

[UIKit](#) / [...](#) / [UIButton.Configuration](#) / `prominentClearGlass()`

Type Method

prominentClearGlass()

Creates a configuration for a button that has a prominent, clear Liquid Glass style.

iOS 26.0+ | iPadOS 26.0+ | Mac Catalyst | tvOS 26.0+

```
static func prominentClearGlass() -> UIButton.Configuration
```

Discussion

In tvOS, this button style applies a Liquid Glass effect regardless of whether the button has focus.

See Also

Creating configurations

```
static func plain() -> UIButton.Configuration
```

Creates a configuration for a button with a transparent background.

```
static func gray() -> UIButton.Configuration
```

Creates a configuration for a button with a gray background.

```
static func tinted() -> UIButton.Configuration
```

Creates a configuration for a button with a tinted background color.

```
static func filled() -> UIButton.Configuration
```

Creates a configuration for a button with a background filled with the button's tint color.

```
static func borderless() -> UIButton.Configuration
```

Creates a configuration for a button that has a borderless style.

```
static func bordered() -> UIButton.Configuration
```

Creates a configuration for a button that has a bordered style.

```
static func borderedTinted() -> UIButton.Configuration
```

Creates a configuration for a button that has a tinted, bordered style.

```
static func borderedProminent() -> UIButton.Configuration
```

Creates a configuration for a button that has a prominent, bordered style.

```
static func glass() -> UIButton.Configuration
```

Creates a configuration for a button that has a Liquid Glass style.

```
static func prominentGlass() -> UIButton.Configuration
```

Creates a configuration for a button that has a prominent Liquid Glass style.

```
static func clearGlass() -> UIButton.Configuration
```

Creates a configuration for a button that has a clear Liquid Glass style.

```
func updated(for: UIButton) -> UIButton.Configuration
```

Returns a copy of the configuration, updated for the given button.