

## Class: TouchBarButton

Create a button in the touch bar for native macOS applications

Process: Main *This class is not exported from the 'electron' module. It is only available as a return value of other methods in the Electron API.*

**new TouchBarButton(options)**

- **options** Object
  - **label** string (optional) - Button text.
  - **accessibilityLabel** string (optional) - A short description of the button for use by screenreaders like VoiceOver.
  - **backgroundColor** string (optional) - Button background color in hex format, i.e #ABCDEF.
  - **icon** NativeImage | string (optional) - Button icon.
  - **iconPosition** string (optional) - Can be **left**, **right** or **overlay**. Defaults to **overlay**.
  - **click** Function (optional) - Function to call when the button is clicked.
  - **enabled** boolean (optional) - Whether the button is in an enabled state. Default is **true**.

When defining **accessibilityLabel**, ensure you have considered macOS best practices.

### Instance Properties

The following properties are available on instances of **TouchBarButton**:

**touchBarButton.accessibilityLabel** A **string** representing the description of the button to be read by a screen reader. Will only be read by screen readers if no label is set.

**touchBarButton.label** A **string** representing the button's current text. Changing this value immediately updates the button in the touch bar.

**touchBarButton.backgroundColor** A **string** hex code representing the button's current background color. Changing this value immediately updates the button in the touch bar.

**touchBarButton.icon** A **NativeImage** representing the button's current icon. Changing this value immediately updates the button in the touch bar.

**touchBarButton.iconPosition** A **string** - Can be **left**, **right** or **overlay**. Defaults to **overlay**.

**touchBarButton.enabled** A boolean representing whether the button is in an enabled state.