

INTRO



```
export const getUserWithPosts = (userID: string) =>  
  
  get(`/api/users/${userID}`).then(user =>  
  
    get(user.posts_url).then(posts => ({ ...user.data, posts: posts.data })),  
  
  );
```

```
users(id: 43) {  
  email,  
  username,  
  posts {  
    title,  
    content  
  }  
}
```

GRAPHQL

REST

```
export const getUserWithPosts = (userID: string) =>
  get(`/api/users/${userID}`).then(user =>
    get(user.posts_url).then(posts => ({ ...user.data, posts: posts.data })),
  );
```

GRAPHQL

```
users(id: 43) {
  email,
  username,
  posts {
    title,
    content
  }
}
```



```

buttons,
firstActive,
fullWidth,
style,
) => {
  const styles = {
    active: {
      backgroundColor: cssVariables['--color-grey-700'],
      borderColor: cssVariables['--color-grey-700'],
      color: '#FFF',
    },
    button: {
      flex: fullWidth ? 1 : 'unset',
    },
  };

  const activeStyles = { ...styles.active, ...styles.button };

  return (
    <div className={segmentControl.buttonContainer} style={styl
      <Button
        to={buttons[0].to}
        onClick={buttons[0].onClick}
        className={segmentControl.button}
        style={firstActive ? activeStyles : styles.button}
      >
        {buttons[0].text}
      </Button>
      <Button
        to={buttons[1].to}
        onClick={buttons[1].onClick}
        className={segmentControl.button}
        style={!firstActive ? activeStyles : styles.button}
      >
        {buttons[1].text}
      </Button>
    </div>
  );

```

DECLARATIVE

MAKING THINGS EASIER ON YOURSELF