

## Atividade 1

### Criando um Componente Reutilizável com Props

Crie um componente `ProfileCard` que exibe um nome de usuário, uma biografia e uma imagem de perfil, recebendo esses dados como props. Em seguida, reutilize o componente para exibir diferentes perfis.

#### Exemplo:

```
const ProfileCard = ({ name, bio, imageUrl }) => (
  <div className="profile-card">
    <img src={imageUrl} alt={`${name} profile`} />
    <h2>{name}</h2>
    <p>{bio}</p>
  </div>
);

// Uso do componente
<ProfileCard
  name="Ana Silva"
  bio="Desenvolvedora Frontend apaixonada por React"
  imageUrl="https://exemplo.com/ana.jpg"
/>
```

## Atividade 2

### Implementando um Higher-Order Component (HOC) de Autorização

Crie um HOC chamado `withPermission` que verifique se o usuário tem permissão para ver o conteúdo.

**Exemplo:**

```
const withPermission = (WrappedComponent) => {
  return (props) => {
    if (!props.hasPermission) {
      return <h1>Permissão negada</h1>;
    }
    return <WrappedComponent {...props} />;
  };
};

// Componente que precisa de permissão
const Settings = () => <div>Página de Configurações</div>;

// Uso do HOC
const ProtectedSettings = withPermission(Settings);
<ProtectedSettings hasPermission={true} />
```