

## SEZIONE 1 REACT & [NEXT.JS](#)

- 1) a) SSR genera ad ogni richiesta, SSG le pre genera in build time
- 2)
- 3) b) Recupera dati ad ogni richiesta server-side
- 4) Le HOOKS permettono di gestire il cambio di stato nei componenti senza l'uso di classi, di conseguenza rendendo il codice più pulito e leggibile e riusabile anche per chi visualizza il codice per la prima volta
- 5) b) /pages/api
- 6) c) Un oggetto con stato e setter
- 7)
- 8) c) useMemo

## SEZIONE 2 HTML/CSS/JS

9) b) display: grid

10) Il seguente codice JS inizialmente con il metodo filtra crea un nuovo array che contiene solamente i numeri pari, poi attraverso il metodo map tutti i numeri all'interno dell'array filtrato vengono moltiplicati per 2

11) Le differenze sono : CONST è una costante e non può mutare di valore, LET al contrario di CONST permette al suo valore di mutare ma non di essere dichiarato nuovamente nel codice, e VAR permette di essere dichiarato nuovamente a differenza di LET nello stesso blocco di codice, e il suo scope è globale

12) b) <meta name="viewport" content="width=device-width, initial-scale=1.0">

13) Cosa significa async/await in JavaScript e quando lo useresti? Async permette ad una funzione di operare in modo asincrono garantendo una promessa, await è una keyword che is utilizza sulle funzioni async e fa in modo di aspettare la promessa che venga elaborata o rigettata

## SEZIONE 3 BACK-END & DATABASE

14) select \* from students  
WHERE students.age > 25  
ORDER BY([students.name](#));

15) b) === CONTROLLA ANCHE IL TIPO DI DATO

16) La seguente query effettua il conto, per ciascun utente presente, di tutti gli ordini effettuati

17) c) 404

18) d) PATCH

#### SEZIONE 4

19) Responsive design in breve riguarda la realizzazione del design di un prodotto destinato per tutti i tipi di dispositivi (mobile, desktop, tablet, laptop). Alcune tecniche che adotterei sono il mobile first, che riguarda lo realizzare lo stile destinato per i dispositivi mobile in modo da stabilire una base semplice e funzionale, per poi aggiungere regole e stili più complessi e test su diversi dispositivi e risoluzioni, per garantire una buona esperienza utente ovunque

20) Le tecniche che adotterei per rendere un sito più veloce sono :

1. diminuire la dimensione di file multimediali (foto e video) utilizzando formati adatti al web come ad esempio il formato WEBP per le immagini
2. cercare di ottimizzare e ridurre la quantità di codice del sito web
3. Ridurre o minimizzare richieste HTTP, l'uso di plugin o script esterni non necessari