Xamarin

1. Label

* In Xamarin.Forms, a Label is a UI control used to display a read-only text. It is a simple control that allows you to present information or instructions to the user within your mobile application.
* Attributes of Label:
  + Text: The **Text** property of the Label is used to set the text content that will be displayed.
  + Text Formatting: The **FormattedText** property allows you to apply formatting to different parts of the text. You can use spans to style specific portions of the text differently.
  + Font Properties: You can set properties such as **FontAttributes**, **FontSize**, and **FontFamily** to control the appearance of the text.
  + Color Properties: **TextColor** property allows you to set the color of the text.
  + Alignment: You can control the horizontal and vertical alignment of the text using the **HorizontalTextAlignment** and **VerticalTextAlignment** properties.
* Simple use of Label

A computer screen with text and images

Description automatically generated

A screen shot of a computer

Description automatically generated

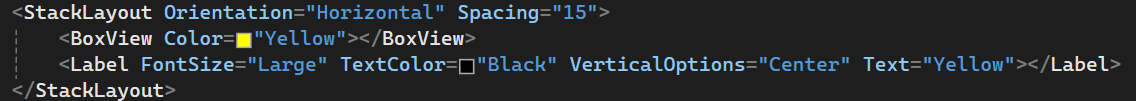
* Use Label with FormattedText

A screen shot of a computer program

Description automatically generated

1. StackLayout

* By Default is vertical, but we can change orientation



* You can also set spacing between those widgets inside the StackLayout by using Spacing



1. Put this command line in the layout xaml file for IOS only

xmlns:ios="clr-namespace:Xamarin.Forms.PlatformConfiguration.iOSSpecific;assembly=Xamarin.Forms.Core"  
ios:Page.UseSafeArea="True"

A screenshot of a computer screen

Description automatically generated

1. Frame:

* In Xamarin.Forms, a Frame is a visual container used to group and provide a visual distinction for a set of controls or content. It adds a border and shadow to its content, making it stand out on the user interface. The Frame can contain any Xamarin.Forms visual element, such as layouts, labels, buttons, and more.
* By default, the Frame has its own padding which is 20 device-independent pixels (dp or DIP), but you can also set for it by using Padding



* Example:

A computer screen shot of a program

Description automatically generated

A screenshot of a computer

Description automatically generated

If you don’t set Padding = 5 for frame

A white rectangular frame with a black border

Description automatically generated

1. Button

* In Xamarin.Forms, a Button is a UI control that allows users to trigger an action when tapped. It is a common element in user interfaces for initiating actions or navigating between pages in a mobile application.
* Attributes of Button:
  + Text Property: The **Text** property of the Button is used to set the text that will be displayed on the button.
  + Command Property: The **Command** property allows you to bind a command from your view model, enabling you to handle the button click in a more structured way.
  + Click Event: You can also handle the button click by subscribing to the **Clicked** event.
  + Styling: You can apply styles to the Button using properties like **BackgroundColor**, **TextColor**, **BorderRadius**
* Define button in xaml



* + Attributes of button which are Text “Click me”, its background color beige, radius of corner is 5
* In the example above, there is attribute “Clicked” which is the action triggered a method named Button\_Clicked when clicking on that button
  + You can define that function in the file xaml.cs of that file xaml



A computer screen shot of a program code

Description automatically generated

1. Entry



x:Name is like id in Android studio or name of tag in html which is used to present that widget in back-end

* In Xamarin.Forms, an Entry is a UI control that allows users to enter and edit text. It's commonly used for capturing user input, such as usernames, passwords, search queries, and more. The entered text can be used for various purposes within the application, such as data entry, searching, or filtering.
* Attributes of Entry:
  + Text Property: The **Text** property of the Entry is used to get or set the text content entered by the user.
  + Placeholder Text: The **Placeholder** property allows you to set a hint or placeholder text that is displayed when the Entry is empty, providing guidance to the user.
  + Keyboard Type: The **Keyboard** property allows you to specify the type of keyboard that appears when the Entry is focused. For example, you can use Numeric, Email, Phone, etc.
  + Text Alignment: The **HorizontalTextAlignment** property allows you to control the horizontal alignment of the text within the Entry.
  + Events: You can handle events like **Completed** when the user finishes entering text.
  + Example:

A black screen with white text

Description automatically generated

1. Units of Xamarin:

* Xamarin.Forms uses device-independent pixels (dp or DIP) as the unit for all measurements of widgets and layout properties. Device-independent pixels provide a consistent way to specify sizes and distances across different devices with varying screen densities.
* When you set properties like WidthRequest, HeightRequest, Margin, Padding, and other layout-related properties in Xamarin.Forms, you are working with device-independent pixels. Xamarin.Forms automatically scales these values based on the screen density of the device, ensuring a consistent layout across devices with different screen resolutions.
* Using device-independent pixels is a good practice in Xamarin.Forms development as it helps in creating responsive user interfaces that adapt well to various screen sizes and densities. It abstracts the physical pixel density of the device, providing a more predictable and maintainable layout experience.

1. Padding = “10” or Margin = “10” means that Left, Top, Right, Bottom is 10. However, if Padding = “0,10,12,20” or Margin = “0,10,12,20” means that Left = 0, Top = 10, Right = 12, Bottom = 20

A screenshot of a computer

Description automatically generated

1. Make sure that under <ContentPage.Content> includes only one layout which can be StackLayout or Grid,…

A screen shot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated