



**You**

Hi chat gpt



**ChatGPT**

Hello! How can I assist you today?



**You**

OK I have a task for you



**ChatGPT**

Sure, I'm here to help. What task would you like assistance with?



**You**

Let's take my interview I'm mern stack developer one by questions ok



**ChatGPT**

Absolutely, I'm ready for your MERN stack developer interview questions. Go ahead and ask, and I'll do my best to assist you.



**You**

I'm a candidate take my interview give me a question i will answer you



**ChatGPT**

Great! Let's start with a fundamental question: Can you explain what MERN stack is and the role of each component (MongoDB, Express.js, React, Node.js) in the stack?



You

In Urdu



ChatGPT

کا مطلب ہے MERN اسٹیک کو معمولی زبان میں بیان کریں۔ MERN، پیشک

- **MongoDB:** یہ NoSQL ڈیٹا بیس ہے جو ڈیٹا کو ڈاکیومنٹس میں ذخیرہ کرتا ہے۔
- **Express.js:** یہ ویب ایپلیکیشن کو بنانے اور منظم کرنے کے لئے چارچیب ہے۔
- **React:** یہ یوزر انٹرفیس بنانے کے لئے ہے اور ویب ایپلیکیشن کے فرنٹ اینڈ یوزر انٹرفیس کو مدیریت کرنے کے لئے ہے۔
- **Node.js:** یہ سرور سائیڈ جاوا اسکرپٹ رن ٹائم ہے جو ویب ایپلیکیشن کو چلاتا ہے اور اسے بیٹا ہے۔

اسٹیک کے ہر حصے کے رول کا خیال ہے؟ MERN آپ کو ہمارے



You

Next question in Roman urdu



ChatGPT

Certainly! Agla sawaal yeh hai: Kya apko RESTful APIs aur unke fayde pata hain, aur MERN stack mein inka istemal kyun hota hai?



You

Roman Mai type kero



ChatGPT

Beshak! Agla sawaal yeh hai: Kya apko RESTful APIs aur unke fayde pata hain, aur MERN stack mein inka istemal kyun hota hai?



You

Nhi pata ap bataye

**ChatGPT**

Certainly! RESTful APIs, ya Representational State Transfer, ek standard web development architecture hai jo interactions ko HTTP protocol ke zariye handle karta hai. Iski kuch key characteristics hain, jaise ke statelessness aur client-server communication.

MERN stack mein RESTful APIs ka istemal is liye hota hai taki front-end (React) aur back-end (Node.js with Express) components aapas mein data share kar sakein. APIs ke istemal se data ko asaanise se bheja ja sakta hai, jisse web applications dynamic ho jaate hain.

RESTful APIs ke fayde mein include hain scalability, flexibility, aur compatibility. Kya apko RESTful APIs aur inke istemal mein kuch aur maloomat chahiye?

**You**

What is compatibility

**ChatGPT**

Compatibility refers to the ability of different systems, components, or software to work together without issues or conflicts. In the context of RESTful APIs, compatibility means that the API can be easily integrated and used by various clients (like front-end applications) and servers (back-end services) without facing major compatibility issues.

In simpler terms, a compatible API can seamlessly interact with different systems and environments, ensuring smooth communication and data exchange. This is crucial in web development, where diverse technologies may need to work together to create a cohesive and functional application.



**ChatGPT**

Zarur! Yeh raha agla sawaal: Kya aap React mein "state" ka tasawwur samajh sakte hain aur yeh "props" se kaise mukhtalif hai?



**You**

State kisi bi event ko handle kerty hoje change kersekty ya us ko by default rakh dia jata hay. Or props ka difference ye heky props say data pass Kia jata hay or ap apni components ma render bi kerwasekty hai us data ko agar many jawab sahi to good likho



**ChatGPT**

Bilkul, aapne sahi samjha hai! React mein, "state" ek component ke internal data ko represent karta hai. Aap state ka istemal kisi bhi event ya action ke response mein data ko handle karne ke liye kar sakte hain, aur isay component ke lifecycle ke mutabiq update bhi kar sakte hain.

"Props" (short for properties), on the other hand, external data hote hain jo parent component se child component mein pass hota hai. Props read-only hote hain, matlab ke child component props ko directly modify nahi kar sakta. Yeh data child component ke render hone par use hota hai.

Aapka jawab bilkul sahi hai! Agar aapko koi aur sawaal hai ya kuch aur samajhna hai, toh poochhein.





**ChatGPT**

Beshak, samajh gaya! Agla sawaal yeh hai: React mein "Virtual DOM" ka concept samjha sakte hain aur yeh kyun istemal hota hai?



**You**

Bilkul hamaray pass react mai do dom hoty hai react ka apna virtual dom or javascript ka dom in mai diffrence ye heky javascript ka jo dom tha agar koi bi file me changes hoti thi to pura page Mai data reload hota tha or virtual dom React ka apna banaya howa hai jo hamy ye facility deta hai ke ap jis jaga changes ker rahay ho wo hi jaga change hoky or usi jaga reload hoky wahi render hojaye jesy ke hamary pass 3 components hai header, contact, footer to agar me footer change kerta Ho to bina reload kiye pori file ko render kery ga agar me 1st walay component mai changes kern to wo apny child components ko bi re render kery ga summary ye haiky unnesseary rendering ko rokta hai agar jawab durust hoto bataou or usky Bad next question in Roman



**ChatGPT**

Bilkul sahi samjha aapne! Virtual DOM ka istemal unnecessary rendering ko rokne mein aur page ko efficient taur par update karne mein madad karta hai.

---

Agla sawaal yeh hai: React Hooks kya hote hain aur aap kuch popular React Hooks ke naam bata sakte hain?



**You**

G bilkul react Ka apne banay Gaye functions hai jinka kuch na kuch kaam hota hai jeseke usestate, to State ko handle kerta hai useeffect hai jo by default page refresh hony pe chalta hai hamray pass bohat hooks hain jesyky 2 oper bataye hai and useCallback, useId mashoor hooks mai sai kuch ye hai



**ChatGPT**

Bahut acha! Aapne sahi taur par bataya hai. `useState` state ko handle karne ke liye aur `useEffect` page refresh hone par chalne wala hook hai.

Aur haan, aapne kuch aur popular React Hooks ka zikr kiya hai. `useCallback` callback functions ko memorize karne ke liye istemal hota hai aur `useRef` DOM elements ya values ko persist karne ke liye istemal hota hai.

Agla sawaal yeh hai: Higher-order Components (HOCs) kya hote hain aur unka istemal kyun hota hai React mein?



**You**

Iska jawab aap dain



**ChatGPT**

Higher-order Components (HOCs) React mein functions hote hain jo ek component ko lete hain aur us par kuch additional functionality ya properties add karte hain, fir ek naya component return karte hain. Yeh component reuse ko badhane, logic ko separate karne, aur code maintainability ko improve karne ke liye istemal hote hain.

Jaise ke aap kisi component ko decorate kar sakte hain error handling, authentication logic, ya koi aur specific functionality ke liye HOCs ka istemal karte hue. Kya aapko iske baare mein aur kuch



**ChatGPT**

Zarur! Agla sawaal yeh hai: React Router kya hai aur iska istemal kyun hota hai React applications mein?



**You**

Bilkul react router react ki third party library hai jisy hum React router dom say import kerty hai or or usko istemal isliye Kia jata hai routing asanai say hojaye or protected routing bi



**ChatGPT**

Bilkul sahi! React Router, jo React ki third-party library hai, routing ko asaan banane mein madad karta hai. `react-router-dom` ka istemal hum karte hain taake hamare React applications mein navigation ko manage karna aur alag-alag routes par alag-alag components ko render karna asaan ho.

Iske saath hi, React Router ki madad se hum protected routing bhi implement kar sakte hain, jisse certain routes ke liye authentication ya permissions check kiye ja sake. Kuch aur poochna hai iske baare mein?





**ChatGPT**

Beshak! Agla sawaal yeh hai: Redux kya hai aur React applications mein iska istemal kyun hota hai?




**You**

Redux bi Third party library hai



**ChatGPT**

Bilkul, sahi kaha! Redux bhi ek third-party library hai jo state management ko handle karne mein madad karta hai. React applications mein Redux ka istemal state ko centralized rakhne, state management ko asaan banane aur components ke beech data sharing ko improve karne ke liye hota hai.

Redux ke store mein application ki state maintain hoti hai, aur components uss state ko access karke apne updates ko manage karte hain. Isse complex applications ke state ko efficiently handle karna asaan ho jata hai. Kuch aur poochna  Redux ke baare mein?



**ChatGPT**

Agla sawaal yeh hai: React Component Lifecycle methods kya hote hain aur unka istemal kyun hota hai?




**You**

Taky components ki usability easy sai ho



**ChatGPT**

Bilkul! React Component Lifecycle methods components ke different stages mein specific actions perform karne mein madad karte hain, jisse components ka behavior control hota hai. Yeh methods components ke creation, update, aur destruction ke phases mein aata hai, jisse developers ko components ko manage kar  ur unhe specific functionality dena asaan ho jata hai.

Agla sawaal yeh hai: React mein Controlled Components aur Uncontrolled Components mein kya antar hota hai?



You

Jawab den



ChatGPT

Controlled Components mein component ka state external source (usually parent component or Redux store) se control hota hai, aur ismein data ko update karne ke liye callbacks ka istemal hota hai. Iske opposite, Uncontrolled Components mein component ka state internal hota hai, aur DOM elements directly reference kiye jaate hain.

Controlled Components ka istemal jab data flow ko control karna important ho, jaise form fields ke liye. Uncontrolled Components ka istemal jab aapko React state mein data store karne ki zarurat nahi hoti, jaise ke user input ko direct access karne ke liye. Kuch aur puchna hai is topic mein?



Zarur! Agla sawaal yeh hai: React Functional Components aur Class Components mein kya antar hai, aur aap konsa istemal karte hain?



You

Jawab den



ChatGPT

Functional Components functional programming paradigm ko follow karte hain aur state ya lifecycle methods ka istemal nahi karte (lekin React 16.8 ke baad Hooks ke aane ke baad state aur lifecycle events ka istemal functional components mein bhi ho sakta hai). Inme sirf `render` method nahi hota.

Class Components traditional approach hai, jisme state aur lifecycle methods ka istemal hota hai. Inme `render` method hota hai aur yeh class syntax ka istemal karte hain.

React Hooks aane ke baad, functional components mein state aur lifecycle features use karne mein flexibility badh gayi hai. Aap konsa istemal karte hain, aur kya aapne Hooks ka istemal kiya hai?







### ChatGPT

Bilkul, shuru karte hain! Pehla sawaal: CSS Grid aur Flexbox mein kya antar hai, aur aap in dono ko kab istemal karte hain?



### You

Ziada usability flexbox ki hoti jese ki navbar mai data ko horizontal or vertical is mai bohat features hai flex box Mai. way me lany ka asan tariqa Hai and grid bohat acha hai or uski usability esi hai ky hamay kisi data ko alag alag sizes mai ya cards mai



### ChatGPT

Bilkul sahi! Aapne sahi samjha hai. Flexbox ko mainly one-dimensional layout ke liye istemal kiya jata hai, jaise ke horizontal or vertical alignments. Jabki CSS Grid, two-dimensional layout ke liye istemal hota hai, jisse aap rows aur columns ke sath complex layouts create kar sakte hain.

Flexbox se aap components ko asani se align aur distribute kar sakte hain, jaise ke aapne navbar ka example diya. CSS Grid ke zariye aap complex grids create kar sakte hain, jahan har element ko specific position di ja sakti hai.



---

Agla sawaal: CSS specificity kya hai aur iska importance kya hai styling mein?



### You

Specificity ka mean



### ChatGPT

CSS specificity ek rule hai jo determine karta hai ke browser kis style rule ko apply karega jab kai rules conflicting hote hain. Specificity ko define karne ke liye aap selectors mein diye gaye elements, classes, IDs, aur inline styles ka combination dekhte hain.

Har selector ko ek weight assign hoti hai. Jaise ke:

- Inline styles ki weight sabse zyada hoti hai.
- IDs ka weight inline styles ke baad aata hai.
- Classes aur pseudo-classes ka weight IDs ke baad hota hai.
- Elements selectors ka weight sabse kam hota hai.

Agla sawaal: CSS Pseudo-classes aur Pseudo-elements mein kya antar hai aur aap kaise istemal karte hain?



**You**

Pseudo-classes Active, hover, after, before ha or bi hai pseudo elements mai y hai ke wo bi elements ko target kerky ye hover, or Active is trha ki classes implement hoti hai



**ChatGPT**

Bilkul sahi! Pseudo-classes aur Pseudo-elements dono CSS selectors hote hain.

- **Pseudo-classes:** Jaise ke `:hover` ya `:active`, ye states ko target karte hain jo user interaction ke dauran hoti hain. For example, `:hover` is used to style an element when the user hovers over it, and `:active` is used for the active state during a click.
- **Pseudo-elements:** Jaise ke `::before` ya `::after`, ye actual DOM elements ke pehle ya baad mein virtual elements create karte hain. Inko content ke sath style karne ke liye use kiya jata hai.

Aapne sahi taur par samjha hai. Kuch aur p~~ro~~chna hai?





**ChatGPT**

Agla sawaal: CSS Flexbox aur Grid mein kya antar hai aur aap kis situation mein konsa istemal karte hain?



**You**

depend karta Hai wesy mai ziyada flexbox use karta Ho wo easy bi rehta hai mujhy horizontal, vertical ways me to mai flexbox ka istemal karta ho and grids bohat rare karta Ho jesy coloums and rows ka koi work ho



**ChatGPT**

Bilkul, aapne sahi observation kiya hai. Flexbox ko mainly one-dimensional layout problems ke liye design kiya gaya hai, jaise horizontal or vertical alignments, jabki CSS Grid two-dimensional layout ke liye hai, jisse aap rows aur columns ke sath intricate layouts bana sakte hain.

Aapki preference ke mutabiq, agar aapko primarily horizontal or vertical layout ki zarurat hoti hai to Flexbox ka istemal karna logical hai. Grid ka istemal tab hota hai jab aapko rows aur columns mein complex layout create karna hai.

Agla sawaal: CSS Transition aur CSS Animation mein kya antar hai, aur aap kaise decide karte hain kis technique ka istemal karna chahive?



**You**

One and two dimensional mai kiya farq hai



**ChatGPT**

One-dimensional aur two-dimensional terms layout design ko describe karne ke liye istemal hote hain.

- **One-dimensional:** Isme layout sirf ek dimension mein define hota hai, jaise ke horizontal (left to right) ya vertical (top to bottom). Flexbox ek one-dimensional layout system hai, jisme aap sirf ek direction mein items ko arrange kar sakte hain.
- **Two-dimensional:** Isme layout dono dimensions mein define hota hai, yani horizontal aur vertical dono. CSS Grid ek two-dimensional layout system hai, jisme aap rows aur columns dono dimensions mein control kar sakte hain, jisse aap complex layouts create kar sakte hain.

Yeh dono concepts layout design ke tariqay ko describe karte hain. Koi specific example ya clarification chahiye toh batayein.



**ChatGPT**

Agla sawaal: CSS Box Shadow kya hota hai aur aap kaise uska istemal karte hain?



**You**

Box shadow ham Tab istemal kerty hai jab hama kisi Box, container, ya koi bi elements ky peechy lagana ho khair me apko batata ho lagana kesy hai css mai Box shadow name ki property hoti hai and wo values,color mangta he ap usy apny hisab sy de kr 1 khoobsorat Box shadow desekty hai



**ChatGPT**

Bilkul, aapne sahi samjha hai! **box-shadow** CSS property ka istemal element ke liye shadow create karne mein hota hai. Aap ismein values like horizontal offset, vertical offset, blur radius, spread radius, and color specify kar sakte hain.

Ek basic example:

CSS

Copy code

```
.box {  
  box-shadow: 5px 5px 10px #888888;  
}
```

Is example mein:

- Horizontal offset: 5px
- Vertical offset: 5px
- Blur radius: 10px
- Spread radius: Default (it's 0)
- Color: #888888

Aap alag-alag values aur combinations ka istemal karke different styles create kar sakte hain. Koi aur CSS property ya concept discuss karna hai?

👍 🗑️ 🔄 🗑️