

Teorihandbok Jessika Wiberg

1. React är ett JavaScript-bibliotek för att bygga användargränssnitt. Det använder en virtuell DOM samt är uppbyggt baserat på komponenter. Fördelarna kan vara återanvändbarhet, uppdateringar som är effektiva genom virtuell DOM och optimering för prestanda. Det är även enkelt att integrera react med andra bibliotek och ramverk. Nackdelarna kan vara inlärningskurvan och komplexiteten hos verktygen som används.
2. Rendering är processen där en webbsida eller användargränssnitt genereras och visas för användaren. Den virtuella DOM:en är en enklare kopia av den faktiska DOM:en som används i React för att optimera uppdateringar och förbättra prestandan.
3. JSX står för JavaScript XML som är en förlängning som används i React för att skriva HTML-liknande kod inom JavaScript. Det förenklar definitionen av komponenter och rendering. JSX hjälper till och underlättar när man skapar komponenter och rendering i React.
4. En exception, det vill säga ett undantag, är en oväntad händelse eller felaktigt tillstånd som stör den normala körningen av ett program. Den är ett verktyg för att hantera fel och kontrollera flödet, då detta avbryts kontrolleras det av undantagshanteringen för att hantera felet och därmed vidta lämplig åtgärd.
5. Autentisering i webbapplikationer syftar till att verifiera användarens identitet. Den används för att bevilja eller neka åtkomst till specifika resurser eller funktioner baserat på användarens auktorisering, såsom användarnamn och lösenord, och att bara den användare som är auktoriserad får tillgång till känslig information.

I App.jsx importerar jag React, useState och styled-komponenter, tillsammans med flera komponenter som Navbar, Footer, Hero och Card. Jag importerar också assets som bilder. Dessutom importerar jag React Router-komponenterna för att hantera routing.

Koden definierar några komponenter med hjälp av styled-components-biblioteket. Dessa komponenter inkluderar tex MyButton, GridContainer, StyledFooter, LogoContainer, StyledImg och MainContainer. Dessa komponenter används för att applicera anpassad styling på elementen i min kod. I min kod definieras olika komponenter, till exempel Navbar, Footer, Hero och Card, och renderar dem i lämpliga delar av applikationen. När dessa komponenter renderas skapas det som användaren ser.

Komponenten App.jsx är huvudkomponenten i din applikation. Den sätter upp routningen med hjälp av React Router och renderar komponenterna baserat på de definierade routerna. Den använder GridContainer-komponenten för att strukturera layouten i applikationen. LogoContainer-komponenten visar en bild på en logotyp med hjälp av StyledImg-komponenten och Navbar-komponenten renderas också. I min kod använder jag React-komponenter och förklarande/deklarativ syntax för att bygga användargränssnittet. När dessa komponenter renderas och skapas så uppdateras den virtuella DOM:en och därefter appliceras uppdateringarna på den faktiska DOM:en. Detta innebär att jag drar nytta av Reacts DOM-koncept för att optimera rendering och uppdateringar i min applikation.

Inuti MainContainer-komponenten har jag en Switch-komponent som hanterar de olika routerna. Beroende på den aktuella routern renderas olika innehåll, som "Om oss", "Portfölj" eller "Hem", som inte är färdigbyggt. Det finns också en rubrik och två Card-komponenter (Card och CardTwo) som renderas. Card-komponenten visar en container med en bild som heter tex StyledPool lite textinnehåll. Den inkluderar också en styled knapp kallad StyledButton.

Sammanfattningsvis sätter koden upp en grundläggande React-applikation med routning, stiliserade komponenter och rendering av olika komponenter baserat på de definierade routerna.