

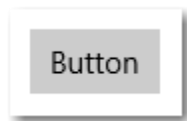
Chapter 2: Desktop Application

UWP apps (textbox, button, label)

UWP apps automatically adjust the size of controls, fonts, and other UI elements so that they are legible on all devices.

When your app runs on a device, the system uses an algorithm to normalize the way UI elements display on the screen. This scaling algorithm takes into account viewing distance and screen density (pixels per inch) to optimize for perceived size (rather than physical size).

A button gives the user a way to trigger an immediate action.



Occurs when a button control is clicked.

Syntax

C#

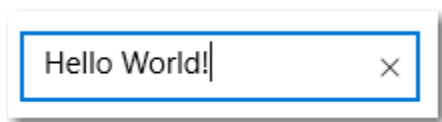
`public event RoutedEventHandler Click`

A label is the name or title of a control or a group of related controls.

- Use a label to indicate to the user what they should enter into an adjacent control. You can also label a group of related controls, or display instructional text near a group of related controls.
- When labeling controls, write the label as a noun or a concise noun phrase, not as a sentence, and not as instructional text. Avoid colons or other punctuation.

- When you do have instructional text in a label, you can be more generous with text-string length and also use punctuation.
- The TextBox control lets a user type text into an app. It's typically used to capture a single line of text, but can be configured to capture multiple lines of text. The text displays on the screen in a simple, uniform, plaintext format.
- TextBox has a number of features that can simplify text entry. It comes with a familiar, built-in context menu with support for copying and pasting text. The "clear all" button lets a user quickly delete all text that has been entered. It also has spell checking capabilities built in and enabled by default.

Examples



Create a text box

Here's the XAML for a simple text box with a header and placeholder text.

```
<TextBox Width="500" Header="Notes" PlaceholderText="Type your notes here"/>
```