it’s BG

Use this site to look how our color will look like.

<https://color.adobe.com/>

<https://colorhunt.co/> This is ready to use color panel. Many hot and popular type

Typography:

We know there is manly 2 font family

Serif : which has feet for stand character

Sans-serif: is good font for simple, and good for body text

In one view just use 2 fonts not more than that. And make sure both font are in same mood, time era, one of that can be bold.

Give bold and bright color which particular word you want to highlighted

Alignment for each item should be same.

Add white space that make you more luxuries

UX (user experience): drive users automatically

Make sure below point

Simplicity

Consistent for design and user experience

Note human reading point is like **F** starting from top and left then go down so that’s why we have important things at right of your web side.

Another view pint is **Z**

<https://www.dailyui.co/>

This is web site which give task to design website and in below url you can see how they looks like

<https://collectui.com/>

<https://www.canva.com/en_gb/> This is online graphic design site sign up for free and

click on create new design here you can create a logo and all here we are just creating other thing also like website look like.

Here you can publish as web page and see how it look as web site this just view not code make note

Java script: it’s interrupter language like python and ruby fast and work browser side manly use to manage or make a liveness to html element.

Initially it make to just work with HTML but now days it’s used for backend and logic also.

We can start testing our java script in chrome browser> developer tools> console > alert(“hello”); this will show you alert on browser.

You have to use shift to write on second line enter is not take you through next line only run your code

We have kind of editor in developer tool source > Filesystem > snippets> create new file.js and edit it and you can run it.

To clear console history use ctrl + k or right click and clear console.

But we see last variables are still there in page so we can remove that refreshing page

Now you can follow MDN document for java script

<https://github.com/rwaldron/idiomatic.js#naming> This is also good documents for programing

to know data type of the your input we can use

typeof(12)

typeof(“hello”)

variable:

the way we can defined the variable is

var myName = “Amardip”;

to update variable you doesn’t need to use var you can simple used myNmae= “Mahi”

This is how we can use variables in and call variables

var myName="Amardip";

alert (myName);

This is how we can take users input and then we can use it in our program.

var myName= prompt("Enter your name" );

alert ("hello " + myName);

variable names:

give meaning full name

can’t begin with number you can start with character and then have number inside it

use underscore for word separator

only $ and \_ are valid symbol in variable.

Use camel case myNameOfMovies

We can join concanet string by using +

To get length means number of character we use **variable.length;**

In JS you can comment by //msg or /\* msg \*/

var myMsg = prompt("Type your msg");

var msgSize = myMsg.length;

alert("You have enterd " + msgSize + " charcter, You left only" + (140 - msgSize) + "character" );

slice: it help you to cut and chop the string can take x and y as parameter.

That x = starting of the string this all work as index or as array

Y = end of the string where you want to cut the staring.

var.slice(x, y)

var myMsg = prompt("Type your msg");

var msgSize = myMsg.length;

alert("You have enterd " + msgSize + " charcter, You left only" + (140 - msgSize) + "character" );

alert("we are going to cut down you msg and end result look like this" )

alert( myMsg.slice(0, 140)); // this is the main line of code which actually chop down

we can reduce the code also like below

alert(prompt("Enter your msg").slice(0, 10));

var.toUpperCase() // use to change the all character as upper case which is there in var

var.toLowerCase()

name= prompt("Enter your name");

firstChar=name.slice(0,1).toUpperCase();

alert( "Hello " + firstChar + name.slice(1,name.length).toLowerCase());

The above code make your first char as capital and rest all lower no matter how you type.

Math operator: all are working same as normal math operator work only

Var a = 3 + 3;

Var m= 3\*4;

Var s=7 -4;

Var d= 12 / 4;

Only modulo is working differently

Var mo= 9 % 6;

Var test = 3 + 10 \* 2 // in this first multiplication happen and then it do addition this rule

But you can make this more clear by adding ( ) prentesy and also change the order of handling operation

Var test = (3 + 10 )\* 2 // in this it will do addition first and then multiplication

dogAge= prompt("Enter your Dog Age");

alert("if your dog was a man than he was now: " + ((dogAge -2 ) \* 4)+ 21 );

x++ // this is familiar to you increment by 1

x- - // for decrements

function: it’s a block of code which is run by calling it’s name

syntax:

function name(params) {

//your code

}

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

**function** myFunction() {

// your code

} // don’t used ; here at end of function { }

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

// calling function

myFunctio();

naming convention is same as variable for function.

function getMilk() {

console.log("move up")

console.log("move down")

console.log("enter in shop")

console.log("take milk")

console.log("give money")

console.log("move up")

console.log("move down")

console.log("enter in home")

console.log("give it to me")

}

//calling function

getMilk();

now we are looking parameter for function and we have to add that parameter in function and when we are calling function we have to pass that parameter in ( )

function getMilk(bottles) {

console.log("take " + bottles + " milk")

}

//calling function

getMilk( 12);

for we can do any operation on user input

//declaring function

function milkPrice(bottles) {

console.log("The Totale Price of " +bottles + " Milk bottles is " + bottles \* 20 )

}

//calling function

milkPrice(12);

Math.floor( 15 / 3); This will make a round of not get fluting point

function milkPrice(money) {

var numberOfbottles = Math.floor( money / 1.5);

console.log("You are going to by " + numberOfbottles);

}

//calling function

milkPrice( 4);

return type in function:

return can be any number, variable

function milkPrice(money) {

var numberOfbottles = Math.floor( money / 1.5);

console.log("You are going to buy " + numberOfbottles);

return money % 1.5; // here we defined the return and that give model

}

//calling function

milkPrice( 4);

Math.random(); // this is use in js to create random number it’s 16 decimal 0.92992 like

Dis number generator

var n = Math.random();

n = n \* 6;

var diec= Math.floor(n + 1 );

console.log(diec);

love percent random calculating

var yourName = prompt("Enter your name ");

var yourLover = prompt("Enter Your love one number");

var lovePercent = Math.random()

lovePercent = lovePercent \* 100 + 1

console.log("your love Percent is " + Math.floor(lovePercent) + "%")

Condition:

If

The only deferent here we used === for equal to

var lovePercent = Math.random()

lovePercent = lovePercent \* 100 + 1 ;

**if (lovePercent > 70 ){**

alert("your love Percent is " + Math.floor(lovePercent) + "%" + " Great your have lots of chances")

}**else{**

}

console.log("your love Percent is " + Math.floor(lovePercent) + "%")

=== equal to. we can use = = also but in this it never checks the data type

!== Not equal

> greeter than

< lesser than

>= greeter than equal

<= lesser than equal

var lovePercent = Math.random()

lovePercent = lovePercent \* 100 + 1 ;

if (lovePercent > 70 ){

alert("your love Percent is " + Math.floor(lovePercent) + "%" + " Great your have lots of chances")

}

else if (lovePercent < 30 ){

alert("This is not your game" + " your love Percent is " + Math.floor(lovePercent) + "%" )

}else if (lovePercent > 30 && lovePercent < 70 ){

alert("Think one more time" + " your love Percent is " + Math.floor(lovePercent) + "%" )

}else{

console.log("your love Percent is " + Math.floor(lovePercent) + "%")

}

Above one is the complete example of the if else and else if

Here we use and && operator we also have 2 more

&& And

|| or

! Not

Array:

We var but it never store more than one item so we have array it help us to store more than one item.

We can update array and add remove item form it. And way of creating array is like below

Var number = [1, 2, 3, 4, 5 ] # use [ ] and , use to separate.

And array is start with 0

We can get the array length like number.length;

var getstList = ["Amar","Ajay","mahi","yash"]

console.log(getstList[0]);

console.log(getstList.length);

As we use **in** to check item is present in array or not we we use **includes** and that give you true or false in return.

**getstList.includes(name)**

var getstList = ["Amar","Ajay","mahi","yash"];

var name = prompt("ENter your name :");

if( getstList.includes(name)){

alert("wllecome " + name);

}else{

alert("May be next time");

}

Array\_name.push(“values”);

Means put a values in array that whay we use push and it always push item at the end of array

Array\_name.pop(“values”);

// this is to remove the last item from array.

This is the fizzBuzz challenge

var output= [];

var count = 1;

function fizzBuz(){

if(count%3 === 0 && count%5 ===0 ){

output.push("fizzBuz");

} else if ( count%5 ===0 ){

output.push("Buz");

}else if ( count%3 === 0 ){

output.push("fizz");

}else{

output.push(count);

}

count++ ;

console.log(output);

}

// calling function

fizzBuz();

taking this code one level a head

var output= [];

var count = 1;

function fizzBuz(){

**if(count%3 === 0){**

**if (count%5 === 0){**

**output.push("fizzBuzz");**

**}else{**

**output.push("fizz");**

**}**

} else if ( count%5 ===0 ){

output.push("Buz");

}else{

output.push(count);

}

count++ ;

console.log(output);

}

In above case we have nested if else in our code.

While ( condition ){

// your code

}

While run until condition not satisfied

var i =1

while (i < 101){

console.log(i);

i++

}

Now we can call this while loop in fizzBuzz function inside code is below

Note: be careful while loop may be create infinity loops also

var output= [];

var count = 1;

function fizzBuz(){

**while (count <= 100){**

if(count%3 === 0){

if (count%5 === 0){

output.push("fizzBuzz");

}else{

output.push("fizz");

}

} else if ( count%5 ===0 ){

output.push("Buz");

}else{

output.push(count);

}

count++ ;

**}**

console.log(output);

}

1. var numberOfBottles = 99
2. while (numberOfBottles >= 0) {
3. var bottleWord = "bottle";
4. if (numberOfBottles === 1) {
5. bottleWord = "bottles";
6. }
7. console.log(numberOfBottles + " " + bottleWord + " of beer on the wall");
8. console.log(numberOfBottles + " " + bottleWord + " of beer,");
9. console.log("Take one down, pass it around,");
10. numberOfBottles--;
11. console.log(numberOfBottles + " " + bottleWord + " of beer on the wall.");
12. }

For loop: here we doing var declaration; condition ; and increment

for (starting point; ending point ; changing can be increment or decrement ){

//your code

}

for (i=0 ; i <10; i++){

console.log(i) //any code you can put here

}

----

This code will run 10 time

Now we are going see how we can do fizzbuzz with for loop

var output= [];

function fizzBuz(){

**for ( var count= 1; count <= 100; count++ ){**

if(count%3 === 0){

if (count%5 === 0){

output.push("fizzBuzz");

}else{

output.push("fizz");

}

} else if ( count%5 ===0 ){

output.push("Buz");

}else{

output.push(count);

}

**}**

console.log(output);

}

Where to use which loop

While loop : use where we have to check state

For loop: use to iterate, like iterate on files.

Now we are going to use java script in our website:

When we want to load or run the java script from html code we have to use it in any html tag we many use in body

<body **onload="alert('Hi Amardip');"> #** we use alert just to msg we use single quote onload is IMP key word

<h1>Hello</h1>

</body>

Other way is add script in body like below

<script type="text/javascript">

alert("Hello word")

</script>

Call JS file from HTMLM

<script src="**index.js**" charset="utf-8"></script>

// we have index.js in same folder but you separate folder than you can mention path.

Note: consider if we have a JS to modified html tag and if that html tag is not yet created than our JS will failed. So try to mentions JS at the end or somewhere we have created that html tag

DOM (Document object module):

html tree generator extension: to see how the element are placed

we can select html element by using **document.firstElementChild**

And the hirarkey go down and down

document.firstElementChild. firstElementChild.

And we can store this in var like

Var **head** = document.firstElementChild.firstElementChild

And then we can change the it using js

**head**.**innerHTML**= “Good Bye”;

# here the main key is innerHTML and we changed it to Good bye. We can also change property like color changes and all.

**header.style.color="red"**

we can do some operation on this click press a button. And **querySelector() in this we can pass action so here we pass input and give click method**

document.querySelector(“input”).click();

so every object has a property and method

property: to describe about object

object.color; # get property

object.color= red; # this is to set the property

ex: innerHTML, style, firstChiled

Method: what object can do

object.drive(); # this will give you functionality the this way we call method

ex: clieck() appendChild() setAttribute()

we can get a elements by it’s tag name and if we have more than one element with same tag than it will give you array

document.getElementsByTagName(“li”);

now it will give you a values in array but can’t directly change ths values of array like below

document.getElementsByTagName("li").style.color="red";

The above code will give you error because we never pass which item of array we want to change.

**document.getElementsByTagName("li")[2].style.color="red";**  # this is how it will change the color of thread item

in same way we have a document.getElementsByClassName(“btn”) in this we are going to select the element by it class name. in this also we have to use array no matter we have single item or multiple

document.getElementByID(“myImg”) This is only take single item in this case we doesn’t need array

document.getElementById("title").innerHTML="Good Bye";

in query selector we can use ID, class or element itself and defined same as we use in CSS like ID # class .

element h1 or li

**document.querySelector("#title").style.color= "Blue";**

But IMP is we can search with combination like inside li (list item) find a(ancon tag)

**document.querySelector("li a ")**

IT work same as we see in CSS for space and . to combined 2 elements.

**document.querySelector("li.item ") # mostly used**

when you query get have multiple matches than you will get only first match as per your query

what if we want all element which is match with your query tha we have to All

**document.querySelectorAll("li.item ")**

in this we have to use index as per other selector to get exact items

Note: we can change the html element just different is in css we have \_ but in JS we use camel case

fontSize = JS

font\_size =CSS

document.querySelector("button").style.backgroundColor="yellow";

we can get the classes of the element like

document.querySelector("button").classList

we can add class in html by using js like

document.querySelector("button").classList.add(“myclaass”) # now you can see this class in HTML code.

document.querySelector("button").classList.toggle(“myclaass”)

# toggle work as if this class in applied than remove it if not applied then add it.

In this case we have created huge class property in css file and using below code we are going to add huge class in our html now we that css rule is applicable on html tag h1

document.querySelector("h1").classList.add("huge")

we can change the text by using **textContent =”urtext”;** or **innerHTMl= “urtext”**

**but the innerHTML is good it give you all HTML property and you can change all HTML property**

**innerHTMl= “<em>urtext</em>” make italic**

but in textContent your just passing text you never have power to update the HTML

we can get the any tags attribute inside any tags in below case we are getting the value of herf it’s actually url

same we way we can set the value for any attribute in that case you have to give to parameter in which in is like herf and other one is your updated URL

document.querySelector("a").**getAttribute**("**href**") # you can only view

document.querySelector("a").**setAttribute**("href" , "**https://techmahi.ml**") # can modify

Event listener this use to assign the action to any button. **addEventListener()** inside this first parameter is type that can be click or anything and second we can call a function or add any code to perform the action.

Note: when u calling function in this don’t give **()** just name it or it will run that function at same time when it loads never wait to get button click.

We have many type listeners but for now we are just using click listener

document.querySelector("button").addEventListener("click" , **handelClick**);

function **handelClick**(){

alert("I got click ");

}

We can call anonymous function, as name say we don’t have name to function so we just have to defend function inside the event listener. As it hasn’t had name i

document.querySelector("button").addEventListener("click" , function (){

alert("I got click ");

});

Note: when we open developer tool we see all html tag has sumber with $ like $0 , $1 and that we can access though the console

$0.innerHTML=”new datat”;

Till now we see the function that take parameter to use in the function but we can pass the other function and pass this current parameter for that function. It look like below

Ex:

Function name\_of\_fun( par1, par..n, **operator**){

return operator(par1, par..n)

}

Calling function

name\_of\_fun( par1, par..n, **other\_fun\_name**)

example:

function add(num1, num2){

return num1 + num2

}

function mull(num1, num2){

return num1 \* num2

}

function cal(num1, num2, operator){

return operator(num1 , num2)

}

// calling function

cal( 2, 3, mull) # here we are passing the parameter to mull function

And we can all

Cal(2,3,add) # here we are passing the parameter to add function

We have one more tool called as debugger just type debugger and call function with all parameter

debugger

function\_name(1,2, otherFunction);

Audio play: we can play audio in JS like below

var audio = **new Audio**(path/songs.mp3');

audio.**play()**;

we can use this code in any function and than call this function where we want

see in our code we have button and we call same method whichever button get pressed but now we need to know which button has pressed. That we can see by using key word **this**

**console.log(this.innerHTML);**

now we are not only button html code but we are dig into it and get text which is assign to that text

we can created a object for storing info it more like array or like but store in key values per

var nameOfVar = {

name: “mahi”,

age: “25”,

languages: [“hindi”,”marathi”]

}

We can retrieve the values of object like

nameOfVar.name # this will give you name

nameOfVar.age # this will give you age

constriction function: it many used to create or feed a data make you easy. All word letter should be capital not like camel case

function **BellBoy**( name, age, haswork, languages){

this.name = name;

this.age = age;

this.haswork = haswork

this.laguages= laguages

}

TO create object for this function like below way.

Var bellBoy1 = **new** **BellBoy**(“mahi”, 19, ture, [“hinidi”,”marathi”])

Var var\_name = new **constarttion\_function\_name** (“mahi”, 19, ture, [“hinidi”,”marathi”])

See this one is more essay and shorter way of creating object

Switch: we know the switch it’s use to get the check expression if is match or true then check the use check case is that true than run the code and break to exit on this point run other code

if expression is match with case than it will run the code or break iy

**Default** is like else key word in if else

Note case start with : and end with break;

Ex: here we are matching button text if that is match with case than it run block of code which is inside of that case.

var buttonTxt= this.innerHTML;

switch (buttonTxt) {

case 'w':

var tom1 = new Audio('sounds/tom-1.mp3');

tom1.play();

break;

case 'a':

var tom2 = new Audio('sounds/tom-2.mp3');

tom2.play();

break;

case 's':

var tom3 = new Audio('sounds/tom-3.mp3');

tom3.play();

break;

case 'd':

var tom4 = new Audio('sounds/tom-4.mp3');

tom4.play();

case 'j':

var snare = new Audio('sounds/snare.mp3');

snare.play();

break;

case 'k':

var crash = new Audio('sounds/crash.mp3');

crash.play();

break;

case 'l':

var kickBass = new Audio('sounds/kick-bass.mp3');

kickBass.play();

break;

default: console.log(this.innerHTML)

}

In constriction function for object we can add function in it means our object has function attached to it.

function BellBoy( name, age, haswork, languages){

this.name = name;

this.age = age;

this.haswork = haswork

this.laguages= laguages

**this.moveSuitcase = function(){**

**alert(“may I take your suitcase”);**

**pickupSutcase();**

**move();**

**}**

}

To creating object

Var bellBoy1 = **new** **BellBoy**(“mahi”,

19,

ture,

[“hinidi”,”marathi”]

, **moveSuitcase: function(){**

**alert(“may I take your suitcase”);**

**pickupSutcase();**

**move();**

**}**

)

To call this function

bellBoy1.moveSuitcase();

keyboard key event:

**TL;DR keypress is now deprecated, you should use keydown instead.**

 you should be using the **keydown** event listener instead of **keypress**.

**Keydown** is method use to know keyboard key is press but we have to active this on entire web page or it will work only or that particular html element so we use document to listen on entire page.

This how we can get the which key is press on keyboard but it give you other information too and we are interested in key and for this we have a **key** option

document.addEventListener("keydown", function(event){ console.log(event)});

document.addEventListener("keypress", function(event){ console.log(event)});

document.addEventListener("keypress", function(event){ console.log(event.key)});

we can pass one function input to other function also

document.addEventListener("keypress" , function(**event**){

**makeSound(event.key);**

});

In this we are passing one more function in this we have makesound we called it callback function it wait to finish something

Inside that function al we can pass event to check the what event occur in this we are passing event.key to mkaeSound function

Here we called it as event but you can call it e or env anything you want

Means this **function** get call when keypress event happed on our document(web page)

We have a setTimeOut function that wait for some time

It take function we can use anonymous function also , code , time our in milisec.

Jquery: This JS library most used library in JS it’s reduce code so you don’t have to write much code

How to incorporate jquery in our web site

Here also we have to use CDN use popular CDN good if you use google CDN.

URL of google CDN:

<https://developers.google.com/speed/libraries#jquery>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

**This should be a above (first) to jS cdn in your HTML code**

Shouldn’t be there in head section

In jquery we have a short form of document.querySelector that’s $ or jquery

$("h1").css("color","red"); # this is the basic code of jquery which will check h1 color to red by changing it’s css property.

But we see if there have jquery cdn is not loaded in than it will never work for this we can check is this loaded or not if loaded than your jquery work

$(document).ready(function(){ $("h1").css("color","red"); });

In this we check entire document is ready and once jquery is ready then in callback function run in this function we are making changing h1 color to red it help when your jquery cdn not in body of html or not before js cdn. If that placed right then you did not need to use this code also.

Minifiyer: this use reduce the size of your code file and work exactly same. Remove comment and space

$("h1") # in this we are selecting h1 but we can use one than one tag combination same like css

$("#body h1")

$("#body.h1")

$("bodyh1")

We can select one or many element(like querySelectorAll) of html like same way

$("h1") # can select one or many h1

$("button") # work like querySelectorAll can select many buttons to

If we never pass the second values of css function then it will work as get method

$("h1").css("color"); # it will give current color of h1 you can store it in console

console.log($("h1").css("color"));

We can add any class which is created in CSS that we can assign to any html element like below

Css code:

.bigfont{

color: yellow;

font-size: 10rem;

}

Js jquery code

$("h1").addClass("bigfont") # we have create bigfont in css than assign it h1 tag

Same way we can remove the any class from html

$("h1").removeClass("bigfont") # this will remove class from html tag

$("h1").addClass("bigfont newFont") # we can add multiple class in same quote “” just separate them by **space**

We can do cross check also is that html elemet has class by below code. By using hasClass

$("h1").hasClass ("bigfont newFont") # it will check bigfont and newFont are present or not

We can manipulate text using jquery also there is 2 method for this

$(“h1”).text(“ newText”); # this will only manipulate text

For manipulating a html code as like innerHTML we have html() class

$(“h1”).html(“ <em>newText </em>”); # this we we can add html also

Manipulating attribute:

$(“img”).attr(“src”) # this will use to get the source of image we can add this to var or print it by console.log

To set attribute value we have to use second parameter as new value for attr function

$("a").attr("href" ,"http://techmahi.ml"); # in this we are setting a url or a tag from google to techmahi

To get all class of any element we can use below code

$("a").attr("class");

Adding eventListner using jquery:

We have .click method that and inside that we can add the callback or anonymous function that run when particular event is happened.

$("h1").click(function() { $("h1").css("color", "red")});

When we want to change the color by pressing any button in normal js we used for loop to iterate on all button but in jquery we can use below code and it will make your life easy.

$("button").click(function() { $("h1").css("color", "red")});

Same way we are checking which key is pressed on input text box and given log of it

$("input").keypress(function() { console.log(event.key)});

For entire web page we have to use document without “” in place of input

$(document).keypress(function() { console.log(event.key)});

Here is the task for us we want to change h1 text what we type in our web page

The answer:

$(document).keypress(function() { $("h1").text(event.key)});

Shorter way of the event listener is using on method it take 2 input first to what it has to listen that can be click, keypress, mouseover etc. and second one is the callback function

$("button").on("mouseover", function() { $("h1").css("color", "red");});

$("h1").on("click", function() { $("h1").css("color", "red");});

$("input").on("keypress", function() { $("h1").css("color", "red");});

We can add or remove new html element

$("h1").**before**("<button>My\_button<button>"); # this will add button before h1 tag

$("h1").**after**("<button>My\_button<button>"); # this will add button **after** h1 tag

$("h1").**prepend**("<button>My\_button<button"); # this will add button in h1 tag at the start of the <h1> means just after the opining tag

$("h1").**append**("<button>My\_button<button"); # this will add button in h1 tag at the end of the <h1> but before the end of h1 tag means inside the h1 only

To remove any element.

$("button").remove() # will remove all buttons

$("h1").remove() # this will remove h1 tag

We have other method also where we can show and hide the element also or other animation

$("h1").hide(); # hide the h1 tag

$("h1").show(); # show back the h1 tag

$("h1").toggle(); # show back the h1 tag

$("h1").fedeOut();

$("h1").fedeIn();

$("h1").fedeToggle();

$("h1").slideUp();

$("h1").slideDown();

$("h1").slideToggle();

$("h1").animate({opacity: 0.5}); # animate is function that give ability to do custom animation we use opacity key here to light down 0.5 means 50%

We have other key margin

$("h1").animate({margin: “20%”});

We can to multiple animation by join it by .

$("h1").hide().show().animate({opacity: 0.5});

Run linux terminal on windows system

Install hyper terminal

Install git bash

Than open this like and copy content of this link into hyper terminal > edit > preference.

<https://gist.github.com/coco-napky/404220405435b3d0373e37ec43e54a23>

than cross check by running this cmd

echo $SHELL # output should be bin/bash

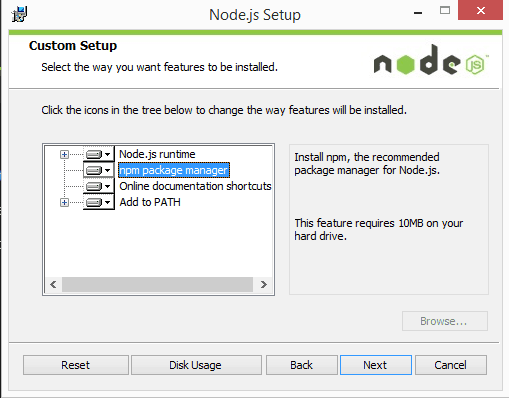
Backend development: this contain servers, application(logic), databases. Frontend + backend = full stack development.

What is node JS: it give power to interact with system it’s hardware and all. That we use for server side and perform task and logic.

**Install Node.js on Windows**

**Installation Steps**

1. **Download the Windows installer from the**[**https://nodejs.org/en/**](https://nodejs.org/en/)
2. **Choose the LTS version**that’s shown on the left.
3. **Run the installer** (the .msi file you downloaded in the previous step.)
4. **Follow the prompts in the installer** (Accept the license agreement, click the NEXT button a bunch of times and accept the default installation settings).



5.**Restart your computer.** You won’t be able to run Node.js® until you restart your computer.

6. Confirm that Node has been installed successfully on your computer by opening a Hyper terminal and typing in the commands node --version

You should see the version of node you just installed.

Now we have a node install we can run any js file on node we just have to go there in same give below cmd

node filename.js # it will give u a files output on console.

**REPL( read evaluation print loop ):** it’s give power to run ad hoc code as like cmd means same ass we run code in chrome console

To access it we just have to type node in our terminal than you will enter in your node REPL to exit use CTL + C

And here we can use same cmd which we have run on chrome console.

Note: Advantage of this you can give starting character and by tab auto complete you can see all possibility.

You can refer node js document on <https://nodejs.org>

Where we same lots for pre build method and we can incorporated that method in your code

const means constant it like variable we can defend it but can’t be change after.

Now we are creating one cons and assign fs method to it for get that method we use require key word

const const\_name = require(“module\_name”)

const fs = require("fs");

now we can use fs module to perform various task we have method like **copyFileSync** this use to copy file take 3 parameter first one source, distention , optional

const fs = require("fs");

fs.copyFileSync("file1.txt","file2.txt");

NPM: we see we have a seen the we can use module like fs for file system this from the node but if we want to use other people module (external module) in this case we have to use a NPM ( node package module)

<https://www.npmjs.com/>

NPM is preinstalled with nodejs to start your npm project we have to use init the npm it will ask you a project name and the version you can keep the same or change and ask many info like repo or other.

Once it done it will create one file package.json that content all information about packages

Go to npm side and search the package which we want there you can see how can we install it and use it also.

npm init # create a project and package.json

now we are installing superheroes

npm install superheroes

now add the index.js and add below code

const superheroes = require('superheroes'); # adding superheroes package

var name = superheroes.random(); # calling random method of it and adding var

console.log(name); # printing name var on console

| **Command** | **NPM** | **Yarn** |
| --- | --- | --- |
| Initialize a project | npm init | yarn init |
| Run tests for the current package | npm test | yarn test |
| Check for outdated packages | npm outdated | yarn outdated |
| Publish a package | npm publish | yarn publish |
| Run a script | npm run | yarn run |
| Manage local package cache | npm cache clean | yarn cache clean |
| Log in or out | npm login/logout | yarn login/logout |
| Install dependencies | npm install | yarn |
| Install packages | npm install [package name] | yarn add [package name] |
| Uninstall packages | npm uninstall [package name] | yarn remove [package name] |
| Update manager | npm update | yarn upgrade |
| Update packages | npm update [package name] | yarn upgrade [package name] |
| Install packages globally | npm install --global [package name] | yarn global add [package name] |
| Uninstall packages globally | npm uninstall --global [package name] | yarn global remove [package name] |
| Interactive dependency update | npm run upgrade-interactive | yarn upgrade-interactive |
| Run package remotely |  | yarn dlx |
| Check licenses |  | yarn licenses ls |

Express: this is the framework for node as we see jquery is for JS we have a Express it’s also do same things it help you to reduce the code

In node you can create a desktop application also but express is very specific, to web development

**Creating our first server node and express**:

Mksdir my-express-server

Cd my-express-server

Touch server.js

Npm init

This is the installation method of express you can find here

<https://expressjs.com/en/starter/installing.html>

we already created project we just have to install it by below cmd

npm install express

we reference express as app you can refer other name but this is best practice

const express = require("express");

var app = express(); # mostly use app

app.listen(3000); # we are staring server on 3000 port

when we run the cmd will never get back it’s hung but you can access your server on <http://localhost:3000/>

here also we can add the anonymous function to print on console that we have start server on 300 port

const express = require("express");

var app = express();

app.listen(3000, function()

{

console.log("server started on port 3000 you can access it http://localhost:3000");

});

We can access it on localhost but we are getting can’t find get / because yet we never have added any get method we will do this.

So we have to use get method in this we have to use first parameter as / means root directory for get and second anonymous function this function has 2 parameter request and respond inside this function we can call response method with send key word and output will be (“hello”) so when we hit the url we get the output as hello.

const express = require("express");

var app = express();

app.get("/", function(request, response ) # we use short of it as req , res

{

response.send("hello") } ); # here we can pass html code too

app.listen(3000, function()

{

console.log("server started on port 3000 you can access it http://localhost:3000");

});

Now we add one more path or page for route by using same method

app.get("**/contact",** function(req, res)

{

res.send(" contact Me: mahisakapal@gmail.com");

})

<http://localhost:3000/contact>

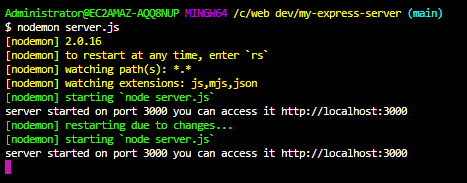
Note: Nodemon This is the utility which monitor your server and if any changes it will restart server so you don’t have to do this manually

Npm install –g nodemon

To use it: when we installed we just have start our sever.js with nodemon like below

nodemon sever.js # ex nodemon filename.js

now if we make a changes in any file we have don’t need to restart server it will done automatically.



**Type rs in console to restart the server forcefully**.

Now forward we are not making any web side we are working with web app means till now we are getting web page and processing client side but now onward we are going to process it server side.\

To start any project you need

Project dir inside that filename.js

Npm init

Npm install express

Than write a code and start your js file with node or nodemon

Till now we are sending a text to response but we can send the html file too

res.sendFile(“filename/withpath”)

but this static way but we can used dynamic way also for that we have to use \_ \_ and dirname + “index.html” you can see how it exactly looks like in code

res.sendFile( **\_\_dirname** + "/index.html");

This is the HTML code

<h1>Calculator </h1>

<form action="**/**" method="**post**">

<input type="text" placeholder="number1" name="**num1**" value="">

<input type="text" placeholder="number2" name="**num2**" value="">

<button **type="submit"** name="submit">Calculate </button>

</form>

We used **post** in our form because we pass our data to server and ask it to process where it sending **/** this local where it sending be you never give that also same to **/**

This is how your directory looks like

app.get("/", function(req, res)

{

res.sendFile( **\_\_dirname** + "/index.html");

});

Web error code

100 +++ = hold on

200 +++ = you got a success

300 +++ = security issue

400 +++ = client side issue

500 +++ = server side

We see when we use submit button it post your data to / this route but we never have a any method to handle it so we use below method to handle it

app.post("/", function(req, res)

{

res.send("We got your post data");

}

);

All this activity you can see there in your chrome developer tool > network

But wait we need that input data to process right for that we need npm package **body-parser**

**Npm install body-parser**

Once install you add this in your project

Const bodyParser = require(“body-parder”);

App.use(bodyParser.**urlencoded**({extended: ture })) # here we have other input method also like, urlencoded, json, text but this one is good for security.

We can access the input like

res.body.name\_of\_elemet # in our case we have res.body.num1 and we have res.body.num2

<input type="text" placeholder="number2" name="**num2**" value="">

const express = require("express");

const bodyParser = require("body-parser")

var app = express();

app.use(bodyParser.urlencoded({extended: true}));

app.listen(3000, function()

{

console.log("server start on port: 3000 ");

});

app.get("/", function(req, res)

{

res.sendFile( \_\_dirname + "/index.html");

});

app.post("/", function(req, res)

{

var num1 = Number(req.body.num1 ) ;

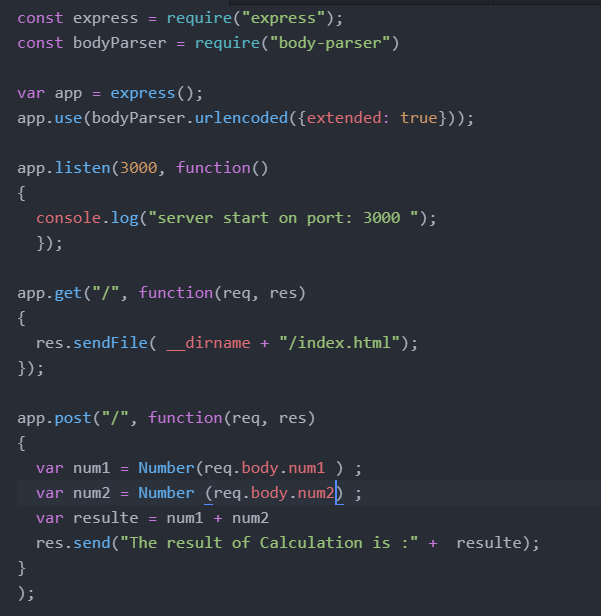
var num2 = Number (req.body.num2) ;

var resulte = num1 + num2

res.send("The result of Calculation is :" + resulte);

}

);



Now we are adding one more page and there also we are doing some task

**BMI Routing Challenge**

We're going to turn the previous BMI calculator code we wrote into a real website using what we've learnt in this module. Follow the steps below to complete the challenge:

1. Create a new file called **bmiCalculator.html** inside the Calculator folder from the last challenge

2. Add the **HTML boilerplate** and set the page title to BMI Calculator

3. Add an **HTML form**, this form will make a **post request** to our server at the route /bmicalculator. The form will have 2 inputs, weight and height with placeholder text that tell the user what they should type into which text box.

4. Add a button which says “Caculate BMI”

5. Add the get and post methods for the /bmicalculator route into the same server.js file we've been using

6. Use sendFile() to send the **bmiCalculator.html**page as a response inside the get method.

6. Add an h1 that says **BMI Calculator**

7. Inside server.js , create 2 variables, one for weight and one for height.

8. Use the BMI calculator code you wrote previously, or write some new code to calculate and send back the result as text. It should read something like "Your BMI is **n**" where n is equal to the calculated BMI based on their weight and height inputs.

BMIcalculator.html

<body>

<h1>BMI Calculate</h1>

<form action="/**bmicalculator**" method="post">

<input type="number" placeholder="enter\_hight" name="hight" value="">

<input type="number" placeholder="enter\_wight" name="wight" value="">

<button **type="submit**" name="button"> Calculate BMI</button>

</form>

</body>

Server.js file

const express = require("express");

const bodyParser = require("body-parser")

var app = express();

app.use(bodyParser.urlencoded({extended: true}));

app.listen(3000, function()

{

console.log("server start on port: 3000 ");

});

app.get("/", function(req, res)

{

res.sendFile( \_\_dirname + "/index.html");

});

**app.get("/bmicalculator", function(req, res)**

**{**

**res.sendFile( \_\_dirname + "/BMIcalculator.html");**

**});**

app.post("/", function(req, res)

{

var num1 = Number(req.body.num1 ) ;

var num2 = Number(req.body.num2) ;

var resulte = num1 + num2;

res.send("The result of Calculation is :" + resulte);

}

);

**app.post("/bmicalculator", function(req, res)**

**{**

**var hight = parseFloat (req.body.hight ) ;**

**var wight = parseFloat (req.body.wight) ;**

**var resulte = hight / (wight \* wight)**

**res.send("The result of Calculation is :" + resulte);**

**}**

**);**

**API(application programing interface) :**

It’s use to give only particular data when someone hit that particular API instead of giving full database or data access. It’s use to create software or interact with external application.

It Contained:

**Endpoint**: it’s url may be for get or post method

**Paths**: This to get more specific data we can use together with endpoint

ex: <http://endpoint/path>

**Parameters** : This to get more specific data we can use together with endpoint with **?**

Ex: [http://endpoint**?**key=values](http://endpoint?key=values)

If we have more than one query than only first query(parameter) has a ? and other can use with &

[http://endpoint**?**key=values&key1=values1&key2=values2](http://endpoint?key=values&key1=values1&key2=values2)

**Authentication** : this is to identify you

Now we are try open weather API this is not require to pay we are taking free subscription.

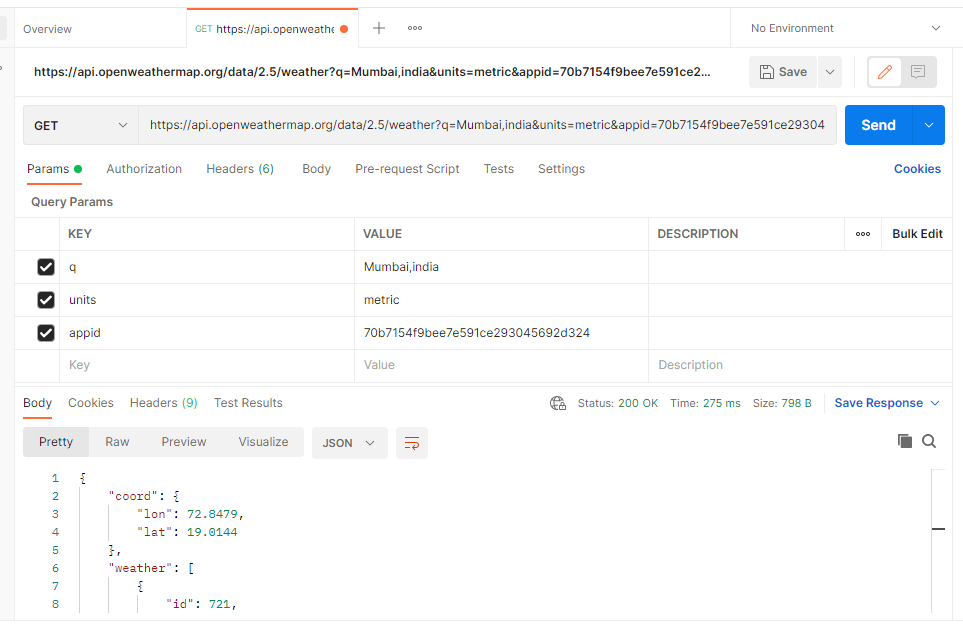
<https://home.openweathermap.org/>

https://api.openweathermap.org/data/2.5/weather?**q=Mumbai**,india&**units=metric**&appid=70b7154f9bee7e591ce293045692d324

we see all data we get is in **JSON or XML, HTML or in plain text**

like we add unite= matrices to get temp in census we can add many other and it’s hard to check our API URL in browser so we have **POSTMAN** app that is free to use

how same thing looks in in POSTMAN.



<https://chrome.google.com/webstore/detail/json-viewer-pro/eifflpmocdbdmepbjaopkkhbfmdgijcc>

This is chrome extinctions help you get nice view in chrome browser.

To call external API we have to use https node module previously we were using request module but that is deprecated for API now

HTTPS is native module so we don’t need to install using npm

In below code we have use https module and as it get response from url and pass to console

const express = require("express");

const https = require("https");

var app = express();

app.listen(3000, function()

{

console.log("server start on 3000")

});

app.get("/", function(req, res)

{

**const url = "https://api.openweathermap.org/data/2.5/weather?q=Mumbai,india&units=metric&appid=70b7154f9bee7e591ce293045692d324#";**

https.get(url, function(response)

{

**console.log(response);**

});

});

Now we are tacking that in json format.

pp.get("/", function(req, res)

{

const url = "https://api.openweathermap.org/data/2.5/weather?q=Mumbai,india&units=metric&appid=70b7154f9bee7e591ce293045692d324#";

https.get(url, function(response)

{

console.log(response.statusCode);

**response.on("data", function(data) #** taking data

{

var witherData = **JSON.parse(data); #** passing that data to json

console.log(witherData);

});

});

});

Here we use JSON.parse that make your string into json and we can make json to string by using

JSON.stringify

Till now we are getting entire json but we need particular attribute we store all json data in witherData

Now inside that we have tem so we do

witherData.main.temp # mean we get only temperature

we can use that chrome extension to get the exact path need not to see line by lie

we can use res.send one but we can use many res.write to write multiple lines on page.

res.write("<h1>This is the temperature of Mumbai : " + temp +"</h1>" );

res.write("<h2> This is the description : " + description +"</h2>" );

res.send();

HTML templet for uniform

So in case where most of html contained is same only change very few in that case rather than creating new html page every time we can create a html templet and then will change the only few thing.

Automatically in one html page

To archive this, we have online tool **ejs.com**

Full form is embedded java script templet

To use this in our project we have to use npm tool to install ejs module

Npm I ejs

Than we have use this method to use ejs in our project

app.set(‘view engine ’, ‘ejs’);

make sure app is instead before it like **const app = express();**

to use this express we have create a views folder in side this folder we have to create .ejs file you can give any name to this file and write your html code here in this file.

Which word you want to change can be replace with **< % = var %>**

We can change value of the var in our js file as per our logic

Now we have to use res.render inside that we have give name of ejs file name if we have given list.ejs tha use list only thane {key: value } in our case key is kindOfDay and value is day

list.ejs

<!DOCTYPE html>

<html lang="en" dir="ltr">

<head>

<meta charset="utf-8">

<title></title>

</head>

<body>

<h1> Today is the day **<%= kindOfDay %>** </h1>

</body>

</html>

App.js

const express = require("express");

const https = require("https");

const bodyParser = require("body-parser");

var app = express();

**app.set('view engine', 'ejs');**

app.use(bodyParser.urlencoded({extended: true}));

app.listen(3000, function()

{

console.log("server start on 3000")

}

);

var day = "";

app.get("/", function(req, res)

{

var today = new Date();

if( today === 6 || today === 0 ){

day = "weekend";

}else {

day = "weekday";

}

**res.render**("**list**", {**kindOfDay: day** })

}

);

Print the exact day

const express = require("express");

const https = require("https");

const bodyParser = require("body-parser");

var app = express();

app.set('view engine', 'ejs');

app.use(bodyParser.urlencoded({extended: true}));

app.listen(3000, function()

{

console.log("server start on 3000")

}

);

var day = "";

app.get("/", function(req, res)

{

var today = new Date();

var currentDay = today.getDay();

switch (currentDay ) {

case 0:

day = "sunday"

break;

case 1:

day = "monday"

break;

case 2:

day = "tu"

break;

case 3:

day = "wen"

break;

case 4:

day = "thus"

break;

case 5:

day = "friy"

break;

case 6:

day = "star"

break;

default:

console.log("curren day is equle to :" + currentDay);

}

res.render("list", {kindOfDay: day })

}

);

Now we are running the code inside the HTML

We have ejs tag for achieving task like control flow it’s change as per our requirement <% java script %>

Means we can use this tag to run the java script in between

And we have to add this tag for every line of java script code not like starting and ending its use for all line

In this what we have variable that is already has value from js code we are check doing control flow with the help of if else code

<body>

**<%** if (kindOfDay === "sunday" || kindOfDay === "star") {**%>**

<h1 style="color: red"> <%= kindOfDay %> Todo list </h1>

**<%** }else {**%>**

<h1 style="color: yellow"> <%= kindOfDay %> Todo list </h1>

**<%** } **%>**

</body>

Note: all our logic should be in js file and this kind of ejs code very relay when we need to do some small control flow .

Scope: we have to kind of variable local and global, local and access in same function only but global can access it from any function.

If we have variable in if else or in other code this can be access as same as global variable.

We can defined variable in 3 way

Var num = 1 ;

Const num =1 ; # it can’t be changed # it local in if else or while or any block. not like var

let num =1 ; # it local in if else or while or any block. not like var . **recommended to use**

till now we see we can give path css file and it use in our web but when we are working on express we have to tell express to get the css file from particular location.

So basically in create a one **public** folder inside that we can create a folder and keep our all browser side processing files like css folder or any src folder inside that folder and code we use in js is

app.user(express.static(public));

Note: in our html or ejs file you just have to give a path from css folder/filename.css Not a public folder

Ejs allow us to create layout it’s kind of templet

In this we are going to keep top and bottom part of page same for all page

Create header.ejs and footer.ejs created in view folder

We have copy what we want in header in header.ejs file and foot context in footer.ejs

<%- include("filenameWIthoutExtention") -%>

<%- include("header"); -%>

<%- include("footer"); -%>

<%- include("./path/of/file\_withoutExtention "); -%>

Or can be

<% include ./path/of/file\_withoutExtention %>

Or can

<%- include header.ejs -%> # with extension

You can add this 2 line for all ejs templet and you get same style for all page same header and footer

We can create our local module

Just create new js file create your function inside that file and give return and use

module.exports = functionName ;

and where you want to use this module you have to add it

const **date** = require ( \_\_dirname + “/filename.js”);

call that in other file like

**date**();

date.js

**module.exports.getDate = getDate;**

function getDate(){

let today = new Date();

//var currentDay = today.getDay();

let option = {

weekday: "long" ,

day: "numeric",

month: "long"

};

let day = today.toLocaleDateString("eg-us" , option) ;

return day;

}

**module.exports.getDay = getDay;**

function getDate(){

let today = new Date();

//var currentDay = today.getDay();

let option = {

weekday: "long"

};

let day = today.toLocaleDateString("eg-us" , option) ;

return day;

}

Note we can use export only no need to use module.export

Calling in app.js

const date = require( \_\_dirname + "/date.js");

let day = date.getDate()

let day = date.getDay()

we are not actually pushing the node module folder to git we are just having package.json

and the we only run npm install and it will download all packages which is required as per package.json file.

Note: we we use #page\_name in nav bar to redirect in html but in ejs we have to use /page name

HTML:

<li id="home"><a href="#">HOME</a></li>

<li id="about"><a href="#about ">ABOUT US</a></li>

EJS:

<li id="home"><a href="/">HOME</a></li>

<li id="about"><a href="/about">ABOUT US</a></li>

Use class =”from-group” for div when we never have a space in any to element which is right after div

<div class="mb-3 **form-group**">

<textarea class="form-control" id="exampleFormControlTextarea1" rows="3"></textarea>

</div>

**<button class="btn btn-primary btn-sm" type="submit" name="button">Submit Data</button>**

We have a loop which is iterate through each item

const array1 = ['a', 'b', 'c'];

array1.forEach(element => console.log(**element**));

console.log(**element**);

Output:

// expected output: "a"

// expected output: "b"

// expected output: "c"

posts.forEach(function (post)

{

console.log(post.titleInput)

}

)

Express routing parameter: express allow as to create a dynamic path base routing in this we ahev to pass the parameter. Reference link is below.

<https://expressjs.com/en/guide/routing.html>

in get method we have to give “/path/:parameter\_name “ , function (req, res )

we can call this route like below

req.params.parameter\_name

you can access it like

/path/parameter\_name # here parameter name can be dynamic

So we know we can ca use path but in this case what we have a case issue like upper case and like – like url to overcome form this we have a lodash concept this is utility liber

This lib can be use to make lower case string or upper case or other function you can check documents and

Npm I lodash

Add in in js

var \_ = require('lodash'); # make sure you have use \_ only for defined

to apply any method we have to use below code this ex for lowerCase

\_.lowerCase([String=’your text that need to make lower’]) # for array

\_.lowerCase(your text ); # for single string

Note this can haled your lower upper issue and also can handle space in word you have to use – to represent space like **New side** you can use **new-side** in your url

Database:

This is used to store a data we have a have sql and no sql db no sql is best for the modern use

<https://kkovacs.eu/cassandra-vs-mongodb-vs-couchdb-vs-redis/>

No sql like mongodb we can store a data in json file. In no sql databases is not good in relationship

SQL:

Play ground link : <https://sqliteonline.com/>

CREATE TABLE product (

id int NOT NULL,

name STRING NOT NULL,

price MONEY,

PRIMARY key (id)

)

INSERT INTO product VALUES( 2 , "ajay", 3.50);

SELECT \* FROM product ;

Select name, price from product,

SELECT \* from product where price = 2.5 and id = 3 # this is to get row with to check statement

To update any row we have to use update keyword and then table name set column which we want change the value and unique column with it’s value

UPDATE table\_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;

UPDATE product set price = 3.5 where id = 3

To alert table for add or delete.

ALTER TABLE table\_name  
ADD column\_name datatype;

ALTER TABLE product add item int;

DELETE FROM table\_name WHERE condition;

DELETE from product WHERE item = 20

Now we are creating a table which has foreign key which is use to create relation in 2 tables. And the sntax is

**Foreign key (current\_tabel\_column) REFERENCES other\_table\_name (it’s\_column\_name)**

CREATE TABLE orders (

id int not NULL,

order\_number int,

customer\_id INT,

product\_id INT,

PRIMARY KEY (id),

FOREIGN key (product\_id) REFERENCES product(id)

);

Inner join in sql.

SELECT column\_name(s)  
FROM table1  
INNER JOIN table2ON table1.column\_name = table2.column\_name;

SELECT Orders.OrderID, Customers.CustomerName  
FROM Orders  
INNER JOIN Customers ON Orders.CustomerID = Customers.CustomerID;

Installing Mongodb on MAC:

<https://blog.londonappbrewery.com/how-to-download-install-mongodb-on-mac-2895ccd2b5c1>

Installing Mongodb on windows:

<https://medium.com/@LondonAppBrewery/how-to-download-install-mongodb-on-windows-4ee4b3493514>

Installing Mongodb on Linux:

<https://www.mongodb.com/docs/manual/administration/install-on-linux/>

Mongo DB documents:

<https://www.mongodb.com/docs/manual/>

To start with mongo, we have to start **mongod** this is server for mongo db and it use 27017 port

Keep this terminal running and start another terminal and type mongo it will start your terminal now here we can run our cmd.

No your on mongo console you can use: help to see all working cmd

Show dbs # to see all database

Use db\_name # to use or select the database if you give name which is not there it will create a new database.

db # to see which db your currently using

inside we have to create a collection and inside that we have to put our data there is 2 way to achieve it.

db.collection.insertOne() New in version 3.2

db.collection.insertMany() New in version 3.2

db.user.insertOne ( { # here use is one collection(table in sql)

name: “Amar”,

age: 26,

status: “pending”

} )

# in this we are passing the the name of collection if it’s not present it will create, and in ( { key: value} )

Show collection # to see all the present collection in your database

This is the example of many:

|  |
| --- |
| db.collection.insertMany( |
| [ <document 1> , <document 2>, ... ], |
| { |
| writeConcern: <document>, |
| ordered: <boolean> |
| } |
| ) |
| db.user.find() # it will give all documents list which is available in collection. |

This was simple find but we ca use many query select

In find we have a to parameter one is query, projection means what filed we want to return if it’s 1 means true show that filed and 0 means false means don’t show this filed

db.product.find( { \_id: 1}, {name: 1})



db.product.find( {\_id: {$lt: 2}})

db.product.find( { name: "pencil"})

db.inventory.find( { status: **"D"** } )

db.inventory.find( { status: { $in: [ **"A"**, **"D"** ] } } )

The following example retrieves all documents from the inventory collection where status equals either "A" or "D":

db.inventory.find( { status: **"A"**, qty: { $lt: 30 } } )

The following example retrieves all documents in the inventory collection where the status equals "A" **and** qty is less than ([$lt](https://www.mongodb.com/docs/manual/reference/operator/query/lt/#mongodb-query-op.-lt)) 30:

db.inventory.find( { $or: [ { status: **"A"** }, { qty: { $lt: 30 } } ] } )

The following example retrieves all documents in the collection where the status equals "A" **or** qty is less than ([$lt](https://www.mongodb.com/docs/manual/reference/operator/query/lt/#mongodb-query-op.-lt)) 30:

|  |
| --- |
| db.inventory.find( { |
| status: **"A"**, |
| $or: [ { qty: { $lt: 30 } }, { item: /^p/ } ] |
| } ) |

In the following example, the compound query document selects all documents in the collection where the status equals "A" **and** either qty is less than ([$lt](https://www.mongodb.com/docs/manual/reference/operator/query/lt/#mongodb-query-op.-lt)) 30 or item starts with the character p:

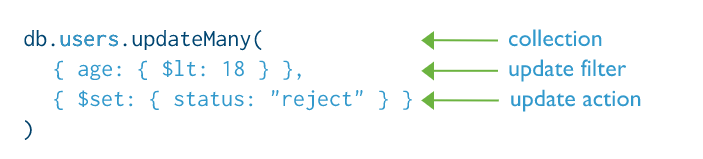
We can use many oprater like $lt or $gt or $eq check out the documents.

Updating recoding

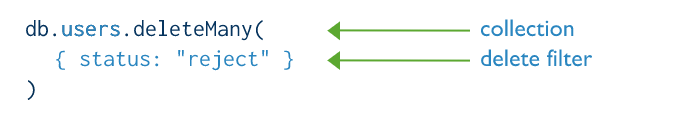
db.product.updateOne({\_id: 1}, {$set:{stock: 12}})

# $set to add or update document filed.

* [db.collection.updateOne()](https://www.mongodb.com/docs/manual/reference/method/db.collection.updateOne/#mongodb-method-db.collection.updateOne) *New in version 3.2*
* [db.collection.updateMany()](https://www.mongodb.com/docs/manual/reference/method/db.collection.updateMany/#mongodb-method-db.collection.updateMany) *New in version 3.2*
* [db.collection.replaceOne()](https://www.mongodb.com/docs/manual/reference/method/db.collection.replaceOne/#mongodb-method-db.collection.replaceOne) *New in version 3.2*



* [db.collection.deleteOne()](https://www.mongodb.com/docs/manual/reference/method/db.collection.deleteOne/#mongodb-method-db.collection.deleteOne) *New in version 3.2*
* [db.collection.deleteMany()](https://www.mongodb.com/docs/manual/reference/method/db.collection.deleteMany/#mongodb-method-db.collection.deleteMany) *New in version 3.2*



db.product.deleteOne({\_id: 2})

Relationship:

Here in none sql db we make a relationship by creating one more embedded documents inside it we can create array also for multiple documents.

db.user.insert ( {

\_id: 2,

name: "book" ,

price: 12,

review: [

{

AuthoreName: "test",

Rating: 5,

Review: "super product"

},

{

AuthoreName: "foo",

Rating: 4,

Review: "good product"

}

]

} )

This kind of relation call as one to many

Now we are using mongo db in our node application for this we need drivers as we know for java application to make connection to db we have jdbc driver we same concept here.

We have old traditional drivers (native) way and **mongoose** this new way of drivers

Native: <https://www.mongodb.com/docs/drivers/> The URL of drivers.

For node here is quick start link to get driver in your project.

<https://www.mongodb.com/docs/drivers/node/current/quick-start/>

or

<https://mongodb.github.io/node-mongodb-native/3.1/quick-start/quick-start/>

once this mongo, and assert npm package is install we have to use mongodb and assert in our project by using require key word

const MongoClient = require('mongodb').MongoClient;

const assert = require('assert');

const url = 'mongodb://localhost:27017';

const dbName= 'myproject';

const client = new MongoClient(url);

client.connect(function(err){

assert.equal(null, err);

console.log("your connected" );

const db = client.db(dbName);

client.close();

});

If you have a mongod server running and your both npm package mongo, and assert are install then no you will get connection success in your terminal.

const MongoClient = require('mongodb').MongoClient;

const assert = require('assert');

const url = 'mongodb://localhost:27017';

const dbName= 'myproject';

const client = new MongoClient(url);

client.connect(function(err){

assert.equal(null, err);

console.log("your connected" );

const db = client.db(dbName);

insertDocuments(db, function(){ # we now adding new data than closing connection

client.close();

})

});

const insertDocuments = function(db, callback) {

// Get the documents collection

const collection = db.collection('fruits');

// Insert some documents

collection.insertMany([

{name : "apply",

price: 12

},

{name : "orange",

price: 10

},

{name : "mango",

price: 20

}

], function(err, result) {

assert.equal(err, null); # checking error

console.log("Inserted 3 documents into the collection");

callback(result);

});

}

If You Have Forgotten to Quit the Mongod Server

You can use CTRL + C in your Terminal to shut down your mongod connection.

If you have closed down Terminal or Hyper and forgot to close down your mongod connection, you might get an error that says:

2018-11-04T16:17:53.473+1300 E STORAGE [initandlisten] Failed to set up listener: SocketException: Address already in use

2018-11-04T16:17:53.474+1300 I CONTROL [initandlisten] now exiting

2018-11-04T16:17:53.474+1300 I CONTROL [initandlisten] shutting down with code:48

In this case, you'll have to follow these steps to manually shut down the mongod process:

1. Open up a fresh Hyper Terminal tab

2. Paste the command below into Hyper:

sudo pkill -f mongod

3. Now enter the password that you use to log on to the Mac.

4. Open a new Hyper terminal, you should now be able to run the mongod command again.

**Note:**

**This is most tidies work with native mongo driver but when we use mongoose it more essay to code and will look that now**.

Mongoose:

This is the url of mongo: <https://mongoosejs.com/>

**const** mongoose = require('mongoose');

mongoose.**connect**('mongodb://localhost:27017/test');

**const** **Cat** = mongoose.**model**('Cat', { name: **String** });

**const** kitty = **new** **Cat**({ name: 'Zildjian' });

kitty.**save**().**then**(() => console.**log**('meow'));

This is the code we have to use.

Below is the code to replace old data entry code

const mongoose = require('mongoose');

mongoose.connect("mongodb://localhost:27017/fruitsDB", {useNewUrlParser: true});

const fruitSchema = new mongoose.Schema( # here we defining schema

{

name: String,

rating: Number,

review: String

});

const **Fruit** = mongoose.model("Fruit" , fruitSchema); # here we use Fruit but in mongo #you can find it as Fruits for collection

const fruit = new **Fruit** ({

name: "Apple",

rating: 3,

review: "good"

})

fruit.save(); # and saving data

Note: when we give collection Fruit in data it’s show Fruits when we give person it make it people non to purl form.

To add many we first have to create const for every fruit and in insertMany we have to pass array inside array pass all const name and then write an if else code to check it run successfully or not

const **kiwi** = new Fruit ({

name: "kiwi",

rating: 3,

review: "ok"

})

const **orange** = new Fruit ({

name: "orange",

rating: 3,

review: "ok"

})

const **banana** = new Fruit ({

name: "banana",

rating: 3,

review: "ok"

})

Fruit.**insertMany**([**kiwi, orange, banana**], function(err){

if(err){

console.log(err);

}else{

console.log("added successfully");

}

});

TO find a all items from fruits collection.

**Fruit**.find(function(err, fruits) {

if (err){

console.log(err);

}else {

console.log(fruits);

}

})

We can use mongoose.connection.close(); #to close connection

As we know how we add the not null in sql we can use

**const** schema = **new** **Schema**({

name: {

type: **String**,

required: true

}

});

Or like this

**const** breakfastSchema = **new** **Schema**({

eggs: {

type: **Number**,

min: [6, 'your msg if condition is false'],

max: 12

},

bacon: {

type: **Number**,

required: [true, 'Why no bacon?']

},

drink: {

type: **String**,

enum: ['Coffee', 'Tea'],

required: **function**() {

**return** this.bacon > 3;

}

}

});

In older way we are just giving the value to our key like name: “abc” but when we use a validator we should use object {} inside this we can can add other property like type and required

For more information go thought the link: <https://mongoosejs.com/docs/validation.html>

To update one recode:

Fruit.updateOne({ condition}, {what to modify } function(err){

if (err) {

console.log(err);

}else {

console.log("successfully");

}

});

To delete one, recode.

fruit.deleteOne({**name: "Apple"**}, function(err){

if (err){

console.log(err);

}else {

console.log("recode deleted");

}

});

For deleting many recodes:

Fruit.deleteMany({name: "Apple"}, function(err){

if (err){

console.log(err);

}else {

console.log("recode deleted");

}

});

Consider we have one collection for people and one for fruits and we want to add fruits for people collection in this case we don’t have to add all schema in people collection instead we can call fruit schema in person collection.

Than in our people where we creating recode we just have to add this fruit name and it will create relations.

How to run find cmd in app.js. with query name = test if you keep query {} blank this will find all

Item.find({ name: "test"}, function(err, Items) {

if (err){

console.log(err);

}else {

mongoose.connection.close();

console.log(Items);

}

});

As we know for submit button we have action to submit it and for but for get input from check box we should use to attribute one onChange and other is value

onChange="this.form.submit()

value= <%= item.\_id %>

so the code of check box look like this

<input type="checkbox" name="test" value= <%= item.\_id %> onChange="this.form.submit()">

And we have create post method check that value from checkbox which has name = test

app.post("/delete", function(req,res){

console.log(req.body.test);

});

Mongo atlas database:

We have go on mongo console and create free database select any cloud I have selected aws than we have to create a cluster select free tier one and then we have create a use where we give that user atlas admin access you can give read and write only also Note: don’t use @ in password it create a problem

To test connection select 4.4 or below version and then check for the mongo shell or mongosh

To connect your application as per language select drive and use in your code just update password.

RESTfull API:

**Re**presentation **s**tate **t**ransfer:

We have below rest http verb

get: read

post : create

put : update here we update hole recode

patch : update here we updating only required part

delete : delete

robomongo is the mongo database we use for graphical database for our API.

Download and open it when we click on connect it ask you where to connect here we use localhost for now download from here

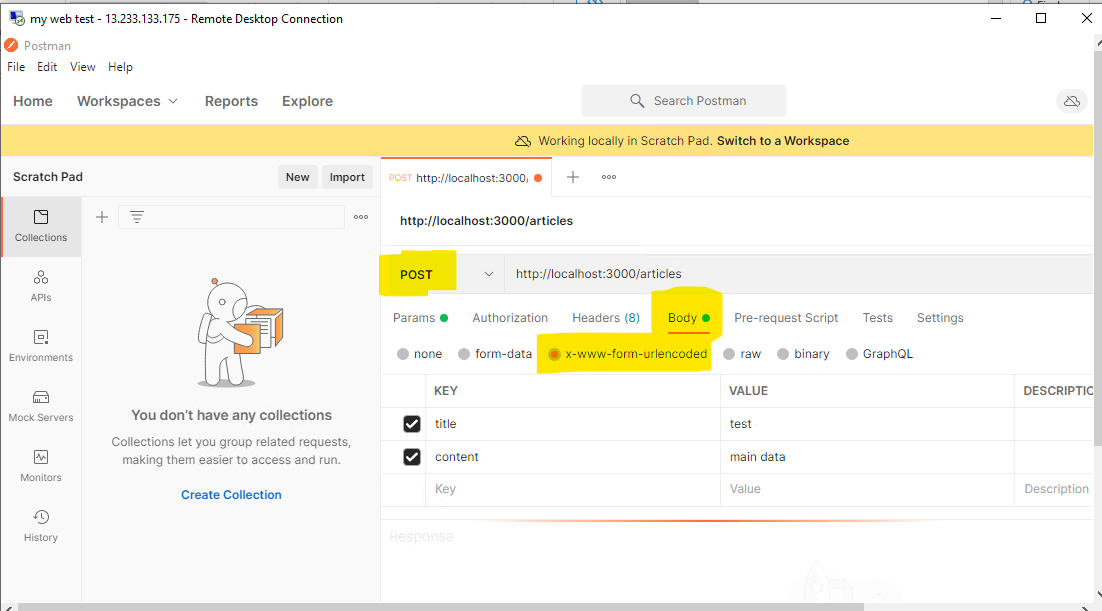
https://blog.robomongo.org/studio3t-free/

This is tool for gui view of database.

We use post to create a recode but in api we never have a GUI so we use post man

But we can use url to post it and we have title filed and contain filed and we pass our value /articles for post

http://localhost:3000/articles?**title=testFromPost**&**content=Any data from you**

****

**Now we have requirement to delete all data here we are using delete method inside that we delete all document from article collection.**

app.delete("/articles", function(res,req){

Article.deleteMany({}, function(err){

if(!err){

res.send("deleted successfully");

}else {

res.send(err);

}

});

});

In our current code we have a use to many method for same route so express give good solution for this that is shown below

app.route('/book')

.get((req, res) {

res.send('Get a random book')

})

.post((req, res) {

res.send('Add a book')

})

.put((req, res) {

res.send('Update the book')

})

In bow code for /book rout we write code for get, post, put

Originally it look like below and then we add code and for good look we re formatted

app.route('/book').get().post().delete();

now we are looking for how to get specific recode.

app.route("/articles/:articalTitle")

.get(function(req, res){

Article.findOne( {title: req.params.articalTitle }, function(err, foundArticle){

if (foundArticle) {

res.send(foundArticle);

} else{

res.send("we are unable to find the artical");

}

} );

})

Authentication and security:

All level of authentication code is available her on <https://github.com/londonappbrewery/Authentication-Secrets>

We have to create user in db and if it’s added then we can render user to secret page

mongoose.connect("mongodb://localhost:27017/userDB" ) // connection created & db also

const userSchema = { // collection schema created

email: String,

password: String

}

const User = new mongoose.model("User", userSchema); // now collection created using schema

const newUser = new User({

email: req.body.username,

password: req.body.password

}) // assing data in to field taking input from text html

newUser.save(function(err){

if (err){

console.log(err);

}else {

res.render("secrets")

}

// saving data to if any error we show on console is there is no error we render to secrets

Now are writing a code for login in which we are checking user input by which we get from res.body.email and res.body.password

app.post("/login", function(req, res){

const username = req.body.username ;

const password = req.body.password

User.findOne({emai: username }, function(err, findUser){

if(err){

console.log(err);

}else{

if(findUser){

if(findUser.password === password){

res.render("secrets")

}else {

console.log("wrong username and password");

}

}

}

})

});

Level 2 authentication:

Now we are going to encrypt password there are many encryption type here we first see caesar cipher in which we shift alphabets to number of position.

But for mongo we have to use this npm package to implement it

Npm I mongoose-encryption

This use AES-256 encryption

Once it installby npm we have use it our app by require key word to import in in our app.js

Thane we have to change schema and it will look like below basically we use mongoose.Schema

const userSchema = new mongoose.Schema({

email: String,

password: String

})

After this we create a one secret and will use that for encryption

**const secret = "Thisismylongsecret";**

than we use userSchema.plugin than we use encrypt then we will use assign that to our secret

userSchema.plugen(encrypt, {secret: secret});

in above case we encrypt al our database or collection but we want to encrypt only password field

for this we use below code there right before } we have to add encryptedFields: [‘field\_name’] as below

userSchema.plugin(encrypt, {secret: secret, **encrypatedFields: ['password']** } );

in this case we just added password field but we can use many field by use , comma separation.

That’s it we don’t have change anything else we just change schema and we use plugin to encrypt the filed and we are done

For application requisite it will automatically handle encrypt and decrypt operation.

**If you face nay issue like Error: Authentication code missing**

**Try to delete the data base and re run the application**

In this we have change to get hacked if anybody got access of our secret the he can decrypt database and can see password.

**const secret = "Thisismylongsecret"; # this we defined in our code**

**level 3**

now we are making that secret as **environment variable**

now we need to install .ENV package from by npm

npm I .ENV

and we just require it here we never use const

require("dotenv").config();

first we have to create .env file this is the hidden file.

Vi .env

And the we have to add our data in our case it secret in name= value format

**SECRET=Thisismylongsecret**

**We can access it but we need dotenv in our app.js the we can access it like below**

**process.env.SECRET # here I have SECRET in .env file you can create any other and call it**

**now we have to modify our secret which we are calling in** userSchema.plugin now updated code is below

userSchema.plugin(encrypt, {secret: process.env.SECRET , encryptedFields: ["password"] });

we have gitignore file temple for all kind of project we can find the on below link

<https://github.com/github/gitignore>

level 3 of security:

hashing: you can hash your peace of data but there in almost no change to return it in normal text

how it’s working : when we register we save user password in hash and when user try to login we convert his login password in hash and campier against database if both are match then it can login

<https://cryptii.com/> # here we can find lots of type encryption or we can decrypt it also.

Now we are using md5 for hashing

Add in require code remove the userSchema.plugin remove mongoose-encryption from require

And when we are saving that data in database in register page we have to convert it md5

Const md5 = require (‘md5’)

app.post("/register", function(req, res){

const newUser = new User({

email: req.body.username,

password: **md5**(req.body.password)

})

**And save thing we have to do for login process where we passing password**

app.post("/login", function(req, res){

const username = req.body.username ;

const password = **md5**(req.body.password)

User.findOne({emai: username }, function(err, findUser){

if(err){

console.log(err);

}else{

if(findUser){

if(findUser.password === password){

res.render("secrets")

}else {

console.log("wrong username and password");

res.redirect("login");

}

} } })

});

This is the side which give list of application that send you your password in pain text when you forget

<https://plaintextoffenders.com/>

<https://haveibeenpwned.com/> to see your password is use or hacked by someone

but we see there is md5 hash table and most people use good compute to use that table and hack you in seconds.

Level 4:

Hashing and salting: in which we add our own peace of data (random character) in data base. and make difficult to hack

And we see md5 is process more in good compute so in industry we use bcrypt hashing

And also we can add salt rounds in which we add random character and then again from that hash we add new salt and create new hash now it will called as 2 rounds of slat.

Install bcrypt by npm

Npm I bcrypt

const bcrypt = require('bcrypt'); # add in app.js

const **saltsRounds** = 10 ; # you can select more or less number of salts round

now things are little deferent

we have to use this code for hashing and salting or password

bcrypt.hash(yourpasswowrd, salt, function(err, hash) {

// Store hash in your password DB.

});

Our code will look like this

app.post("/register", function(req, res){

bcrypt.**hash**(**req.body.password**, saltsRounds, function(err, hash) {

const newUser = new User({

email: req.body.username,

password: **hash** # we are hash which we save by using in above method

})

newUser.save(function(err){

if (err){

console.log(err);

}else {

res.render("secrets")

}

});

});

});

To login we have to use below code.

bcrypt.compare(myPlaintextPassword, hash, function(err, result) {

// result == true

});

And now where we checking user input is same what we save in database is completely change.

app.post("/login", function(req, res){

const username = req.body.username ;

const password = (req.body.password);

User.findOne({emai: username }, function(err, findUser){

if(err){

console.log(err);

}else{

if(findUser){

bcrypt.compare(**password**, **findUser.password**, function(err, result) {

//here taking password whcih we got from user and comparing whic user password(findUser.password) whcih is in database

if (result === true){

res.render("secrets")

}

});

} } } } )

});

Note: if you face any issue after applying any encryption you have to drop your collection.

Level 5 :

Cookies and session: cookies are used to save user activity data in browser and this cookie can be accessible from other web sides also.

We basically use passport for cookies and session.

We have to install passport packages by npm

npm I passport passport-local passport-local-mongoose express-session

NOT(express-sessions)

Now we are have to remove a bcrypt and our registration method and login method

This 3 packages we need to use in our app.js

const session = require('express-session');

const passport = require('passport');

const passportLocalMongoose = require('passport-local-mongoose');

we have to use below code before we create a db connection and right after we use app.use

app.use(session({

secret: "This is my secret",

resave: false,

saveUninitialized: false

});

app.use(passport.initialize());

app.use(passport.session());

where we have a userSchema we have to use this code below

userSchema.plugin(passportLocalMongoose);

use this code right after we create mongoose model

passport.use(User.createStrategy());

passport.serializeUser(User.serializeUser());

passport.deserializeUser(User.deserializeUser());

and we have a passport mongoose package that handle a creating new use in mongo db

with this method we have option to use login register method

Level 6:

OAuth: open stander authorization, in this we are using popular app to authorize us like gmail or facebook. That also help to know how is how is my friend from facebook are using this application.

The advantage of this is we are get authenticate by our facebook or gmail

In this we can have granule data access for such application like FB and Gmail

It can have ReadOnly or read and write access

And user can DE authorize or revoke a access

Once we login to facbook through our application we go a authentications code or access token

Lets start implementing it

In this we are using

Passport-google-oauth20 to packages Note this todays latest version it may be change when your implementing it.

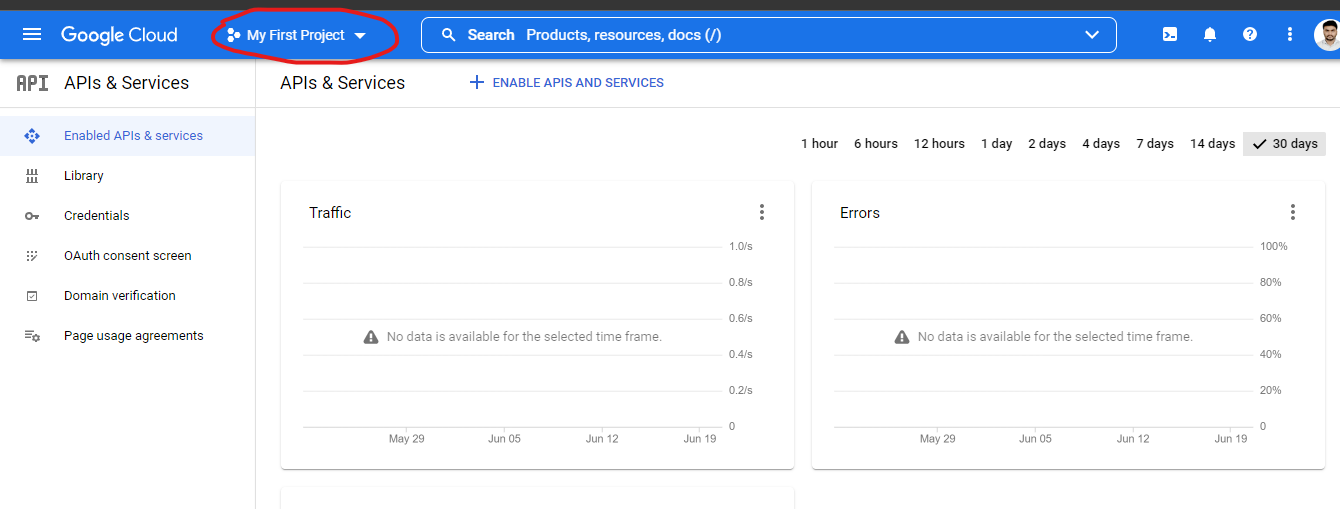
<http://www.passportjs.org/packages/passport-google-oauth20/>

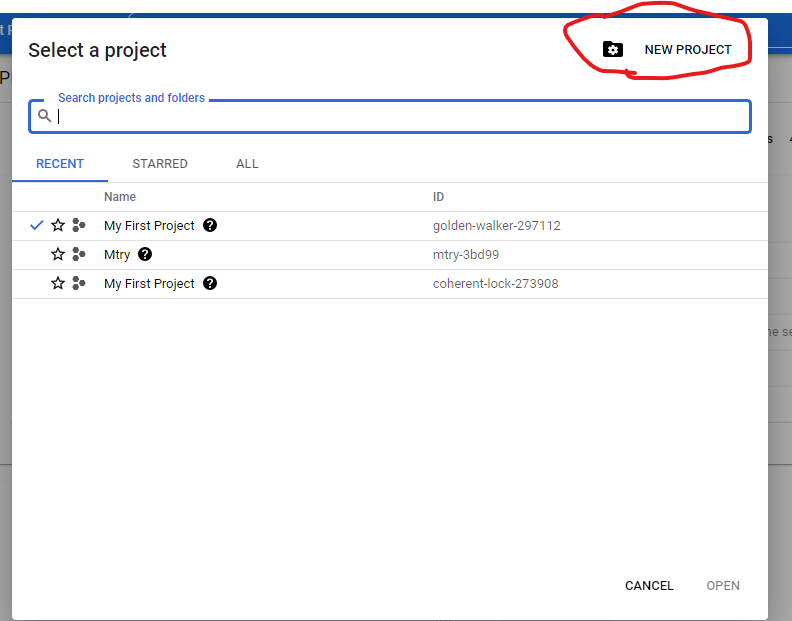
npm install passport-google-oauth20

install it by above cmd

the we have to create new project on below link

<https://console.cloud.google.com/apis/dashboard?pli=1&project=mtry-3bd99>

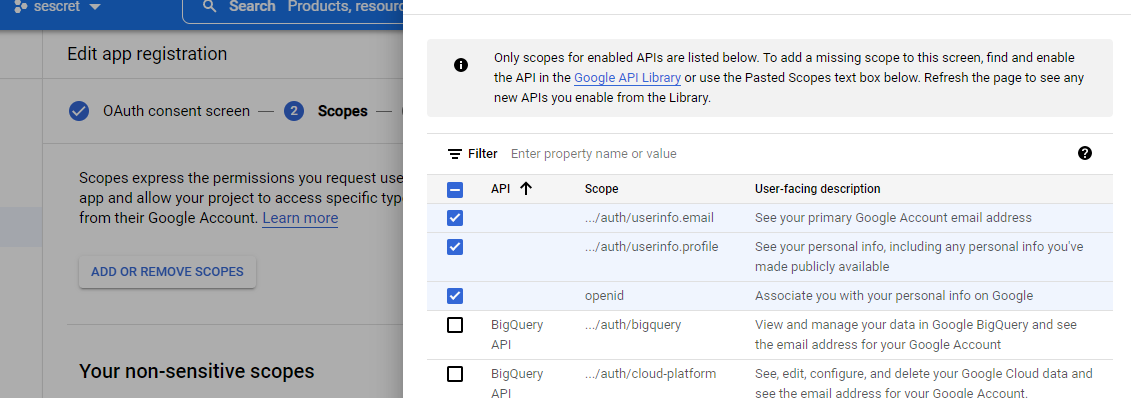




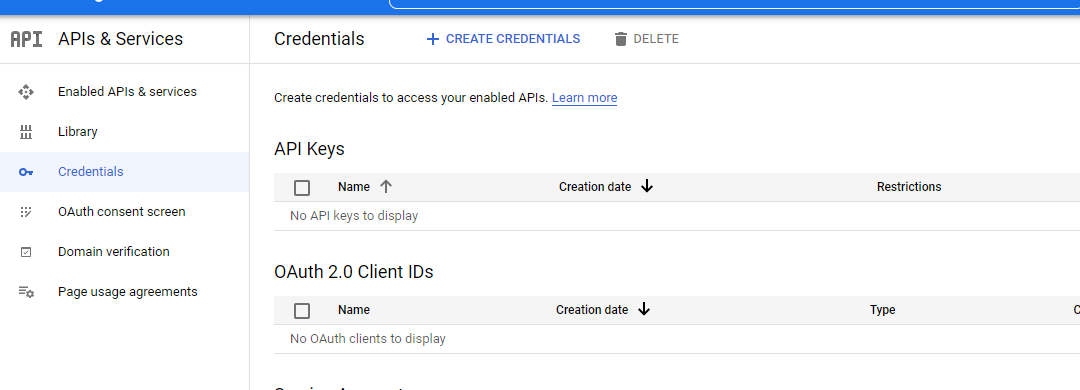
Then we gave name to our project as secret and save it.

Now to go to credential and click on OAuth consent screen > give your app name here email id and logo

Once that is save you will go to this scree to add scope for now we are just adding this 3

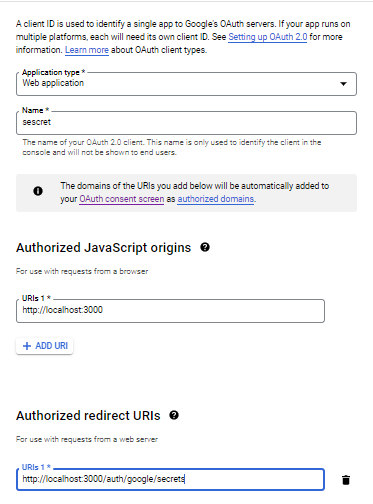


Once create click on create credentials



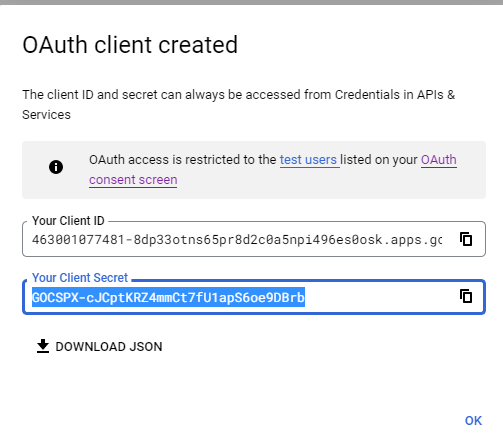
Select Create OAuth client ID

Select web app then > give application name > and give your application url > the url where we want user should be redirected to when we authenticate it.



Now you will get client id and secrets

After this we have to enable google plus API google +



Go if you save this .env file

Till now we have done what we need to do from google side now we are moving to write code

First we have to add the require part

**var** GoogleStrategy = **require**('passport-google-oauth20').Strategy;

we have to use below code right after serialize and de serialize user after this passport.deserializeUser

passport.use(**new** GoogleStrategy({

clientID: GOOGLE\_CLIENT\_ID,

clientSecret: GOOGLE\_CLIENT\_SECRET,

callbackURL: "http://www.example.com/auth/google/callback"

},

**function**(accessToken, refreshToken, profile, cb) {

User.findOrCreate({ googleId: profile.id }, **function** (err, user) {

**return** cb(err, user);

});

}

));

Change the place holder it same as variable value what we save in .evn we we have to it here for client id or slienetSecrete and call back URL the we take from .env file by process.env.CLIENT\_ID

And call back url what we set in google project.

That is <http://localhost:3000/auth/google/secrets>

If there is any issue go you git hub and see what is the issue for this <https://github.com/jaredhanson/passport-google-oauth2>

Install this package to handle User.findOrCreate method

npm install mongoose-findorcreate

then use require code in app.js

const findOrCreate = require('mongoose-findorcreate');

then we have to add plugin for findOrCreate

userSchema.plugin(findOrCreate);

now we have ejs code sign up button which redirect as **/auth/google** so we have to write get method for this. path

app.get("/**auth/google**", passport.authenticate('google', {scope: ["profile"]})

);

This methods are google method so we have write it in same way not like JS same can fine on his page

<http://www.passportjs.org/packages/passport-google-oauth20/>

In above code we are using passport.authenticate method to get from google and we retrieve only profile.

And once we login by google we re directed to **/auth/google/secret that’s what we give here in google project. Now we have to create get method by using below code**

app.get("/auth/google/secret",

passport.authenticate("google", { failureRedirect: "/login" }),

function(req, res) {

// Successful authentication, redirect home.

res.redirect('/secrets');

});

React:

Good framework for do good view web side like amazon, or e commerce side

Basically we combined html, css, and css that we can use for particular view we create many of them

This function we see in facebook when we load only bottom page updated rest all is as it is

We are using <https://codesandbox.io/>

This is the good tool for practice

JSX is the core component of react.

Now we are writing in index.js

We search a dependency and installed it the we have to add you require code in index.js

The new way of doing this is

import { StrictMode } from "react";

import { createRoot } from "react-dom/client";

so our code look like below

import React from "react";

import ReactDOM from "react-dom";

ReactDOM.render(<h1>Hello </h1>, document.getElementById("root"));

So the code format is look like below

ReactDOM.render(what to show, where\_to\_show, callback\_function));

Render method take only one html tag so when we have many of the html tag to pass we should use div and inside this we have to add the as many tag we have to use

ReactDOM.render(

<div>

<h1>Hello </h1>

<p> This is new text</p>

</div>,

document.getElementById("root"));

we are rendering html in js by help of JSX but this is not all we have many more feature like we can add js in html and that html in js haha … 😂🤣

to call or refer the java script code we just have to use {variable\_name }

var **name** = "Amardip"

ReactDOM.render(

<h1>{**name**}'s Fevourite Foods</h1>,

document.getElementById("root")

);

ReactDOM.render(

<div>

<h1>Hello {name}</h1>

<p> This is your lucky number {Math.floor(Math.random() \* 10)} </p>

</div>,

document.getElementById("root")

);

This is what it look like a when we run js code inside our html.

Note we can’t write statement like if else or while code in this we only can write expression

Statement: here we ask computer to do some task or validate, check.

Expression: we just run code and get single output from that.

Good refer link ofr statement vs expression <https://www.youtube.com/watch?v=WVyCrI1cHi8&list=PL-xu4i_QDSxcoDNeh8rx5-pHCCTOg0XsI>

Css in react

In in normal case we create code in css file and in html we have to add class and what give name in css we have use here.

But when we using react with jsx we have to use className this is way we have to use in js

Css

.heading {

color: blue;

}

Index.js

<h1 className="heading"> This is my fevourete food </h1>

Index.html

<script src="../src/index.js" type="text/JSX"></script>

For inline style we have a word style but this is not HTML code this is a js so we have to give values in different order

Style={{ color: “red”}}

<h1 className="heading" style={{color: "red"}}>

This is my fevourete food {name}

</h1>

Note: here for passing style we have use {{ }}

And what we use in html or in css like font-size now we have to use in fontSize means camel case

We pass value like 2px or read that we have to use as string “read” , “2px”

In this we have use variable to get a style and we are using that for style

const customStyle = {

color: "green ",

fontSize: "30px",

border: "1px soild black"

};

ReactDOM.render(

<div>

<h1 className="heading" style={customStyle}>

This is my fevourete food {name}

</h1>

</div>,

document.getElementById("root")

);

React component:

In this we are creating function for heading in this function name should be in caps

function Heading(){

return <h1> Hello from amardip</h1>

}

And in our render code we are using function name as HTML

<function\_name> </function\_name> in our case it is <Heading> </Heading>

We know if we are not passing anything in html tag we have to use one tag only and we have to close this in same function.

<Heading/> # you see in normal html tag we use small word but when we use js function html tag we have to use frits latter caps.

Now we are moving this heading code in other jsx file and we can import it in our index.js

And we have to import react in our new jsx file

import React from "react";

function Heading() {

return <h1> Hello from amardip</h1>;

}

export default Heading;

We are exporting it to use in our index.js file by using import and path of our jsx file

import Heading from "./Heading.jsx";

it’s ok when we have one or two tag but with larger tag we have to use App.jsx

this what our index.jsx look like

import React from "react";

import ReactDOM from "react-dom";

import App from "./App";

ReactDOM.render(<App />, document.getElementById("root"));

We are just calling app.jsx in our main index.js file and inside app.jsx we are calling our other jsx module like list heading

Below is code of app.jsx

import React from "react";

import Heading from "./Heading.jsx";

import List from "./List.jsx";

function App() {

return <div>

<Heading /> <List />

</div>

}

export default App;

we when we have defended only single function in that case we can use export default but when we have more than 1 function in the in export we can defind as

export {function\_1, function\_2} ;

and we can import in our app.jsx or index.js like below

import App , {function\_1, function\_2} from “./math.js”;

in this we are importing default App.js and also our function\_1 and function\_2

we can import everything as any name but for now we are importing it as App

import \* as App from “./math.js”;

in this case we have to code in our app.jsx or index.js like

App.default, App.function\_1 and App.function\_2

Note: when we use require it’s a node js and ES6 we use import export

Local Environment:

Make sure you have **node** install that we did while creating our first node app

Vs code installed

Now we have to install **subline babel by josh ping** extension we have to install

Now we are creating our first app open terminal and run below cmd

npx create-react-app my-app

cd my-app

npm start

first time to install all dependency it will take a time.

Now our app contain index,js with below code

import React from 'react';

import ReactDOM from 'react-dom';

ReactDOM.render(<h1> Hello world!</h1>, document.getElementById('root'));

And index.html

<!DOCTYPE html>

<html lang="en">

  <head>

    <title>React App</title>

  </head>

  <body>

    <script src="./src/index.js" type="text/jsx"></script>

    <div id="root"></div>

  </body>

</html>

Note: you can only apply scc style to inside function html code not to index.js

Now we are looking in props

In this we we are adding value in index.js under card where we rendering and we can access that in our function like {props.name\_of\_element}

Our Card.jsx

function Card(porps) {

return (

<div className="my-style">

<h2>{porps.name}</h2>

<img src={porps.img} alt="avatar\_img" />

<p>{porps.tel}</p>

<p>{porps.email}</p>

</div>

);

}

This is code for index.js

ReactDOM.render(

<div>

<h1>My Contacts</h1>

<Card

name="Beyonce"

img="https://blackhistorywall.files.wordpress.com/2010/02/picture-device-independent-bitmap-119.jpg"

tel="+72817821721368"

email="test@gmail.com"

/>

<Card

name="Beyonce"

img="httasdkjsadj.jpg"

tel="+213573217589"

email="testadjadsjh@gmail.com"

/>

</div>,

document.getElementById("root")

);

In this we are passing our card detail over and over again by hand we can pass other js file

The format should be like one object inside that we should have array

Const contacts =[

{

Key: values,

Key: values

},

{

Key: values,

Key: values

}

]

Inside this file we have to export it like export default contacts

And in our app.jsx we are importing

import Contacts from "../contacts.js";

and we are access that array like that

<Card

name={Contacts[0].name}

tel={Contacts[0].phone}

img={Contacts[0].imgURL}

email={Contacts[0].email}

/>

<Card

name={Contacts[1].name}

tel={Contacts[1].phone}

img={Contacts[1].imgURL}

email={Contacts[1].email}

/>

We have a react dev tool that help us to see react component view for this we have to install the react devops tool extension in chrome.

We see in above case we are iterating one by one from contact array for this to make a easy we have a mapping function.

In div we are creating one function called as

{array\_name.map()} and inside this map function we can call other function when we declare that function we have to pass the array that that we have imported in singular for and write a code that return card and we give name then {array\_name.tag\_of\_array} so our code look like below

We are changing the in app.js and all together

Note: if you get error .map() function is not defined check your data js in our conatacts.js we have to export it.

export default contacts;

import React from "react";

import Card from "./Card";

import contacts from "../contacts"; // this is js data file

function createCard(any\_name) {

return (

<Card

key={ any\_name.id} // one filed should be unique

name={ any\_name.name}

img={ any\_name.imgURL}

tel={ any\_name.phone}

email={ any\_name.email}

/>

);

}

function App() {

return (

<div>

<h1 className="heading">My Contacts</h1>

{contacts.map(createCard)} // this loop for all createCard

</div>

);

}

export default App;

here we use .map function to get all element but we have more similar option filter, reduce, find, FindIndex

so how or map function work we

var numbers = [3, 56, 2, 48, 5];

function dabule(x){

return x \* 2 ;

}

numbers.map(dabule); // it will dabule all number

In above case we have array of numbers and the we use .map function on that

Then we create one function in which we pass input from array one by one and the we are running our function on that and giving out put to map function

*//Filter - Create a new array by keeping the items that return true.*

c

Basically add numbers of array

*//Reduce - Accumulate a value by doing something to each item in an array.*

numbers.reduce( function (Accumulate, currentNum){

Accumulate + currentNum ;

});

*//Find - find the first item that matches from an array.*

numbers.find(function (num){

return num > 10

});

*//FindIndex - find the index of the first item that matches.*

numbers.findIndex(function (num){

return num > 10

});

Arrow:

We see we can use the function inside our function and more on that we can have a anonyms function like below but in arrow method we can remove function keyword and use =>

numbers.map(function (num){

return num > 10

});

With arrow method implementation our code look like below

numbers.map( (num) => {

return num > 10;

});

Make sure don’t give a space in = and >

we just have to give parameter in (num) and if you have more then one paramert then use ( ) if you have one then even not need to use it u

numbers.map( num => {

return num > 10;

});

And more one this we can remove return keyword and also { } and now same function look like below

numbers.map( (num) => num > 10 );

I have change our app.js with arrow method

import React from "react";

import Entry from "./Entry";

import emojipedia from "../emojipedia";

function App() {

return (

<div>

<h1>

<span>emojipedia</span>

</h1>

<dl className="dictionary">

{emojipedia.map((emojiTerm) => (

<Entry

key={emojiTerm.id}

emoji={emojiTerm.emoji}

name={emojiTerm.name}

description={emojiTerm.meaning}

/>

))}

</dl>

</div>

);

}

export default App;

in function where we need to check the condition we have to use if else but it not work in jsx because we know it statement not expression so we have ternary operator that work like if else

condistion ? do if true : do if false.