

Bridge

دیزاین پترن هادر React

Structural Patterns

به زبان ساده همراه با مثال های عملی
کاربرد دیزاین پترن ها برای برنامه نویسان React



Structural Patterns

Bridge



الگوی Bridge برای جدا کردن انتزاع از پیاده سازی به کار می رود، به طوری که این دو بتوانند مستقل از یکدیگر تغییر کنند. این الگو زمانی مفید است که بخواهیم منطق یک کامپوننت را از جزئیات پیاده سازی آن (مانند تم ها) جدا کنیم.

فرض کنید می خواهیم یک کامپوننت مانند Button داشته باشیم که از چند تم مختلف پشتیبانی کند. به جای اینکه منطق تم را در خود کامپوننت ها قرار دهیم، می توانیم از این الگو برای جدا کردن کامپوننت ها (انتزاع) از تم ها (پیاده سازی) استفاده کنیم.

بخش ButtonThemeType

یک Type را به شکل زیر با Property هایی که مشاهده می کنید، می سازیم.

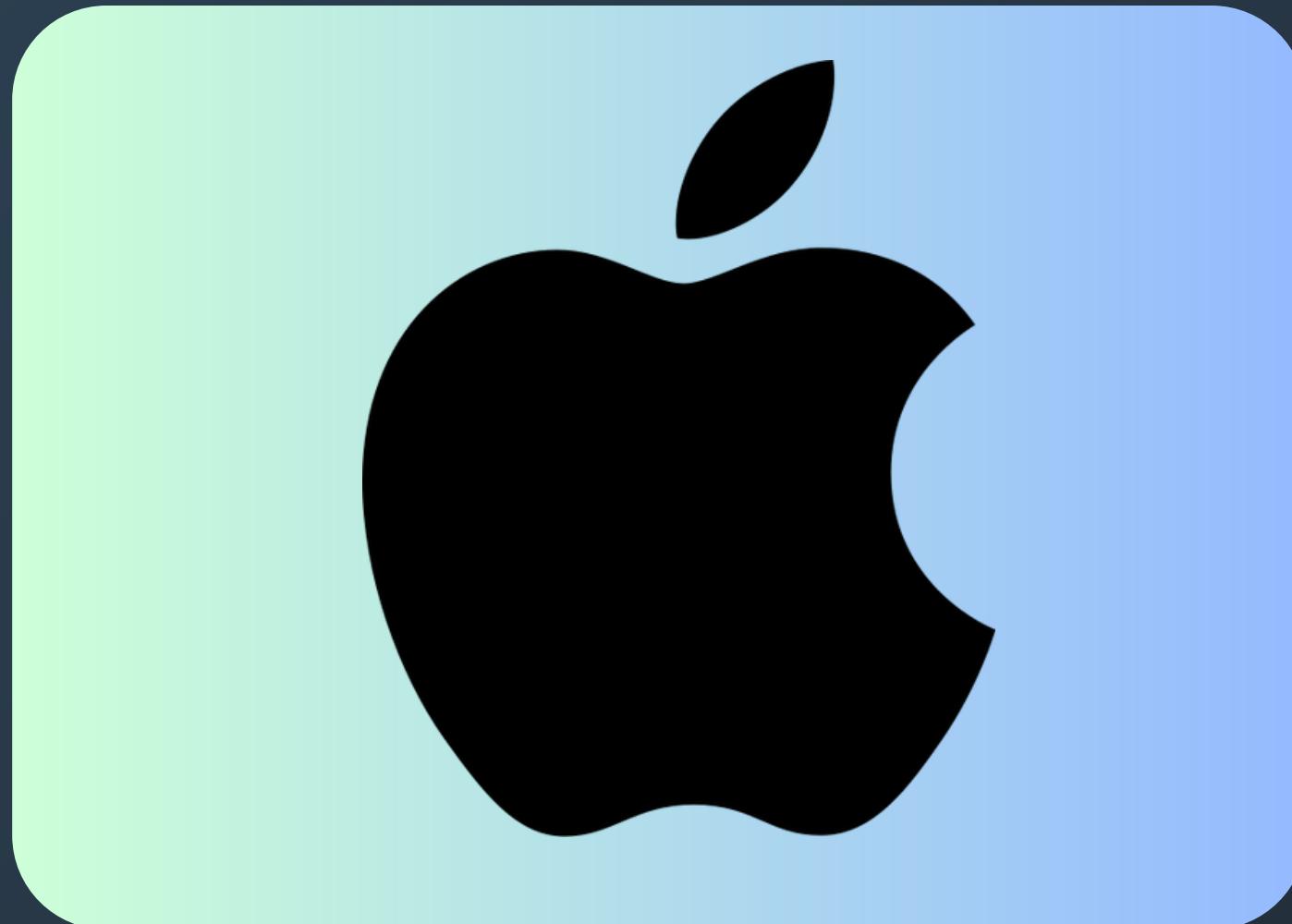


ButtonThemeType

```
● ● ●  
export type ButtonThemeType = {  
  color: string;  
  background: string;  
  borderRadius: string;  
  padding: string;  
}
```

ایجاد Theme های متفاوت

در این بخش می خواهیم دو Theme متفاوت از Apple و Google را پیاده سازی کنیم.
در ادامه به راحتی می توانیم با هر یک از این Button ها Theme خود را بسازیم.



Human Interface Guidelines



Material Design

ایجاد مربوط به Apple Theme

اولین Theme را که نام دارد زیر شکل این Human Interface Guidelines می سازیم.



HIGButtonTheme

```
import { ButtonThemeType } from "../Button/button-theme.interface";

export const HIGButtonTheme: ButtonThemeType = {
  color: '#fff',
  background: "#007aff",
  borderRadius: '12px',
  padding: '10px 20px',
}
```

ایجاد Google Theme مربوط به

دومین نام دارد را به شکل زیر می سازیم.



MaterialButtonTheme

```
import { ButtonThemeType } from "../Button/button-theme.interface";

export const MaterialButtonTheme: ButtonThemeType = {
  color: '#fff',
  background: "#6200ee",
  borderRadius: '4px',
  padding: '8px 16px',
}
```

بخش ButtonProps

در این بخش Props مربوط به Button را می سازیم.



ButtonProps

```
import { ButtonThemeType } from "./button-theme.interface";

export interface ButtonProps {
  theme: ButtonThemeType;
  label: string;
}
```

بخش Button

در این بخش Component مد نظر خود را پیاده سازی می کنیم.



```
import { ButtonProps } from "./Button.props";

const Button: React.FC<ButtonProps> = ({ theme, label }) => {
  return (
    <button
      style={{
        color: theme.color,
        backgroundColor: theme.background,
        borderRadius: theme.borderRadius,
        padding: theme.padding,
        border: 'none',
        cursor: 'pointer',
      }}
    >
      {label}
    </button>
  );
};

export default Button;
```

طرز استفاده از Button

حال می توانیم به شکل زیر هر یک از Theme های موجود را به Button بدهیم.



BridgeDesignPattern

```
import Button from "./pattern/Button/Button";
import { HIGButtonTheme } from "./pattern/design-systems/hig.theme";
import { MaterialButtonTheme } from "./pattern/design-systems/material.theme";

const BridgeDesignPattern = () => {
  return (
    <div>
      <Button
        theme={HIGButtonTheme}
        label="Human Interface Guidelines Button"
      />
      <Button
        theme={MaterialButtonTheme}
        label="Material Design Button"
      />
    </div>
  );
};

export default BridgeDesignPattern;
```