

HTML QUESTIONS

01. `<!DOCTYPE html>` Is it a tag of HTML? If not, what is it and why do we use it?

=`<!DOCTYPE html>` is not a type of HTML. It is used in every HTML document that specifies the type of document used.

02. Explain Semantic tags in HTML? And why do we need it?

=Semantic tags simply mean introducing meaning to a web page rather than a presentation. Some semantic tags are p tags and header tags. p tags mean paragraph tags and h tags mean header tags which search engines and users can quickly understand.

`<p>`Semantic tags simply mean introducing meaning to a web page rather than a presentation.`</p>` // its easy to understand that a paragraph about semantic

But when we use non-semantic tags like

`<div>`Semantic tags simply mean introducing meaning to a web page rather than a presentation.`</div>` //not define or clear what they trying to tell.

We need it because it is easy to read and understand by users and web browsers. It offers a better user experience. It has greater accessibility. It gives additional information about the document.

Using semantic tags gives you many more hooks for styling content means when you design your web page and later on want to change the background color, font size, margin, and padding that time due to semantic tags you can easily able to change.

03. Differentiate between HTML Tags and Elements?

=

Html tags=Html tags are keywords that use to construct a simple HTML document, web page, or website.

Html Elements= In Html, an Element is a section of an Html document. Some Html represents visual components on web pages such as text, images, or buttons.

HTML Tags	HTML Elements
Html tags are building blocks of HTML page	Html elements are components that are used in html page
It consists of an opening bracket and an end bracket	It consists of a starting tag, content, and an end tag
They can not be nested	They can be nested
ex-<title></title>,<body></body>	ex-<p></p>,

04. Build Your Resume using HTML only

=GitHub repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/resume-Html

05. Write Html code so that it looks like the given image

= GitHub repo

link:https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/samplehtml

06. What are some of the advantages of HTML5 over its previous versions?

=HTML stands for Hypertext Markup Language. It is used to create web applications and websites. It is the most widely used language for developing web applications.

Before HTML5:-We were unable to play multimedia on our website without the help of Flash player. We were also not capable of creating static web pages that don't contain any dynamic content or functionality. Html was also not mobile-friendly. Maintaining Html code was also tricky because of its lengthy and complex syntaxes and the code offered no or poor security features. Client-side storage was also not there before HTML5.

Search engine optimization was not supported with earlier versions of HTML and you have to write multiple codes for the same website according to different browsers just because it was not cross-platform.

HTML5:-Html5 introduced <audio><video>and <canvas> tags which made it possible to use multimedia within the websites and make them more dynamic.

HTML5 has short and crisp syntax and comes with smart and improved security features hence it became very easy to write and manage HTML5 code.HTML5 introduced semantic tags which are very helpful in defining your web pages into meaningful sections and are really helpful in search engine optimization.

HTML5 is cross-platform which means there is no need of writing different code for different browsers and hence it saved a lot of time and cost. It is also multiple devices friendly

07. Create a simple Music player using HTML only
= GitHub repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/music.html

08. What is the difference between <figure> tag and tag?

=The <image >tag is used to embed the image in an HTML document whereas the < figure > tag is used to semantically organize the content of an image in the HTML document. These two elements are not interchangeable.

09. What's the difference between an HTML tag and an attribute and give an example of some global attributes?

=A tag is a way of representing an HTML element in a program, while an attribute is a way of describing the characteristics of an HTML element.

Tag	attributes
Html tags are used to hold the html element	Html attributes are used to describe the character of an HTML element in detail
Html tag starts with < and ends with >	HTML attributes are found only in starting tag
HTML tags are almost like keywords where every single tag has a unique meaning.	HTML attributes specify various additional properties to the existing HTML Element

Some global attributes are:- style, class, id, title, date,accesskey, lang, spellcheck, translate

10. build a Table that looks like the given image Link.

=GitHub repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/Table

CSS QUESTIONS

01. What Box Model in CSS & Which CSS Properties are Part of it?

=Box Model is a container that contains multiple properties like border, margin, and padding. It is used for creating and designing the layout of web pages. It can be used as a toolkit for customizing the layout of different elements. The web browser renders all the elements as a rectangular box according to the CSS box model.

Box Model have multiple properties in css some of them are:-

padding:-This property is used to create space around the element, inside any defined border.

margin:-This property is used to create space around the element around the border area.

border:-This property is used to cover the content & any padding & also allows setting the style, color, and width of the border.

content:-The content of the box, where text and images appear.

02. What are the Different Types of Selectors in CSS & what are their advantages of them?

=Selectors in CSS:- Selectors in CSS are used to find or select the HTML elements that we want to style.

The different types of Selectors in CSS and their advantages are:-

Id:-This selectors use the Id attributes to select the Html element It is a unique page so it is used to select only one element within a page. To select an element with a specific id, write(#)character followed by the id of the Element.

Class:- These selectors use the class attributes to select the Html element. It is common, at a time we can style many elements within a page. To select elements with a specific class, write(.) character, followed by the class name

Universal selector:-This selector is used to select all Html elements on the page. To select elements with a universal selector, write(*) character.

Grouping Selector:-The grouping selectors select all the Html elements with the same style definitions. It is used to minimize the code. Commas are used to separate each selector in a grouping.

Element Selector:-The element selector selects the Html element by name.

03. What is VW/VH & How it's different from PX?

= VW stands for viewport Width and VH stands for viewport Height which is the viewable screen's width and height. Viewport units represent a percentage of the current browser .100VH represents 100% Of the viewport's width or height.

It is different from px in that PX belongs to the absolute length whereas VW/VH belongs to the relative length. PX is normatively defined as being exactly 1/96th of 1 CSS inch. It is fixed and exactly the screen size moreover VW/VH 100VH represents 100% Of the viewport's width or height.

Q.4 What difference between Inline, Inline Block, and Block?

=Inline:- The element doesn't start on a new line and only occupies just the width it requires. You can't set the width or height ex:- img, span, an.

Inline-block:-It's formatted just like the inline element where it doesn't start on a new line. But, you can set width and height values. ex:- span, em,b,i.

Block:- The element will start on a new line and occupies the full width available. And you can set width and height values. ex:-div,h1,p,li, section.

05. How is Border-box different from Content Box?

=Border-box and Content Box are two different values of box-sizing

Border-box:-In this value, height width properties are included, and not only height and width but you will find padding and border inside of the box .when using box-sizing:border-box; the size of border-box remains same while the size of content decreases as padding and border growing.

content-Box:- In this value, Only height and width properties are included . It does not include a border and padding to the element. Padding and Border take space outside the element. when we use

box-sizing:content-box; the content size remains the same while the border-box size grows as padding and border grow.

06. What's z-index and How Does it Function?

= Z index (z-index) is a CSS property that defines the order of overlapping HTML elements. Elements with a higher index will be placed on top of elements with a lower index.

Z index works on positioned elements like (position: absolute , position: relative, position: fixed) .z-index wasn't defined, it will have a default value of auto. If you need to put a background element below a container. You can easily place the background under every element by giving it a negative z index like:-

```
#background{  
z-index:-1;  
}
```

07. What are Grid & Flex and the difference between them?

=Grid:-CSS Grid Layout, is a two-dimensional grid-based layout system with rows and columns, making it easier to design web pages without having to use floats and positioning. Tables, grids, and layouts allow us to align elements into columns and rows.

To get started you have to define a container element as a grid with display: grid set the column and row sizes with grid-template-column and grid-template-rows and then place its child elements into the grid with grid-column and grid-row.

Flex:-Css Flex Layout is a one-dimensional layout It is useful in allocating and aligning the space among items in a container(made of grids). It works with all kinds of display devices and screen sizes

To get started you have to define a container element as a grid width display:flex;

Difference between them are:-

Grid	Flex
Two- Dimensional	One-Dimensional
Can flex combination of items through space-occupying features	Can push content element to extreme alignment

Layout first	Content first
--------------	---------------

08. The difference between absolute and relative and sticky and fixed position explain with example.

= Relative:- An element with position:relative is positioned relative to its normal position.Setting the top,right,bottom and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position.Other content will not be adjusted to fit into any gap left by the element.

```
div.relative{  
position:relative;  
left:30px;  
Border:3 px solid #73AD21;  
}
```

Fixed:- An element with position :fixed is positioned relative to the viewport,which means it always stays in the same place even if the page is scrolled.The top ,right,bottom and the left properties are used to position the element. A fixed element does not leave a gap in the page where it would normally have been located.

```
Div.fixed{  
position:fixed;  
Border:1px solid red;  
}
```

Absolute:- An element with position:absolute; is positioned relative to the nearest positioned ancestor(instead of positioned relative to the viewport like fixed).if it has no positioned ancestors it uses the document body and moves along with page scrolling.Absolute positioned element are removed from the normal flow and can overlap elements.

```
Div.absolute{  
position:absolute;  
top:80px;  
}
```

Sticky:-An element with position:sticky; is positioned based on the user's scroll position .A sticky element toggles between relative and fixed depending on the scroll position.It is positioned relative until a given

offset position is met in the viewport-then it sticks in place(like position : fixed).

```
div.sticky{  
position: sticky;  
top:0;  
}
```

09. Build Periodic Table as shown in the below image

=GitHub repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/periodictable

10.Build given layout using grid or flex see below image for reference.

GitHub repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/grid

11.Build Responsive Layout both desktop and mobile and Tablet, see below image fo reference ?

= GitHub repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/responsive

12.Build Complete Homepage of Ineuron (Link) with responsiveness.

=Github repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/ineuron

13. What are Pseudo-classes in CSS & How its different From Pseudo Elements?

= Pseudo class:-It is primarily used to style an element under various states. When referring to the state, this includes the condition or user behavior, for example, hover, active, focus, or disabled.

Pseudo-classes	Pseudo-elements
----------------	-----------------

allow you to style an element in a specific state.	Pseudo-elements allow you to style a part of an element
Pseudo classes are prefixed with one colon(:).	Pseudo elements are prefixed with two colons (::).
ex-hover, active, focus,first-child	ex:-after, before,first-letter

JavaScript Questions

01. What is Hoisting in Javascript ?

= When the javascript engine executes the javascript code,it creates the global execution context. The global execution context has two phases creation and execution

During the creation phase, the Javascript engine moves the variable and function declarations to the top of your code. This is known as hoisting in Javascript.

Let's take an example we declare a counter variable and initialize its value to 1

```
console.log(counter)
var counter=1
```

In this example, we reference the counter variable before the declaration . However, the first line of code does.t cause an error. The simple reason is that the javascript engine moves the variable declaration to the top of the script

In the execution phase the code looks like this:-

```
var counter:
console.log(counter)
counter=1
```

02. What are the different higher-order functions in JS? What is the difference between .map() and .forEach()?

=A higher-order function is a function that accepts functions as a parameter and or returns from them as well.

The different higher-order functions in js are:-

The reduce() method:-The reduce() method iterates through an array and returns a single value. it will sum up all the elements of the array.

The forEach() method:-The forEach() method executes a callback function on each of the elements in an array in order.

The filter() method:- The filter () method executes a callback function on each element in an array. The callback function for each of the elements must return either true or false.

The map() method:-The .map() method executes a callback function on each element in an array. It returns a new array made up of the return values from the callback functions.

The difference between .map() and.forEach():-

.map()	.forEach()
returns a new array	but Does not return a new array
It is used to transform the elements of an array	It is used to loop through the elements of an array
This method can be used with other array methods such as the filter() method	This method cannot be used with other array methods.

03.What is the difference between .call() .apply() and .bind()? explain with an example.

=.call():- This call method binds this value, invokes the function, and allows you to pass a list of arguments.

Ex:-

```
function test(arg1,arg2){
```

```
console.log(this.num,arg1,arg2)    //100,10,20
}  
test.call({nume:100},10,20)
```

apply():-this apply method binds this value to the function and allows you to pass arguments as an array.

Ex:-

```
function test(...arguments){  
  console.log(this.num,arguments)    // 100,[1,2,3]  
}  
test.apply({nume:100},[1,2,3]),
```

bind():- This bind method binds this value to the function and returns a new function.

Ex:-

```
function test(arg){  
  console.log(this.number,arg)  
}  
let bindedfn=test.bind({number:100},"arguments")  
bindedfn()  //100,"arguments"
```

04.Explain Event bubbling and Event Capturing in JavaScript with suitable examples

=Event bubbling:-In event bubbling an event propagates from the target element to the outermost element.

Ex:-

```
</html>  
<body>  
  <h2>Bubbling Event in Javascript</h2>  
  
  <div id="parent">  
    <button>  
      <h2>Parent</h2>  
    </button>  
    <button id="child">  
  
      <p>Child</p>  
    </button>  
  </div>  
</body>  
</html>
```

```

        </button>
    </div><br>

    <script>
        document.getElementById(
            "child").addEventListener("click", function () {
                alert("You clicked the Child element!");
            }, false);

        document.getElementById(
            "parent").addEventListener("click", function () {
                alert("You clicked the parent element!");
            }, false);
    </script>

</body>

```

Event capturing:- In event capturing an event propagates from the outermost element to the target element.

Ex:-

```

<html>
  <head></head>
  <body>
    <article id="ancestor" >
      article element
      <div id="parent" >
        div element
        <p id="child" >
          p element
        </p>
      </div>
    </article>
  </body>
</html>

<script>
  // Script to click event handler to capture on each element
  for(let elem of document.querySelectorAll('*')) {

```

```
    elem.addEventListener("click", e => console.log("Capturing:",  
elem.tagName), true);  
  }  
</script>
```

05.What is function currying with example?

=Function currying in javascript simply means a function with multiple arguments into a nested series of functions each taking a single argument.

Ex:-

```
function sum(a,b,c){  
  Return a+b+c  
}  
sum(1,2,3) //6
```

06.

07. 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.

=A Promise is a javascript object representing a value that will be available after an asynchronous operation is completed. Promises can be returned synchronously like regular values, but the value may be supplied at a later point.

A promise is one of these states:

Pending: initial state, neither fulfilled nor rejected.

Fulfilled: meaning that the operation was completed.

Rejected: meaning that the operation failed

```
const promise = new Promise((resolve, reject) => {  
  // run async code  
})  
  
promise.then(  
  (value) => { /* handle a fulfilled promise result */ },  
  (reason) => { /* handle a rejected promise result */ }  
)
```

```

promise.catch(
  (reason) => { /* handle a rejected promise result */ },
)

promise.finally(
  () => { /* handle a any result */ }
)

```

08. What is 'this' keyword in JavaScript? explain with an example & create

=In javascript this keyword always refers to an object. The thing about it is that the object it refers to will vary depending on how and where this is being called.

Ex:-

```

let product=["Rice", "Dal", "Salt"]
let quantity=[2, 3, 1]
let price=[60, 50, 20]

let totaldata={
  data:[],
  totalproduct: function(){
    let totalitem=0
    for(let i=0; i<this.data.length;i++){
      totalitem+=this.data[i].price*this.data[i].quantity
    }
    console.log(totalitem)
  }
}

for(let i=0;i<product.length;i++){
  let dataobj={}
  dataobj.product=product[i]
  dataobj.quantity=quantity[i]
  dataobj.price=price[i]
  totaldata.data.push(dataobj)
}

totaldata.totalproduct()

```

09.Explain event loop Call Stack Callback queue and Micro Task queue in Your Words

=Event loop:-Event loop is just a guardian who keeps good communication with the call stack and callback queue. It checks if the call stack is free,then let us know the callback queue. Then callback queue passes the callback function to the call stack to be executed. When all the callback functions are executed the call stack is out and global execution context is free.

CallStack:-we can consider callstack as a kitchen where all our code is executed or cooked. whenever we try to run a piece of code it goes to callstack first and then executed. It works in LIFO style which means Last in First out method.

Callback Queue:-It's a guard who monitors the stack of asynchronous callback functions who just completed the task of waiting and passed the gate of web API. The callback queue works using FIFO which means the first in first out method.

Micro Task Queue:- microtask are usually scheduled for things that should happen straight after the currently executing script, such as reacting to a batch of actions or making something async.

10.Explain Debouncing and Create a project where you are using Debouncing

=Debouncing is a programming pattern or a technique to restrict the calling of a time consuming frequently by delaying the execution of the function until a specified time to avoid unnecessary CPU cycles and API calls and improve performances.

Debouncing project:- GitHub repo

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/debounce

11. Explain Closures and Use Cases of Closures.

=A closure is a function having access to the parent scope,even after the parent function has closed.

The use case of closures are:-

- 1 using private variables and methods
- 2.Maintaining state between each function call
- 3.Higher-Order functions

12.Create a Blog web app using JavaScript.

= Github

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/blog

React Questions

01. What's React and What are the advantages of it?

=React is a javascript framework that is used for building user interfaces, which can also be applied for creating user interface components.

Advantages of it are:-

simplified scripting, component-based architecture, faster rendering stable code structure, javascript coding easier, extremely competent,excellent cross-platform support, handling dependencies,ui focused designs.

02. What's Virtual Dom in React & What are its advantages of it?

=The virtual DOM (Document object model) is only a virtual representation of the DOM. Every time the state of our application changes, the virtual DOM gets updated instead of the real DOM

The advantages of it are:-

1.Maximized memory usage:- virtual DOM features make it use less noticeable space in memory usage.

2.CPU:-Virtual dom makes React more CPU intensive

3.High Performance:-when react creates virtual DOM from the beginning .It usseses setState().It helps better performance in less time.

4.Productivity:-Virtual DOM is faster and easier to render than UI in the real DOM.

03. Explain LifeCycle of React Components?

=It is defined as the series of methods that are invoked in different stages of the component's existence.

A react component go through four stages of its life:-

1.Initialization:-This is the stage where the component is constructed with given props and default state.

2.mounting:-Mounting is a stage of rendering the jsx returned by the render method itself.

3.updating:-updating is a stage when the state of a component is updated and the application is repainted.

4.unmounting:- As the name suggests unmounting is the final step of the lifecycle of the component where a component is removed from the page.

04. What's the difference between Functional Components and Class Components?

=

Functional Components	Class Components
It is a plain javascript pure function that accepts props as an argument and returns a react components	Class component creates a render function that returns a React element
There is no render method used in it	It must have a render() method returning JSX
It run from top to bottom and once the function is returned it can't be kept alive.	It is instantiated and different life cycle method is kept alive and is run.
Hooks can be easily used in functional components to make them stateful Ex:- <code>const[name,setName]=React.use State("")</code>	It requires different syntax inside a class component to implement hooks ex:- <code>constructor(props){ super(props) this.state={name:''} }</code>

05.What are the hooks in React & Can we use Hooks in Class Components?

=React Hooks are simple javascript functions that allow components to have access to state and other React features.

We can't use a hook directly in a class component but we can use a hook in a wrapped function component with a render prop to achieve this.

```
export default class App extends Component {
  render() {
    return (
      <UseDarkModeHookWrapperComponent
        render={ (darkMode, setDarkMode) => {
          return (
            <div
              style={{
                display: "grid",
                gridTemplateColumns: "200px",
                gap: "2rem",
                maxWidth: "50%",
                placeItems: "center"
              }}
            >
              <ThemeToggler darkMode={darkMode}
setDarkMode={setDarkMode} />
              hello
            </div>
          );
        }}
      />
    );
  }
}
```

06.What are the LifeCycle method and the advantages of it?

=It is defined as the series of methods that are invoked in different stages of the component's existence.

A react component go through four stages of its life:-

1.Initialization:-This is the stage where the component is constructed with given props and default state.

2.mounting:-Mounting is a stage of rendering the jsx returned by the render method itself.

3.updating:-updating is a stage when the state of a component is updated and the application is repainted.

4.unmounting:- As the name suggests unmounting is the final step of the lifecycle of the component where a component is removed from the page.

The advantage of the lifecycle method are we don't want to React to render your current state. Every time we call `setState()` here the component are re-renders. Moreover, this component usually is used to know if a particular component is not affected directly by the prop changes or the state.

07.What's useState Hook & Advantages of it?

=The useState hook can be used to store track of strings, numbers, booleans, arrays, objects, and any combination of these. It allows you to have the state variable in functional components you pass the initial state to this function and it returns a variable with the current state value and another function to update this value.

Some advantages of it are:-

Easy of use:- easy to use. It takes a single argument which is the initial state and returns an array with two elements.

Simple state updates:- Usestate is suitable for simple updates If you only need to update one piece of state .this is the best choice.

Easy to test:- useReducer makes it easy to test state updates.

08. Explain useEffect & Advantages of it

=The useEffect hook allows allow to perform side effects in our components some examples of side effects are fetching data, directly updating the DOM, and time It accepts two arguments the second argument is optional.

Placing useEffect inside the components lets us access the count state variable(or any props) right from the effect. We don't need a special API to read it It's already in the function scope.

09.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

=ContextApi allows us to pass data through our component trees. we are giving our components the ability to communicate and share data at different levels.

Project Github

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/usecontext

10. Explain useReducer and Its advantages.

=A reducer is basically a function that determines a change to an application's state. It takes the help of action by using it to receive and determine that particular change.

The advantage of using the useReducer() is that it allows us to manage complex state changes in a single, centralized place. we can define all of our state updates and actions in the reducer() function making it easier to maintain and debug our application.

11.build a Todo Web App Using React and useReducer Hook.

=Github

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/usereducer/my-app

12 Build A simple counter app using React

=Project Github

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/counter

13.Build Calculator Using React Only

14.Explain Prop Drilling & How can we avoid it?

=Prop drilling is the most the process of passing data from one component via several interconnected components to the component that needs it.

The issue with prop drilling in react is that data from the parent component must always arrive from every level, even if it is only required At the ery end and not at any other levels. using the useContext hook is one of the better solutions to this.

15.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

=Github

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/taskmanager/my-app

Express Question

04.What is the difference between authentication and authorization?

=

Authentication	Authorization
In the authentication process the identity of users is checked for providing access to the system	While in the authorization process, a person's or user's authorities are checked for accessing the resources
In the authentication process, users or persons are verified	While in this process, users or persons are validated
It needs usually the user's login details.	While it needs the user's privilege or security levels
Authentication determines whether the person is a user or not	While it determines What permission the user have

05.What is difference between common JS and EJS module?

=

Common JS	EJS module
Common JS modules are a popular way to modularize javascript code	EJS modules are a newer way to modularize javascript code
They allow you to export and import modulus using the export and import keywords	EJS module use the module keyword
Common JS all of the dependencies for a project are stored in one file called node_modules.	With EJS modules each dependency is stored in its files.

07. What should we do with the password of a user before storing it into DB?

= Before storing it into DB password are encrypted by the UNIX crypt encrypting algorithm before they are stored in the directly.when crypt is used ,only the 1st 8 characters of a password are used.Passwords longer than 8 characters are truncated.

08.What event loop in NodeJS

=The event loop allows Node.js to perform non-blocking I/O operations despite the fact that javascript is single-threaded. It is done by assigning processes to the operations of the operating system whenever and wherever possible.

Q.9 Create a Full Stack Ecommerce website with all major functionalities.

=Github project

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/Ecommerce

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

- Add blog
 - Delete blog
 - Update blog
 - Replace blog
- =Github project

link:-https://github.com/Monika123465/PlacementAssignment_Monika_kumari/tree/main/blog-craud