

**React/Next.js の最新トレンドに
少しだけ触れてみる**

本日のメニュー

- React とは何か
- Next.js とは何か
- React の課題について
- React Server Components が解決する React の課題
- Next.js の App Router について
- React Server Components と App Router のデモ

本日のゴール

React Server Components と App Router をふんわり理解している



React とは何か

- React
 - UI を簡単に構築するための JavaScript ライブラリ
 - コンポーネントという概念を使ってパーツを組み立てる感覚で画面を構築できる



Next.js とは何か

- Next.js
 - React のフレームワーク
 - React の機能を拡張してより使いやすくしたもの
 - ex. ルーティングや SSR など



React の課題について

- クライアント側でアプリの全ての JavaScript をレンダリングする
- クライアント側のパフォーマンス悪化が懸念

No Pre-rendering (Plain React.js app)

Initial Load:

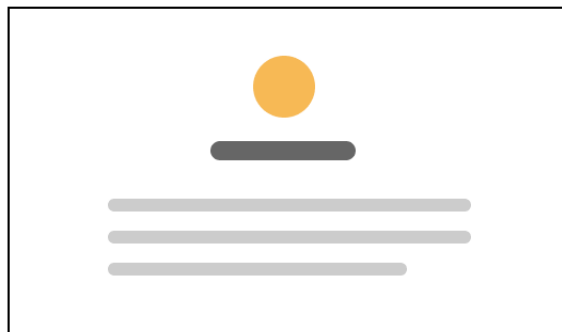
App is not rendered



JS loads
→

Hydration:

React components are initialized and App becomes interactive



そこで

React Server Components が登場!

React Server Components(RSC)とは

- コンポーネントを「クライアント側でレンダリングされるコンポーネント」と「サーバー側でレンダリングされるコンポーネント」に分ける React の新機能
- クライアントに送信される JS の量（クライアント上でレンダリングされるコード量）が減るため、パフォーマンスの向上が期待されている

要は

今までクライアント側でしか実行できなかった **React** が、サーバー側でも実行できるようになったということ

React Server Components(RSC)のレンダリングの流れ

1. サーバー側でサーバーコンポーネントをレンダリングする
2. サーバーコンポーネントの HTML とクライアントコンポーネントの JavaScript をクライアントに送信する
3. クライアントコンポーネントをレンダリングする
4. 生成した HTML を DOM に反映させてクライアント側で表示する

React Server Components(RSC)のレンダリングの流れ

1. サーバー側でサーバーコンポーネントをレンダリングする
 2. サーバーコンポーネントの HTML と クライアントコンポーネントの *JavaScript* をクライアントに送信する
 3. クライアント側でクライアントコンポーネントをレンダリングする
 4. 生成した HTML を DOM に反映させてクライアント側で表示する
- > サーバーコンポーネントの分、クライアントの負担が減る

Next.js の App Router について

- Next.js には二つのモードがある
 - Pages Router と App Router
- App Router が現在推奨されているモード
- App Router では、React Server Components が採用されている
- デフォルトだと、実装したコンポーネントはサーバーコンポーネントになる
- クライアントコンポーネントにする場合は、`'use client'` を記述する必要がある

つまり

App Router を使うことで、React Server Components を簡単に実装できる

React Server Components と App Router のデモ

まとめ

- React では、クライアント側の負担の増加が課題だった
 - アプリを丸ごとクライアント側で構築するのは、パフォーマンス的に厳しい
- そこで登場したのが React Server Components
 - コンポーネントを「サーバーコンポーネント」と「クライアントコンポーネント」に分ける
 - クライアント側に送信する JS の量を減らすことに成功
- Next.js の App Router は React Server Components がベースになっている

主な資料

- [一言で理解する React Server Components](#)
- [Next.js 公式ドキュメント](#)
- [What's "Next" JS Meetup](#)
- [Nextjs で理解する React Server Components 徹底解説【React18】](#)

