**HTML/CSS**

* What are block elements?

-A Block element always takes up the full width and has a top and bottom margin.

* What were the new tags/elements brought in HTML5?

-Elements Article, Audio, Video, footer, header,

-Tags color, date, email, month, number, search

* Have you used PreCSS? What is it?

- No, I haven’t used it yet. PreCSS lets us use Sass-like markup and staged CSS

features in CSS

* What is the difference between Margin and padding?

- padding is the space between the content and the border, whereas margin is the space outside the border.

* Is padding included in the element?

- yes

* What types of selectors are used in CSS?

- Type, class, and id selectors , Attribute, Pseudo-classes and pseudo-elements

And combinators

* What are the available values for property "position"? What happens when you use each value?

- Static

The element is positioned according to the normal flow of the document.

- relativ

The element is positioned according to the normal flow of the document, and then offset *relative to itself* based on the values of top, right, bottom, and left.

* How can I show two div tag elements side by side?

- There are 3 ways in which we can show two div side by side

1. Float
2. Flexbox
3. Grid

* Explain the term responsive web design?
* Responsive Web Design mean that user can render our website on a different variety of devices and screen sizes by automatically adapting to the screen.
* What is css flexbox?

- flexbox layout allows us to easily format HTML, flexbox makes it simple to align items

Vertically and horizontally using rows and columns.

* What is CSS opacity?

-The opacity CSS property sets the opacity of an element. Opacity is the degree to which

content behind an element is hidden, and is the opposite of transparency

* When click on button u need to scroll to the particular div using HTML

-Using Scroll to method

* What is specificity in CSS ?

- Specificity is the means by which browsers decide which CSS property values are the most relevant to an element, therefore, will be applied.

* What is FlexBox?

- Flexbox is one-dimensional layout system that we can use to create a row or a column axis layout

* Pseudo classes and elements

-Pseudo-classes select regular elements but under certain conditions, like when their position relative to siblings or when they’re under a particular state.

Dynamic pseudo-classes

* :link, :visiter, :hover, :active, :focus,

### UI element states pseudo-classes

* :enabled, :disabled, :checked

### Structural pseudo-classes

* :first-child, :nth-child(n), :nth-last-child(n), :nth-of-type(n), :nth-last-of-type(n), :last-child, :first-of-type ,:last-of-type ,:only-child, :only-of-type ,:root ,:empty

### Other pseudo-classes

* :not(x), :target, :lang(language)

## JavaScript

* Difference between null and undefined?
* Null is an assigned value. It means nothing. Undefined means variable has been declared but not defined yet.
* What are promises ?
* A Promise is an object that may produce a single value

Some time in the future : either a resolved value, or a

Reason that it’s not resolved.

* What is Async await?
* Async allows us to write promises based code as if it was synchronous and it checks that

We are not breaking the execution thread. It operates asynchronously via the event-loop.

Async functions will always return a value. It makes sure that a promise is returned and if it is not returned then javascript automatically wraps it ina promise which is resolved with its value.

Await function is used to wait for the promise. It could be used within the async block only. It makes the code wait until the promise returns a result. It only makes the async block wait.

* Javascript is single threaded or multiple threaded and asynchronous or synchronous?
* JavaScript is a single threaded language because while running code on a single thread, it can be really easy to implement as we don’t have to deal with the complicated scenarios. SInce, JavaScript is a single threaded language, it is synchronous in nature.
* How to center the div tag on dom?
* document.getElementById("f").setAttribute("align", "center");
* What are callback functions?
* A callback is a function passed as an argument to another function

This technique allows a function to call another function

A callback function can run after another function has finished.

* What is Closure how do we implement closure?
* A closure is the combination of a function bundled together (enclosed) with references to its surrounding state (the lexical environment). In other words, a closure gives you access to an outer function’s scope from an inner function. In JavaScript, closures are created every time a function is created, at function creation time.

function init (){

var name = “Shakti”;

Function displayName(){

alert(name);

} displayName();

}

init();

* **What is Debouncing and throttling ?**
* **Debouncing: In the debouncing technique, no matter how many times the user fires the event, the attached function will be executed only after the specified time once the user stops firing the event.**
* **Throttling is a technique in which, no matter how many times the user fires the event, the attached function will be executed only once in a given time interval.**
* Prototype in javascript with example.

* **In JavaScript, a prototype can be used to add properties and methods to a constructor function. And objects inherit properties and methods from a prototype.**
* **Const personPrototype = {**

**greet(){**

**console.log(“hello”);**

**}**

**}**

**Const carl = Object.create(personPrototype;**

**carl.greet();**

* **Event loop.**
* **The event loop is the secret behind JavaScript’s asynchronous programming. Js executes all operations on a single thread, but using a few smart data structures, it gives us the illusion of multi-threading.**
* What is the use of filter, map, reduce ?
* The map method is used for creating a new array from an existing one, applying a function to each one of the elements of the first aray.
* The filter method takes each element in an array and it applies a conditional statement against it. If this conditional returns true, the element gets pushed to the output array. If the condition returns false, the element does not get pushed to the output array.
* The reduce method reduces an array of values down to just one value. To get the output value, it runs a reducer function on each element of the array.
* Difference between map and for loop
* **Difference between slice and splice**
* **The splice method returns the removed items in an array. The slice method returns the selected element in an array, as a new array object. The splice method changes the original array and slice method doesn’t change the original array.**
* Difference between Standard function and Arrow Function
* Regular functions created using function declarations or expressions are ‘constructible’ and ‘callable’. Since regular functions are constructible, they can be called using the ‘new’ keyword. However, the arrow functions are only ‘callable’ and not constructible.
* **Difference between == and ===**
* **== means equal to whereas === means equal value and equal type**
* **What is ES6?**
* **ES6 stands for ECMAScript 6. ECMAScript was created to standardize JavaScript, and ES6 is the 6th version of ECMAScript ,it was published in 2015, and is also known as ECMAScript 2015.**
* **Oops concept**
* **Object-oriented Programming treats data as a crucial element in program development and doesn’t allow it to flow freely around the system. It ties data more securely to the function that operates on it and protects it from accidental modification from an outside function. OOP breaks down a problem into several entities called objects and builds data and functions around these objects.**
* **How can you achieve a class feature before ES6?**
* **Prior to ES6, JavaScript had no concepts of classes. To mimic a class, you often use the constructor/prototype pattern.**
* Tell me about Promises?
* “Producing Code” is code that can take some time, “Consuming Code” is code that must wait for the result. A promise is a javascript object that links producing code and consuming code.
* Difference between normal function and an Arrow function scope wise?

## 

## React Js

* What is React Js
* Advantages of React
* How does React Js virtual Dom work ?
* What is the state?
* Difference between state and prop.
* What is the use of refs?
* How to change state?
* What are higher order components?
* How can you update the state from the child component?
* What is the Error boundary? Why use it and how?
* What is component and types of components
* Explain about lifecycle methods
* What is Pure component and which lifecycle method do we use for pure component
* Explain React router concepts
* How do you programmatically  navigate to other page?
* How is React Single Page Application explain with an example.
* What is redux & what are the redux principal
* Explain in detail about different components of redux?
* How do we mimic lifecycle methods in hooks
* How many states promise to have after resolution?
* What are Hooks?
* How to use rest api in React JS
* Tell me a feature that is not available in functional component yet (Note: which is available in class component)
* Explain about useSelector, useDispatch.
* Local storage and Session Storage
* How to stop Rendering
* Set time out/Set interval
* What is Hoisting?
* How to fetch api?
* What is AWS
* How to run bootstrap ?
* What is server side rendering In React?
* life cycle methods of React Component.
* What are the three principles of Redux ?
* what is Props drilling

## Problem Solving:

* Find the second Highest element from the array const arr = [12, 35, 1, 10, 34, 1]
* Program to find out prime numbers in a given array.
* Write a function which takes an array and returns the sum of max, min values in that array, finally attach this to a prototype object.
* Function getMin(arr, n) {

let res = arr[0];

for (let i = 1; i < n; i++)

res = Math.min(res ,arr[i]);

return res;

}

* Function getMax(arr, n) {

let res = arr[0];

for (let i = 1; i < n; i++)

res = Math.max(res ,arr[i]);

return res;

}

* Function getSum(arr, n) {

let min = getMin(arr, n);

let max = getMax(arr, n);

return min + max;

}

* Problem:

arr=[1,7,-6,15,7,9,0,2,-3]

Seq=[7,15,-3]

Return true if the sequence is present in array in the same order else return false

* Create a counter with increment and decrement in react.

**Import {useState} from ‘react’;**

Function App () {

**Const [counter, setCounter] = useState(0);**

return(

**<>**

<p>{counter}</p>

<button **onClick={()=> setCounter(counter + 1)}**>**Increment**</button>

<button **onClick={()=> setCounter(counter -1 )**}>**Decrement**</button>

**</>**

)

**}**

* Create a form using react and show data on screen in a list.
* Write a program to find the sum of two arrays
* Write a program for addition of each index of two array
* Arrange the number into descending order

**Algo and DS**

* Types of sorting technique(selection sort,merge sort,quick sort,Radix sort) explain any one
* Bubble sort algorithm code
* Binary search algorithm
* How can I detect a linked list that is circular?
* What is binary search?

**General Questions:**

* What challenges do you face in your project and how do you solve it

NODE:-

[NodeJS important questions](https://digikull-my.sharepoint.com/personal/abhi_digikull_com/Documents/Microsoft%20Teams%20Chat%20Files/Node%20JS%20important%20questions.pdf)