← HTML QUESTIONS →

(30 Marks)

Q.1 <!DOCTYPE html> is it a tag of html? If not, what is it and why do we use it?
(1 Marks)

Answer.1 No, <!DOCTYPE html> is not a tag of HTML. It is a declaration that tells the browser what version of HTML the document is written in. It is not an element, and it does not have a closing tag.

Q.2 Explain Semantic tags in html? And why do we need it? (**2 Marks)**

Answer.2 Semantic tags in HTML are tags that describe the meaning of the content they contain. For example, the <header> tag is a semantic tag that indicates that the content it contains is a header.

Q.3 Differentiate between HTML Tags and Elements? **(2 Marks)**

Answer.3 An HTML tag is a pair of characters that surround text or other HTML elements. Tags are used to identify the different parts of a web page, such as headings, paragraphs, images, and lists.

Q.4 Build Your Resume using HTML only **(5 Marks)**

Answer.4
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/blob/master/html/Resume.html

Q.5 Write Html code so that it looks like the given image Link **(5 Marks)**

Answer.5
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/blob/master/html/question_5.html

Q.6 What are some of the advantages of HTML5 over its previous versions? **(2 Mark)**

Answer.6 Improved support for multimedia,Better performance,Enhanced accessibility,Increased compatibility

Q.7 Create a simple Music player using html only **(5 Marks)**

Answer.7
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/blob/master/html/musicPlayer.html

Q.8 What is the difference between <figure> tag and <img> tag? **(1 Marks)**

Answer.8 The <figure> tag is used to semantically represent a self-contained content, such as illustrations, diagrams, photos, code listings, etc. It can be used to group related content together and to provide a caption for the image.

The <img> tag is used to simply insert an image into the document flow. It does not provide any semantic information about the image

Q.9 What's the difference between html tag and attribute and give example of some global attributes? **(2 Marks)**

Answer.9 An HTML tag is a pair of characters that surround text or other HTML elements. Tags are used to identify the different parts of a web page, such as headings, paragraphs, images, and lists.

An HTML attribute is a piece of information that can be attached to an HTML tag. Attributes provide additional information about the element, such as its id, class, or style.

Q.10 build Table which looks like the given image Link **(5 Marks)**

Answer.10
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/blob/master/html/Table.html

← CSS QUESTIONS →

(60 Marks)

Q.1 Whats Box Model in CSS & Which CSS Properties are part of it ? **(2 Marks)**

Answer.1 The CSS box model is a way of describing the layout of an HTML element. It consists of four parts:

Content: This is the actual content of the element, such as text, images, or other elements.

Padding: This is the space around the content. It is transparent and does not affect the layout of the element.

Border: This is a line that surrounds the content and padding. It can be of any color or style.

Margin: This is the space outside the border. It is transparent and does not affect the layout of the element

Q.2 What are the Different Types of Selectors in CSS & what are the advantages of them?

(2 Marks)

Answer.2 There are many different types of selectors in CSS, each with its own advantages.

- **Type selectors:** Select elements based on their type, such as or <p>.
- **Class selectors:** Select elements that have a specific class, such as .my-class.
- **ID selectors:** Select elements that have a specific ID, such as #my-id.
- **Attribute selectors:** Select elements that have a specific attribute, such as [href] or [title].
- **Pseudo-class selectors:** Select elements based on their state or condition, such as :hover or :active.
- **Pseudo-element selectors:** Select parts of an element, such as ::first-line or ::after.

Q.3 What is VW/VH & How its different from PX? **(2 Marks)**

Answer.3 VW and VH are CSS units that represent a percentage of the viewport width and height, respectively. PX is a CSS unit that represents a pixel

Q.4 Whats difference between Inline, Inline Block and block ? **(3 Marks)**

Answer.4 Inline elements are displayed on the same line as the text around them. They do not take up a new line in the document flow.

- **Inline-block elements** are displayed on the same line as the text around them, but they can take up a new line in the document flow.
- **Block elements** are displayed on a new line in the document flow. They can contain other elements, such as inline and inline-block elements.

Q.5 How is Border-box different from Content Box? **(2 Marks)**

Answer.5 The border-box value of the box-sizing property includes the content, padding, and border of an element, while the content-box value only includes the content of the element

Q.6 What’s z-index and How does it Function ? **(2 Marks)**

Answer.6 z-index is a CSS property that specifies the stacking order of positioned elements. An element with a higher z-index value will be displayed in front of an element with a lower z-index value

Q.6 What’s Grid & Flex and difference between them? **(5 Marks)**

Answer.7 Grid and Flexbox are both CSS layout models that can be used to create complex layouts. Grid is designed for two-dimensional layouts, while Flexbox is designed for one-dimensional layouts

Grid can be used to create complex layouts, such as grids and calendars. For example, you could use Grid to create a grid of images or a calendar of events.

Flexbox can be used to create simple layouts, such as lists and menus. For example, you could use Flexbox to create a list of products or a menu of options.

Q.7 Difference between absolute and relative and sticky and fixed position explain with example. **(5 Marks)**

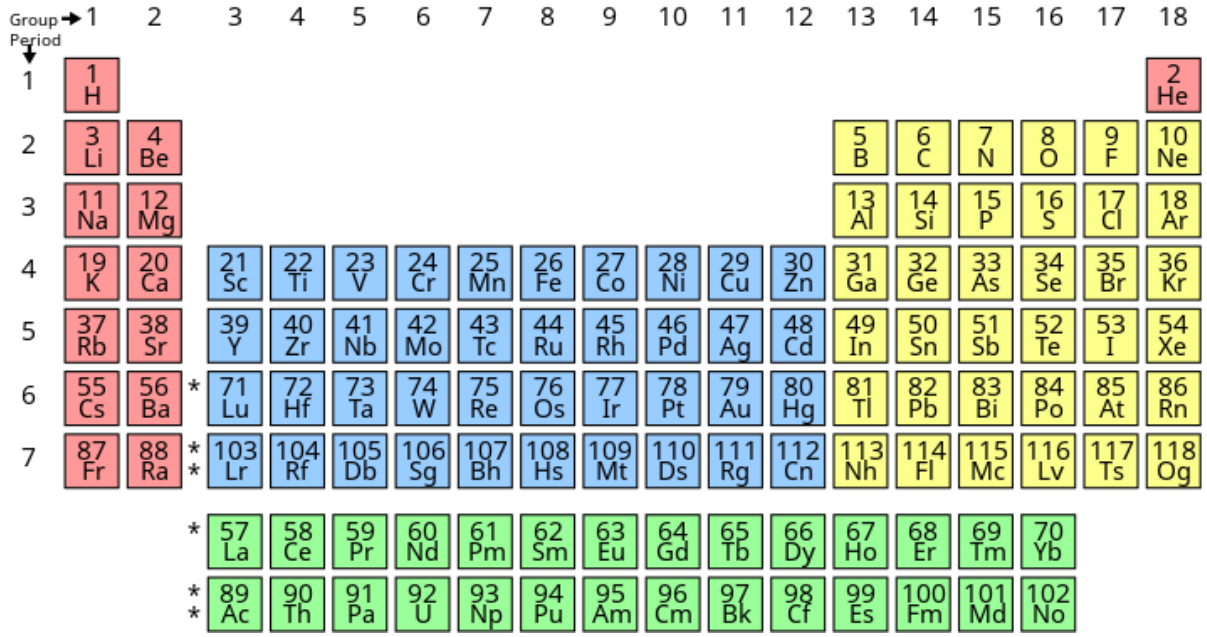
Answer.7 Absolute positioning removes an element from the normal flow of the document and positions it relative to its nearest positioned ancestor.

- **Relative positioning** positions an element relative to its original position in the normal flow of the document.
- **Sticky positioning** positions an element relative to its original position in the normal flow of the document until it reaches a certain point in the document, then it "sticks" to the top of the viewport.
- **Fixed positioning** positions an element relative to the viewport and does not scroll with the page

Q.8 Build Periodic Table as shown in the below image **(10 Marks)**

Answer.8

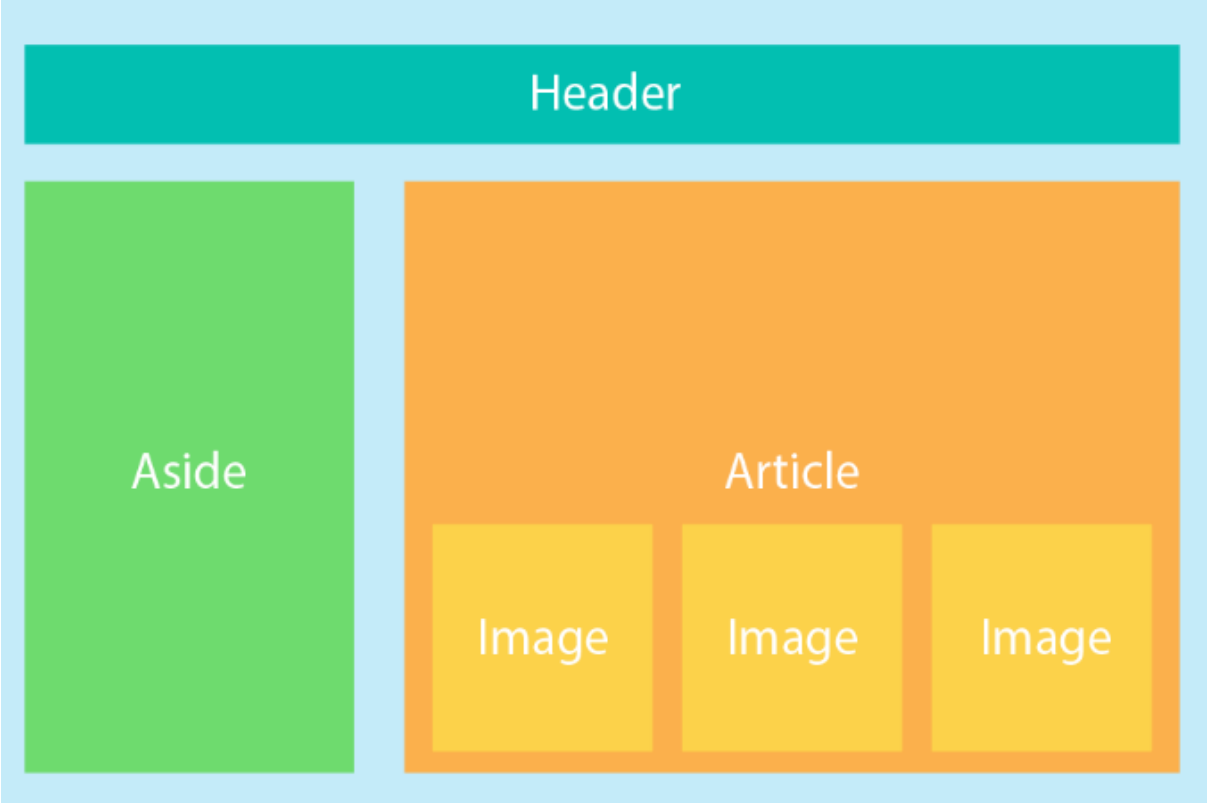
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/css/question_8



Q. 9 Build given layout using grid or flex see below image for reference . **(5 Marks)**

Answer.9

https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/css/question_9



Q.10Build Responsive Layout both desktop and mobile and Tablet, see below image for reference ? **(10 Marks)**

Answer.10
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/css/question_10

Q.11 Build Complete Homepage of Ineuron (Link) with responsiveness.**(10 Marks)**
Answer.11 Link to be added

Q.12What are Pseudo class in CSS & How its different From Pseudo Elements? **(2 Marks)**

Answer.12 Pseudo-classes are used to select elements based on their state or condition, while pseudo-elements are used to add content to an element. Pseudo-classes are preceded by a colon (:) and are not case-sensitive, while pseudo-elements are preceded by two colons (::) and are case-sensitive

← JavaScript Questions →

(40 Marks)

Q.1 What is Hoisting in Javascript ? **(3 Marks)**

Answer.1 Hoisting in JavaScript is the process of moving all variable and function declarations to the top of their scope, before execution of the code. This means that even if a variable or function is declared in the middle of a script, it can be used anywhere in the script after it is declared

Q.2 What are different higher order functions in JS? What is the difference between .map() and .forEach() ? **(2 Marks)**

Answer.2 A higher-order function is a function that takes another function as an argument or returns another function. Higher-order functions are a powerful tool in JavaScript, and they can be used to write more concise and efficient code

Q.3 What is the difference between .call() .apply() and .bind()? explain with an example **(2 Marks)**

Answer.3 call(), apply(), and bind() are all methods that can be used to change the context of a function. The context is the object that the this keyword refers to inside the function

call() takes the context as its first argument, followed by the arguments to the function.

apply() takes the context as its first argument, followed by an array of arguments to the function

bind() takes the context as its first argument, and returns a new function that has the specified context

Q.4 Explain Event bubbling and Event Capturing in JavaScript with suitable examples **(2 Marks)**

Answer.4 Event bubbling and event capturing are two ways that events propagate in JavaScript. Event bubbling is when an event starts at the target element and bubbles up to the parent elements. Event capturing is when an event starts at the top of the document and captures down to the target element

Q.5 What is function currying with example? **(2 Marks)**

Answer.5 Function currying is a technique for transforming a function that takes multiple arguments into a sequence of functions, each of which takes

a single argument. This can be useful for breaking down complex functions into smaller, more manageable pieces.

Q.6 Explain execution context diagram of following code snippets, use white board to draw execution context diagram **(4 Marks)**

```
Code Snippet 1
console.log('First');
setTimeout(() => console.log('Second'), 0);
console.log('Third');

Code Snippet 2

console.log('First');
function secondCall() {
  console.log('Second');
}
setTimeout(secondCall, 2000);
setTimeout(() => console.log('Third'), 0);
console.log('Third');
```

Answer.6 Function currying is a technique for transforming a function that takes multiple argu

Q.7 What are promises? What are the different states of a promise? Support your answer with an example where you need to create your own promise. **(2 Marks)**

Answer.7 Promises are objects that represent the eventual completion or failure of an asynchronous operation. They have three states: pending, fulfilled, and rejected. You can create your own promise using the new Promise() constructor. The then() method can be used to attach a callback function that will be called when the promise is fulfilled or rejected

Q.8 What is 'this' keyword in JavaScript? explain with an example & create **(4 Marks)**

Answer.8 The this keyword in JavaScript refers to the current context. The current context can be a function, an object, or the global object. The value of this can change depending on how the function is called. You can also create your own context using the call() and apply() methods

Q.9 Explain event loop Call Stack Callback queue and Micro Task queue in Your Words **(2 Marks)**

Answer.9 The event loop is a mechanism that JavaScript uses to manage asynchronous code. It has three main components: the call stack, the callback queue, and the microtask queue. The call stack is a stack of function calls, the callback queue is a queue of functions that need to be executed, and the microtask queue is a queue of microtasks. The event loop works by continuously checking the callback queue and the microtask queue, and executing the first function in each queue if it is not empty

Q.10 Explain Debouncing and Create a project where you are using Debouncing **(5 Marks)**

Answer.10 Debouncing is a technique used to prevent a function from being called too frequently. It is often used with functions that are expensive to call, such as functions that make network requests or access the database. Debouncing can be implemented using a timer or a queue.

Q.11 Explain Closures and Use cases of Closures **(2 Marks)**

Answer.11 A closure is a function that has access to variables from its enclosing scope, even after the enclosing function has returned. Closures can be used to implement a variety of programming patterns, such as memoization, currying, and partial application.

Q.12 Create a Blog web app using JavaScript **(10 Marks)**

- Fetch data from https://jsonplaceholder.typicode.com/posts and show it to ui
- User can also add new blog
- Add Delete functionality also

ANSWER

12.https://github.com/mfaizk/Placement_Assignment_MOHD_FAIZ_KHAN/blob/master/javascript/question_12

← React Questions →

(50 Marks)

Q.1 What's React and What are the advantages of it? **(1 Marks)**

Answer.1 React is a JavaScript library for building user interfaces. It is declarative, efficient, and flexible. Here are some of the advantages of React:

- **Declarative:** React is a declarative library, which means that you tell React what you want your UI to look like, and React figures out how to get there. This makes React code easier to read and understand.
- **Efficient:** React uses a virtual DOM, which means that it only updates the parts of the DOM that need to be updated. This makes React applications more efficient and responsive.
- **Flexible:** React is a very flexible library. It can be used to build a wide variety of user interfaces, from simple to complex.

Q.2 What's Virtual Dom in React & What are the advantages of it? **(2 Marks)**

Answer.2 The virtual DOM is a lightweight representation of the actual DOM. It is used in React to make changes to the UI more efficient. Here are some of the advantages of using a virtual DOM in React:

- **Performance:** React only updates the parts of the DOM that need to be updated. This makes React applications more efficient and responsive.

- **Readability:** React code is more readable because it is declarative. This means that you tell React what you want your UI to look like, and React figures out how to get there.
- **Testability:** React code is more testable because it is declarative. This makes it easier to ensure that your code is working as expected.

Q.3 Explain LifeCycle of React Components? **(1 Marks)**

Answer.3 React components have a lifecycle that consists of three phases: Mounting, Updating, and Unmounting.

- **Mounting:** This is the first phase and it happens when the component is first created. During this phase, the component's state and props are initialized and the component is rendered for the first time.
- **Updating:** This phase happens when the component's state or props change. During this phase, the component is re-rendered.
- **Unmounting:** This is the last phase and it happens when the component is removed from the DOM. During this phase, the component's state and props are cleaned up.

Q.4 Whats the difference between between Functional Components and Class Components? **(1 Marks)**

Answer.4

Syntax: Functional components use a function with a return statement to render, while class components use the render() method.

- **State:** Functional components use the useState() hook to manage state, while class components use the this.state property.
- **Lifecycle methods:** Functional components do not have lifecycle methods, while class components have a number of lifecycle methods, such as componentDidMount() and componentDidUpdate().
- **Performance:** Functional components are generally faster than class components, because they do not create a new instance of the component each time they are rendered

Q5. What are the hooks in React & Can we use Hooks in Class Components? **(2 Marks)**

Answer.5 Hooks are a new feature in React that allow you to use state and other React features without writing a class. Hooks are functions that "hook into" React state and lifecycle features from function components. They do not work inside classes

Q6.What are the LifeCycle method and the advantages of it? **(2 Marks)**

Answer.6 The lifecycle methods are a set of functions that are called at specific points in the life of a React component. They allow you to control how the component is rendered, how it reacts to changes in state or props, and how it is destroyed.

They allow you to control how the component is rendered, how it reacts to changes in state or props, and how it is destroyed.

- They make it easier to write reusable code.
- They make it easier to test your components.

Q7.What’s useState Hook & Advantages of it? **(2 Marks)**

Answer.7 The useState hook is a built-in hook in React that allows us to manage state in functional components. It takes an initial state value as an argument and returns an array with two elements: the current state value and a function to update that value

Q8.Explain useEffect & Advantages of it **(2 Marks)**

Answer.8 The useEffect hook is a built-in hook in React that allows us to run side effects in functional components. It takes a function as an argument and returns a cleanup function. The function is called after every render, and the cleanup function is called when the component is unmounted.

It is easier to write reusable side-effectful logic.
It is easier to write code that is easier to understand and maintain.

It is easier to write code that is more performant

Q9.Explain Context Api and create a minor project on it **(5 Marks)**

- Create dashboard and with button on clicking on that change theme to dark and light

Answer.9
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/React/question_9

Q10.Explain useReducer and Its advantages. **(2 Marks)**

Answer.10 The useReducer hook is a built-in hook in React that allows us to manage complex state in functional components. It takes a reducer function as an argument and returns a state variable and a dispatch function. The reducer function is used to update the state variable, and the dispatch function is used to trigger state updates

Q11.build a Todo Web App Using React and useReducer Hook**. (5 Marks)**

Answer.11
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/React/question_11

Q12.Build A simple counter app using React **(4 Marks)**

Answer.12
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/React/question_11

Q13.Build Calculator Using React Only **(4 Marks)**

Answer.13
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/React/question_13

Q14.Build a Tic Tac Toe Game using Class Component of React **(5 Marks)**

Q15.Explain Prop Drilling & How can we avoid it? **(2 Marks)**

Answer.15

Use Context API: The Context API is a way to share data between components without having to pass the data as a prop.

•**Use Higher-order component:** A higher-order component is a function that takes a component as an argument and returns a new component. The higher-order component can be used to inject data into the component that it is wrapping.

Q16.Create a task manager where user can create tasks and see his task **(10 Marks)**
Redirect him to task dashboard section after login

- Use https://reqres.in/ api to authenticate user and redirect him to task manager dashboard where he can see his task and create

Answer.16
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/React/question_16

← Express Question →

(50 Marks)

Q.1 Create a simple server using Express and connect with backend and create an endpoint "/post" which sends 20 posts **(3 Marks)**

Answer.1

https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/Express/question_1

Q.2 Explain a middleware and create a middleware that checks is user authenticated or not then send data of post **(4 Marks)**

Answer.2
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/Express/question_2

In Express, middleware is a function that is executed before a request is handled by a route handler. It can be used to do things like logging, authentication, caching, and error handling.

Q.3 Create a backend for blog app, where user can perform crud operations **(10 Marks)**

- Add blog
- Delete blog
- Update blog
- Replace blog

Answer.3
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/Express/question_3

Q.4 What is the difference between authentication and authorization? **(2 Marks)**

Answer.4 Authentication is the process of verifying the identity of a user or service. Authorization is the process of determining what resources a user or service is allowed to access.

Q.5 What is he difference between common JS and EJS module? **(2 Marks)**

Answer.5 CommonJS and EJS are both module systems for JavaScript. EJS is more modern and flexible than CommonJS, but CommonJS is more widely supported.

Q.6 What is JWT and what we can achieve with that create a minor project with jwt **(5 Marks)**

- Login and sign up
- Add authentication using jwt

Answer.6
https://github.com/redwin-0000/Placement_Assignment_Ashutosh_Singh/tree/master/Express/question_4

Q.7 What should we do with the password of a user before storing it into DB? **(2 Marks)**

Answer.7 Before storing a user's password into a database, you should hash it using a secure hashing algorithm and add a random salt to the password.

Q.8 Whats event loop in NodeJS **(2 Marks)**

Answer.8 Node.js's event loop is a single-threaded loop that runs continuously, listening for events and executing callbacks when they occur. This allows Node.js to handle a large number of concurrent requests without blocking the main thread

Q.9 Create a Full Stack Ecommerce website with all major functionalities. **(20 Marks)**

Answer.9 Link to be added

← Submission Process →

There are Two Types of Question Theory based Question and Project based (where you actually have to code)

First of all, You have to create an google doc, where you will add answers of all the questions

If you are attempting a question in which you have to write code, so create a repo push your code to repo and copy the link of repo and add into docs as shown below

Eg. Answer. 6 JavaScript - > github repo link

If you are attempting a theory based question then you have to add answer in same google docs as its

Eg. Answer 1 JavaScript
Hoisting is JavaScript's default behavior of moving declarations to the top.

Then submit that final link (google doc link which has all the answers)

TOTAL MARKS - 230