

이미지 불러오기에 실패했을 때 처리하기

[Image](#) 컴포넌트를 사용해서 이미지를 로드하다가 문제가 발생하면, `onError` 콜백이 호출돼요. 이 `onError` 콜백으로 에러를 처리할 수 있어요.

이 문서에서는 이미지 로드 실패했을 때 에러를 감지하고, 사용자에게 에러 상태를 시각적으로 알리는 방법을 설명할게요.

이미지를 불러오다가 네트워크 문제나 잘못된 URL 등으로 인해 이미지를 불러오지 못할 수 있어요. 이때 `onError` 콜백 함수가 호출돼요. 이 콜백을 사용하면 이미지를 불러오는 데 실패했다는 메시지를 콘솔에 출력할 수 있어요.

아래는 에러가 발생했을 때, 빨간색 테두리를 추가한 `View`를 표시하는 코드예요. `onError` 콜백에서 에러를 감지하고, `hasError` 상태를 업데이트해 에러가 발생했을 때 다른 UI를 보여주는 거죠.

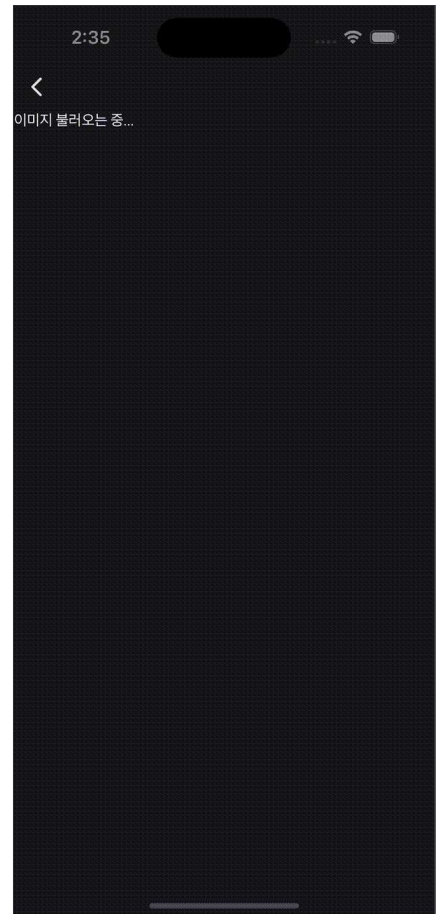
```
import { useState } from "react";
import { View } from "react-native";
import { Image, BedrockRoute } from "react-native-

export const Route = BedrockRoute("/", {
  component: Index,
});

function Index() {
  const [hasError, setHasError] = useState(false);

  return (
    <View>
      {hasError === true ? (
        <ErrorView />
      ) : (
        <Image
```

tsx



```
        style={{
          width: 100,
          height: 100,
        }}
        source={{
          uri: "invalid url", // 잘못된 URL을 사:
        }}
        onError={() => {
          Alert.alert("이미지 에러");
          setHasError(true);
        }}
      />
    )}
  </View>
);
}

/** 임의의 에러 뷰 */
function ErrorView() {
  return (
    <View
      style={{
        width: 100,
        height: 100,
        borderColor: "red",
        borderWidth: 1,
      }}
    />
  );
}
```

레퍼런스

- [Image 컴포넌트](#)

