

React opfrisker

Teoretiske spørgsmål

Spørgsmålene besvares skriftligt

1. Hvad er React og hvad bruges det til?

React er et Javascript-bibliotek lavet til at bygge webapplikationer.

2. Hvad er en komponent i React?

Genanvendelig og uafhængig bider af kode. Fungerer isoleret og returnerer HTML, i modsætning til Javascript funktioner.

3. Hvilke forskellige metoder kan du bruge til at style dine React-komponenter?

Diskuter fordele og ulemper ved mindst to metoder.

CSS Stylesheets er nemme at bruge, men kan skabe konflikter og vedligeholdelsesproblemer.

CSS-moduler undgår disse konflikter og er lettere at håndtere, men kan være mindre dynamiske og adskille styles fra komponentens logik.

4. Hvad er state i React?

I React refererer "state" til et objekt, der håndterer data og bestemmer, hvordan en komponent renderes og opfører sig. State er en central del af React, da den giver komponenter mulighed for at holde styr på dynamiske data, som kan ændres over tid, og reagere på disse ændringer.

5. Hvad er props i React?

Props er et specielt nøgleord der bruges til at sende data fra en komponent til en anden.

6. Hvordan adskiller state sig fra props?

Props er data, der sendes til komponenten udefra og kan ikke ændres af komponenten.

State er data, som komponenten selv administrerer og kan ændre.

7. Hvader hooks i React?

Hooks giver funktionskomponenter adgang til State og andre React-funktioner.

8. Forklar, hvordan useState og useEffect hooks fungerer med eksempler.

Med useState kan vi lave en tæller, og med useEffect kan vi lave en timer, der tæller op hvert sekund.

```
import React, { useState } from 'react';

function Counter() {
  // Vi starter tælleren ved 0
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>Du har klikket {count} gange</p>
      <button onClick={() => setCount(count + 1)}>
        Klik mig
      </button>
    </div>
  );
}

export default Counter;
```

```
import React, { useState, useEffect } from 'react';

function Timer() {
  const [seconds, setSeconds] = useState(0);

  useEffect(() => {
    // Start en timer, der tæller op hvert sekund
    const interval = setInterval(() => {
      setSeconds(seconds + 1);
    }, 1000);

    // Ryd op efter timeren, når komponenten fjernes
    return () => clearInterval(interval);
  }, []);

  return (
    <div>
      <p>Timer: {seconds} sekunder</p>
    </div>
  );
}

export default Timer;
```

9. Hvad er JSX?

Det er ligesom HTML og Javascript blandet sammen. Giver os lov til at skrive HTML-elementer i Javascript.

10. Hvordan adskiller JSX sig fra almindelig JavaScript?

JSX laver HTML-elementer om til React-elementer.

11. Hvad er console.log() og hvilken rolle spiller det i JavaScript udvikling?

Det giver dig lov til at tjekke i konsolen inde i browseren, om der er hul igennem; om koden virker eller om der er nogle fejl der skal rettes.

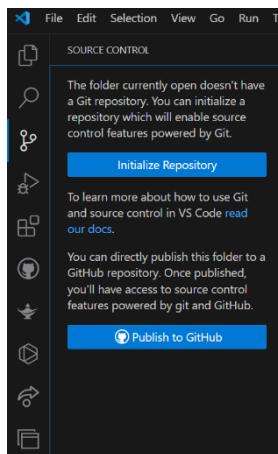
12. Hvornår er det nyttigt at bruge console.log()?

Når koden ikke kommer frem i browseren, og du ikke selv ved hvad fejlen kan være.

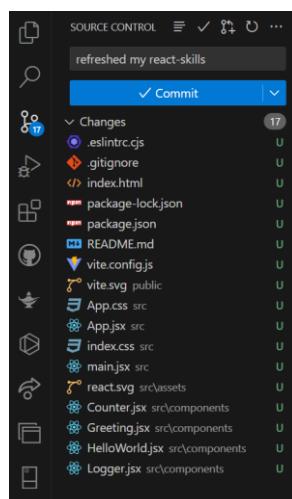
Praktiske opgaver

1. Opret et nyt vite-projekt med et selvvalgt navn
2. Opret en komponent der returnerer en hilsen som f.eks. "Hello, World!"
3. Opret en komponent, der tager 'name' som en prop og viser en besked som "Hello, [name]!".
4. Lav en komponent, der bruger useState hook'en til at tælle, hvor mange gange en knap er blevet trykket på.
5. Lav en komponent, der bruger useEffect til at logge en besked til konsollen, hver gang komponenten renderes.
6. Tilføj din skriftlige besvarelse som en PDF i roden af projektet.

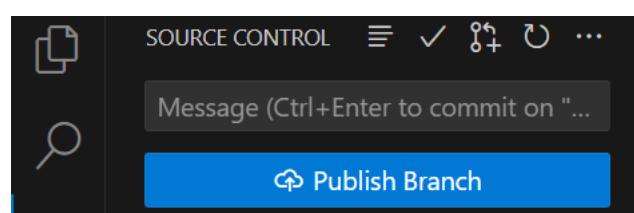
- Åbn Source Control og tryk på initialize repository for at oprette et nyt repository på Github.



- Skriv en commit-besked i 'message'-feltet og tryk 'commit'.



- Tryk derefter 'Publish Branch' og vælg 'Public repository'.



- Send linket til dit repository til mig på Teams når du er færdig