**JavaScript**

1. Java script is an interpreter language which is object based language and scripting language and which has the features like polymorphism, encapsulation, inheritance
2. The main difference between the java and java script is that the java’s code first gets compile than converted to the executable file and than it runs while the java script’s code runs line by line opposite to that of java
3. There are various framework in java script like the bootstrap, angular js, etc
4. Java script can also be used for making the desktop application for windows as well as the mac by the help of the electron js.x
5. Always prefer the Microsoft edge for the development instead of the chrome as some times in the chrome the statements on the console is not printed
6. We can also check the performance of the web app by selecting tools from the browser and for it we have to go to browser and there we have many selection like sources, console, network(No throteling🡪check different networks), performance, memory, security, **lighthouse(make an report before publishing)**
7. And we can put the string in the backtiks instead of the “” and usage of the backticks is the best practice for modern js
8. We can get the information of the button and can set an response by :

let namasteBtn=document.querySelector(“button”);

namasteBtn.addEventListener(“click”,<function-name>);

function <function-name>(){

alert(“Namaste World!”);

}

1. We can make the message to be prompted in the page by the help of the alert method as in the above case.
2. We can also take the value from the user by the help of the :

let name = prompt(“Enter the name of the student”);

🡪And we can easily save it in the variable and use it

1. To make the change in the content which is in the home page we have to make the following type of the code :

namasteBtn.textContent=”name = ”+name

1. The code of the javascript can be written in the three ways
   1. In the .html file
   2. In the new .js file
   3. In the browser
2. We have to always write the javascript in the last of the program as we have to made the code in the .html file render first and then we are able to make it change with the help of js

🡪If we write the java script code in between the .html file than it will give an error

🡪There are three ways by which the js code will run after the html code

1. <script>

Document.addEventListener(“DOMContentLoaded”, function());

{

function updateName(){…}

}

</script>

1. By the help of the differ attribute

🡪The differ attribute will take care that the .html and .css files are rendered first and then the .js file is executed

🡪But by the help of the differ attribute the first js file which we have written will be executed first and then the second one as per the first come first serve model

1. The async attribute is also used when we want to execute the .js file which is not called at the ending of the .html file but in-between

🡪Hence first the .html and .css files load then the .js file

🡪Also in the async attribute the the .js files are loaded together which is opposite to the differ attribute

1. async and defer both instruct the browser to download the scripts in a seprate thread, while the rest of the page (the DOMS, etc) is downloading, so the page loading is not blocked by the scripts.
2. If your scripts should be run immediately and they don’t have any dependencies, then use async
3. If your scripts need to wait for parsing and depend on other scripts and/or the DOM being in place, load them using defer and put their corresponding <script> elements in the order you want the browser to execute them
4. Like java we have to also write the semicolon at the end of the statement in the js
5. And as similar to the java we write the comment in the js with the help of the // and /\*….\*/
6. The js is dynamically typed language which is opposite to the languages like c, java

Eg. **In Java**

**In JavaScript**

var number=12;

var name=”Neel”;

int number=12;

String name=”Neel”;

1. In js there are two types of datatypes
   1. Primitive
   2. User-Defined

🡪In this two types there are many types such as :

1. Primitive
2. Boolean: It represents one of the two values **true or false**

Eg. var YES=true, var NO=false

1. Undefined: A variable is an undefined when **no value is assigned** before using it

Eg. var myVar;

1. String: A string is textual content. It must be enclosed in single or double quotation marks

Eg. var str1=”Hello World”;

Var str2=’Hello World’

1. Null: You can assign null to a variable to denote that currently that variable does not hold any value but it will have later on

Eg. var myVar=null;

1. Number: Number type represents integer, float, hexadecimal, octal or exponential value

🡪In this we can also represent values like plus infinity, minus infinity, NAN(not a number)

🡪When we want to save the big int in the datatype then it is necessary to mention ‘n’ after it’s completion

Eg. var num=23566684425639n;

1. User-Defined
   * 1. Object: It is any other variable, the only difference is that an object holds multiple values in terms of properties and methods.

🡪Properties can hold values of primitive data types and methods are functions

* + 1. Functions: A function allows you to define a block of code, give it a name and then execute it as many times as you want

1. To create the variables in js there are three keywords:
   1. let
   2. const
   3. var
2. If we declare any variable by the help of the const keyword then, its value remains constant and no one can change it
3. The statement in the console can be printed in which we can print the value of the variable in between with the help of the template literal as :

🡪console.log(“${<variable-name>} is variable name”);

1. In js there are five types of the loops
   1. for statement
   2. do…while statement
   3. while statement
   4. for…in statement

Eg. //Object

//Array

Let names=[“Neel”, “Neh”, “Ama”, “Rishi”];

//for in loop

For(let index in names){

Console.log(index, names[index])

}

let animal={

name=”Zebra”,

leg: 4

};

//for in loop

for(let key in animal){

console.log(key, animal[key])

}

* 1. for…of statement

Eg. //Array

Let names=[“Neel”, “Neh”, “Ama”, “Rishi”];

//for of loop

for(let name of names){

console.log(name)

}

1. In java script we can convert the string into the number by the help of :

Number(<String-name>);

🡪While we can convert the string in to the number by the help of the:

Integer.parseInt(<String-name>);

1. We can convert the number into the string by the help of the:

“”+number=””;

<number-variable-name>.toString();

🡪while in java we convert the number into the string by the help of the:

Integer.toString(<Number-value>);

1. In js we can make various operations with string by the help of the pre-defined method, it is not written, if wanted then take it from online

🡪The example of methods are : .length(), .split(), .concat(), ,charAt()and many

🡪The example is <string-name>.toUpperCase();

1. Array in the javascript

Eg. let a=[“neel”, “deven”, “shah”];

1. Object in the java script

Eg. let<object-name>={

name: ”Zebra”,

legs: 4

}

1. Functions in java script

function namasteWorld(name){

console.log(“Namaste ”+name);

}

namasteWorld(“Neel”);

namasteWorld(“Shah”);

🡪One can pass many number of the variables in any function

1. In js one can make an function to be called on clicking button in .html file and the response of that click will be given by the function which is in the .js file

Eg. 🡪In .html file

<button onclick=”giveclick()”>Hello Click Here</button>

🡪In .js file

function giveclick(){

console.log(“Hello friends you are in js file”);

}

1. We can also make any function to be performed when any event is made by the help of the .addEventListener in .js file

🡪But before it we have to define the value of that class or id

Eg. const container=document.querySelector(“#container”);

\*here # is used as it is id if there is class then instead use the .

container.addEventListener(“click”, function(){

console.log(“You clicked”);

});

container.addEventListener(“mouseenter”, function(){

console.log(“You entered”);

});

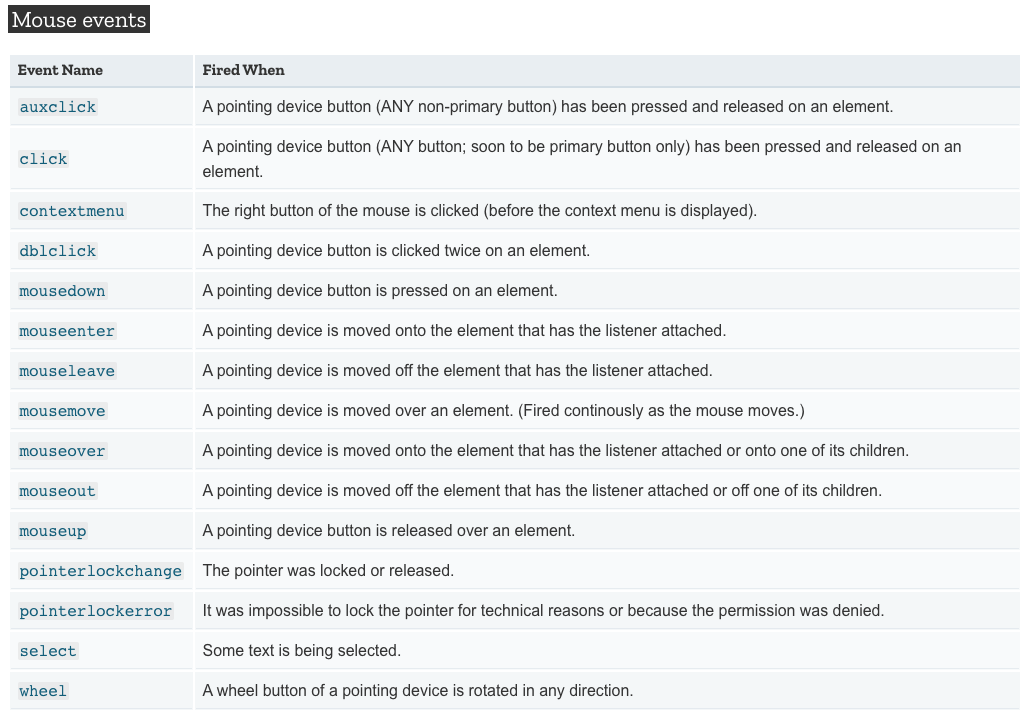
container.addEventListener(“mouseout”, function(){

console.log(“You moved out”);

});

\*here make ensure that you are using the non-chrome browser, as in it the repeated command in not working instead use any other one like Microsoft edge

1. The some of the functions or operations that can be preformed by the help of the event listeners are:



1. The functions in js can be written in two types :
   1. Normal Type(**Use This**)

function sum(a,b){

return a+b;

}

* 1. Arrow Type(**Not Use This**)

Sum= (a,b)=>{

return a+b;

}

1. The task which is stored in function can be set to be done after some time can defined as:

clr=setTimeout(<function-name>,2000);

\*Here in first there is the function name and the second one is time in milli second and where the clr is a variable which is defined as above

🡪The clr can be used to stop the setTimeout cycle, by the help of the clearTimeout(clr) in the console of the browser

1. The same task can be performed after the same interval of the time in js which is defined as:

clr=setInterval(<function-name>,2000);

\*Here in first there is the function name and the second one is time in milli second and where the clr is a variable which is defined as above

🡪The clr can be used to stop the setInterval cycle, by the help of the clearInterval(clr) in the console of the browser

1. We can set the data in the local storage by the help of the following statement :

🡪But we have to set or give the following statement in the console, so that we have the two options

* 1. Write it directly in the console of the browser

localstorage.setItem(‘name’,’neel’);

* 1. In js program in the console.log(<code-here>);

console.log(localstorage.setItem(‘name’,’neel’));

🡪Warning: Do not store the sensitive data in the local storage as it is easily visible and it is stored at the client machine

1. We can also use the data stored in the browser by the help of the following statement :

localhost.getItem(‘<item-name>’);

🡪Here also the name can be got by the two ways

1. The data stored in the browser can be wiped up by the help of the localhost.clear();
2. We can also remove the single item from the local storage by :

localStorage.removeItem(‘<Item-name>’);

1. JSON(JavaScript Object Notation), It is used to receive the data from the java to the js and also used to send the data from the js to the java
2. We can convert an object to the string with the help of the stringify method, The example is as follow :

//For converting the Object into the String

obj={name: “neel”, length: 1, a: {this: ‘that’}}

jso=JSON.stringify(obj);

console.log(typeof jso)

console.log(jso);

//For converting the String into the Object

parsed=JSON.parse(`{“name”:”Neel”,”length”:1,”a”:{“this”:”that”}}`)

console.log(parsed);

1. We can print the new statement or replace the statement in the html file by the help of the following statement in the script tags

🡪In html tags

<h1 class=”statment”>The current time is :<span id=”time”></h1>

🡪In <script> or .js

document.getElementById(‘time’).innerHTML=time+”on”+date;

🡪The main error in this statement is due to that the script tags which are not at the end of the body tag

1. In js all the variables are treated as the object
2. We can also make use of an filter to anything in js, Example:(**In jsx, if in js than do not use {}**)

**<p>{text.split(“ ”).filter((element)=>{return element.length!=0}).length}</p>**

🡪Filters are used to filter any thing

1. We can also use the logical condition in the output like :

<p className=”container”>{text.length>0?text : “Please enter some text”}</p>

1. The following code returns the value of the target or changed value that triggered it:

setText(event.target.value)

1. In onclick method we can not give the function call, we have to give an function
2. By splitting any thing by **“ ”**we can not split any word writed after the enter but by the help of the **/\s+/** we can split both the word after the space and enter
3. We can use the slice method to make the number of the characters limit from a paragraph

console.log(text.slice(0,50));

1. A Map object iterates its elements in insertion order — a for…of loop returns an array of [key, value] for each iteration.
   1. map() creates a new array from calling a function for every array element.
   2. map() calls a function once for each element in an array.
   3. map() does not execute the function for empty elements.
   4. map() does not change the original array.
2. By doing the target=”\_blank”, the new tab is opened for that url, example :

<a href={newsUrl} target=”\_blank” className=”btn btn-sm btn-primary”>Read More</a>

1. We can get the array of the object in js by the help of the :

{this.state.articles.map((element)=>{console.log(element)})}

🡪Here the articles is the array of the object which is being stored in constructor by the help of state keyword

1. By the help of the following code we can get the value which is stored in the articles named variable and which is stored by the help of the state

{this.state.articles.map((element)=>{

            return <div className='col-md-4'>

              <NewsItem target="\_blank" key={element.url} title={element.title} description={element.description} imgUrl={element.urlToImage} url={element.url}/>

              </div>

            })}

1. We can also disable an button by :

<button disabled={this.state.page<1} class=”btn btn-primary onClick={handleCli}”></button>

🡪If the value of the statement inside the braces is true than the disabled will be on

🡪If the value of the statement inside the braces is false than the disabled will be off

1. The Math.ceil function makes the small or the point value to be roundeoff and make it similar to the highest integer of it

Eg. if(this.state.page+1<=Math.ceil(this.state.totalArticles/20))

1. And if we are using the state than we can take the value from the state by using the method like this.state.<variable-name> and it’s example is as shown in above point
2. Backticks are very useful to make the changes like this in the url by the help of the template literal and this facility is not been provided in the “” type. Eg

let url=`https://newsapi.org/v2/top-headlines?country=in&apiKey=90e7f116117d4e6aa91f10765c013282&page=${this.state.page-1}&pageSize=21`;

1. We can make the condition like when the first condition is true than only the second condition will be implemented without using the if else by the help of the && operator and example of it is as follow :

**{this.state.loading && <Spinner>}**

🡪In this type of the operator the id the first condition is proved right than only the second will be implemented otherwise not

1. The javascript follows the asynchronized function and it works on the un-blocking nature, hence some time occurs that if we write an statement first and it will be executed after some other statements as that first statement would have taken more time than the other and hence the statement after it will be executed first than the first one will be executed
2. **Spread Operator**

🡪 Spread syntax can be used when all elements from an object or array need to be included in a list of some kind.

🡪 When we invoke the function, we pass it all the values in the array using the spread syntax and the array name — ...numbers.

🡪 If the array contained more than three numbers, e.g. [1, 2, 3, 4], then it would still work fine, except that all four would be passed, but only the first three would be used unless you added more arguments to the function

🡪 It is commonly used when you want to add a new item to a local data store, or display all stored items plus a new addition.(M.I.M.P.)

1. We can set the value of the any thing in the html by the help of the value={}, Example

**<input type=”text” value={note.title} onChange={onChange}…/>**

1. The json.parse() is used for exchanging the data to/and a web server
2. The json.stringify() method is used to convert a JavaScript object or value to a JSON string
3. We can also make the taking the value of the input compulsory by the help of the **required keyword**

**<input type=”text onChange={onChange} required”/>**

1. We can also make the compulsory minimum length of the input by the help of the minLength keyword

**<input type=”text onChange={onChange} minLength={5}”/>**

1. We can also make the compulsory maximum length of the input by the help of the maxLength keyword

**<input type=”text onChange={onChange} maxLength={5}”/>**

1. The maxLength, minLength, required does not work if there is the onClick() on the submit button instead of the onSubmit(), So to use this methods we should compulsory have the onSubmit() method

**<button type=”submit” onClick={handleClick}>Add Note</button>**

1. Now if one want to use the onClick() and also use the conditions, than the other way is by the usage of the **disabled keyword**

**<button type=”submit” disabled={note.title.length<5 || note.description.length<5} onSubmit={handleClick}>Add</button>**

1. After clicking any button like submit or any thing if you want the page to not be redirect and be remain same as it is than use the preventDefault as the function which is called on the click:

**const handleSubmit=(event)=>{**

**event.preventDefault();**

**}**

1. The fetch() method in JavaScript is used **to request to the server and load the information on the webpages**. The request can be of any APIs that return the data of the format JSON or XML
2. We can also change the state to the opposite, if the variable is having the Boolean value by using the following logic :

setShowCart(state){

            state.showCart =!state.showCart;

        },