

Teoretiske spørgsmål Spørgsmålene besvares skriftligt

1. Hvad er React og hvad bruges det til?

React er et frontend- og open source-JavaScript bibliotek. -Bruges til udvikling af hjemmesider.

2. Hvad er en komponent i React?

Komponenter er ligesom funktioner, som returnerer HTML elementer.

```
Fx function Car() {  
  return <p>Hej</p>;  
}
```

3. Hvilke forskellige metoder kan du bruge til at style dine React-komponenter? Diskuter fordele og ulemper ved mindst to metoder.

CSS module!

Almindelig CSS

4. Hvad er state i React?

React-komponenter har en indbygget state tilstand. Det er her der gemmes værdier...

```
Fx this.state = {brand: "volvo"};
```

5. Hvad er props i React?

Props er faktisk data som bliver overført fra ét dokument til et andet. Det er et sted du opbevarer dine oplysninger og data

Det er Parent som kalder på Child

```
Så render man og return(  
<ChildComponent/>  
)
```

6. Hvordan adskiller state sig fra props?

De er begge vigtige for at man kan managing data og videreføre information.

State er et lokalt bibliotek som man altid kan ændre i via. komponentet.

Props kan kun læses og kan ikke rettes i.. Det er en envejskommunikation fra parent to child.

7. Hvad er hooks i React?

Hooks skal importeres fra react.

Hooks gør at man får adgang til React features..

```
Fx import React, { useState } from "react";
```

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

useState:

en tilstand

useState fungere sådan at man importere det først fra React. Derefter kan man bruge denne hook.

I dette tilfælde kan man skifte mellem farver og finde sin yndlingsfarve. Det er en funktion i react man kan benytte sig af

```
import {useState} from "react";
function FavoritColor (){
  Const [color, setColor] = useState("");
}
```

useEffect:

En Effect man kan bruge.

Det er fx til at fetche data, opdatering af DOM, og en timer

```
import { useState, useEffect } from "react";
import ReactDOM from "react-dom/client";

function Timer() {
  const [count, setCount] = useState(0);

  useEffect(() => {
    setTimeout(() => {
      setCount((count) => count + 1);
    }, 1000);
  });

  return <h1>I've rendered {count} times!</h1>;
}

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(<Timer />);
```

9. Hvad er JSX?

JSX gør det muligt for os at skrive HTML nemmere

const myElement = <h1>I Love JSX!</h1>; (MED JSX)

const myElement = React.createElement('h1', {}, 'I do not use JSX!'); (UDEN)

10. Hvordan adskiller JSX sig fra almindelig JavaScript?

Man kan ikke skrive HTML i alm. javascript medmindre det er JSX så kan man godt

Så JSX gør det nemmere for os at kombinere begge.

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

Console.log gør det muligt at se den kode man har lavet i browseren og om den fungerer

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

For at se om den enkelte kode fungerer.. det er godt at bruge når man skal finde det præcise sted koden gik galt. Så Console logger man bare