What is CDN and Important of CDN ?

A CDN or Content Delivery Network is a network of geographically distributed servers designed to deliver web content efficiently and quickly to users.

By caching content closer to users, CDNs reduce load times and improve the overall performance of websites and applications.

How CDN Works

Origin Server: The main server that stores the original content and handles requests from users.

* Edge Servers: These servers cache copies of the content and are strategically located closer to users to minimize latency. When a user requests content, the nearest edge server responds, reducing the distance data must travel.
* DNS Server: This server translates domain names into IP addresses and directs user requests to the appropriate edge server based on the user's location.

Benefits of Using a CDN

* Faster Load Times: By serving content from the nearest edge server, CDNs significantly reduce the time it takes for web pages to load.
* Reduced Bandwidth Costs: CDNs optimize data delivery, which can lower the amount of data that needs to be sent from the origin server, thus reducing hosting costs.
* Improved Availability and Reliability: CDNs can handle traffic spikes and provide redundancy. If one server goes down, others can take over, ensuring continuous service.
* Enhanced Security: CDNs can help protect against DDoS attacks and other security threats by distributing traffic across multiple servers and filtering out malicious requests.

Future of CDNs

* Increased Demand: As internet traffic continues to grow, the need for efficient content delivery will increase, driving further innovation in CDN technology.
* Emerging Technologies: The integration of AI and machine learning may enhance CDN capabilities, allowing for smarter routing and better resource management.
* Focus on Security: With rising cyber threats, CDNs will continue to evolve to provide robust security features to protect user data and maintain service integrity.

We can add React in our project by injecting CDN links in HTML file

What is crossorigin and why do we use it in React CDN links

The crossorigin attribute is used in HTML to manage how browsers handle cross-origin requests for resources like scripts and images. In React CDN links, it helps ensure that resources from different origins can be accessed securely, improving error handling and enabling features like CORS (Cross-Origin Resource Sharing).

<scriptcrossoriginsrc="https://unpkg.com/react@18/umd/react.development.js"></script>

This file is core JS file and it contains whole code of React JS that is developed by Facebook,

<scriptcrossoriginsrc="<https://unpkg.com/react-dom@18/umd/react-dom.development.js>"></script>

Using this file React interacts with the browser DOM.

Helloworld with React

<script

crossorigin

src="https://unpkg.com/react@18/umd/react.development.js">

</script>

<script

crossorigin

src="https://unpkg.com/react-dom@18/umd/react-dom.development.js">

</script>

<script>

const heading = React.createElement(

"h1" ,

{id: 'heading',xyz: 'abc'} ,

"Hello World From React!"

);

const root = ReactDOM.createRoot(document.getElementById("root"));

root.render(heading);

</script>

in above code react create element is just a Javascript Object,

React.createElement takes three arguments   
1. type of element (eg: h1,h2,div,p,etc)

2. object properties like class, id orother attributes

3. contents element that we need to show in browser.

ReactDOM.createRoot allows us to create root for rendering the Rect application

root.render method is used to render react application to specified root and it replace the content inside root div.

Create Nested Element with Sibllings

const parent = React.createElement(

'div',

{id:'paremt'},

[

React.createElement(

'div',

{id:'child'},

[

React.createElement(

'h1',

{},

"I am h1 tag !"

),

React.createElement(

'h2',

{},

"I am h2 tag !"

)

]

),

React.createElement(

'div',

{id:'child2'},

[

React.createElement(

'h1',

{},

"I am h1 tag !"

),

React.createElement(

'h2',

{},

"I am h2 tag !"

)

]

)

]

);

const root = ReactDOM.createRoot(document.getElementById("root"));

root.render(parent);

we can pass siblings with array.

<script> inside head or body

The scripts are loaded before the body content, which can be useful if you want to ensure that the libraries are available before any other scripts run.

creating a simple prototype, placing the <script> tags at the end of the <body> is often the best practice, as it allows the HTML to load first and improves the user experience

What if there is already content in root div which is render by React using render() function

<div id="root">

<h1> Hello World!</h1>

</div>

inside JS   
  
const root = ReactDOM.createRoot(document.getElementById("root"));

root.render(parent);

If there is already html inside root then it will load on the browser’s DOM and shown on the page but as soon as Javascript reaches script tag which import react code then it will replace html content with react code,  
  
Library Vs Frame work  
  
React is Javascript library.

A library provides a set of helper function or object or module which application call that functionaly and it has narrow scope.

Framework defined open or unimplemented functions or object which use tocreate appliaction and it has wide scope andincludes almost everything required to make application as per needs.