# Using UI

# RectTransform

* UI implementation of Transform. Position, size, anchor and pivot information for a rectangle.
* Anchors - they define the behavior of an UI item upon aspect ratio or resolution changes
* Position - Deepening on the anchors defines the position

# Canvas

* Draw order - UI elements in the Canvas are drawn in the same order they appear in the Hierarchy. You can alter this at runtime using Transform.SetSunlingIndex();
* Render Modes - The Canvas has a Render Mode setting which can be used to make it render in screen space or world space.
  + Screen Space - Overlay - This render mode places UI elements on the screen rendered on top of the scene. If the screen is resized or changes resolution, the Canvas will automatically change size to match this.
  + Screen Space - Camera - This is similar to Screen Space - Overlay, but in this render mode the Canvas is placed a given distance in front of a specified Camera. The UI elements are rendered by this camera, which means that the Camera settings affect the appearance of the UI. If the Camera is set to Perspective, the UI elements will be rendered with perspective, and the amount of perspective distortion can be controlled by the Camera Field of View. If the screen is resized, changes resolution, or the camera frustum changes, the Canvas will automatically change size to match as well.
  + World Space - In this render mode, the Canvas will behave as any other object in the scene. The size of the Canvas can be set manually using its Rect Transform, and UI elements will render in front of or behind other objects in the scene based on 3D placement.
* Pixel Perfect - Force elements in the canvas to be aligned with pixels. Only applies with renderMode is Screen Space. It makes things look a bit sharper, but kinda heavy especially when moving. Good strategy is to disable pixel perfect, move the element and then re-enable it.
* Sort order - you can use it manage which canvas is shown on top or behind other canvases
* Target Display - For Overlay mode - determines the monitor to use
* Additional Shader Channels - 90% of the cases it needs to be set to Nothing as it is as Default. Example usage will be if you want the UI to receive light then you enable the Normal and Tangent channels.

# Canvas Scaler

* The Canvas Scaler component is used for controlling the overall scale and pixel density of UI elements in the Canvas. This scaling affects everything under the Canvas, including font sizes and image borders.
* UI Scale Mode - Determines how UI elements in the Canvas are scaled.
  + Constant Pixel Size - Makes UI elements retain the same size in pixels regardless of screen size.
  + Scale With Screen Size - Makes UI elements bigger the bigger the screen is.
  + Constant Physical Size - Makes UI elements retain the same physical size regardless of screen size and resolution.
  + Scale Factor - Scales all UI elements in the Canvas by this factor.
  + Reference Pixels Per Unit - If a sprite has this ‘Pixels Per Unit’ setting, then one pixel in the sprite will cover one unit in the UI.

# Graphic Raycaster

* The Graphic Raycaster is used to raycast against a Canvas. The Raycaster looks at all Graphics on the canvas and determines if any of them have been hit.
* Ignore Reversed Graphics - Should graphics facing away from the raycaster be considered?
* Blocking Objects - Type of objects that will block graphic raycasts.
* Blocking Mask - Type of objects that will block graphic raycasts

# Basic UI Components

* Image
* Button
* Text
* Events and Triggers - Events like OnClick and Triggers when inheriting different Interfaces
* Slider - The Slider control allows the user to select a numeric value from a predetermined range by dragging the mouse.
* Slider - The Slider control allows the user to select a numeric value from a predetermined range by dragging the mouse.
* ScrollRect - Scroll Rect can be used when content that takes up a lot of space needs to be displayed in a small area. The Scroll Rect provides functionality to scroll over this content.
* Scrollbar
* Mask - A Mask is not a visible UI control but rather a way to modify the appearance of a control’s child elements. The mask restricts (ie, “masks”) the child elements to the shape of the parent
* RawImage - just like image but with less properties
* Input Field - A combination of Components that makes up a InputField for the user to enter his Username/password etc...
* Slider - The Slider control allows the user to select a numeric value from a predetermined range by dragging the mouse.
* ScrollRect - Scroll Rect can be used when content that takes up a lot of space needs to be displayed in a small area. The Scroll Rect provides functionality to scroll over this content.
* Mask - A Mask is not a visible UI control but rather a way to modify the appearance of a control’s child elements. The mask restricts (ie, “masks”) the child elements to the shape of the parent
* RawImage - just like image but with less properties
* Input Field - A combination of Components that makes up a InputField for the user to enter his Username/password etc…

# Helper Components

* Vertical Layout Group - The Vertical Layout Group component places its child layout elements on top of each other. Their heights are determined by their respective minimum, preferred, and flexible heights
* Horizontal Layout Group - places its child layout elements next to each other, side by side. Their widths are determined by their respective minimum, preferred, and flexible widths
* Layout Element - If you want to override the minimum, preferred, or flexible size of a layout element, you can do that by adding a Layout Element component to the Game Object.
* Content Size Fitter - functions as a layout controller that controls the size of its own layout element. The size is determined by the minimum or preferred sizes provided by layout element components on the Game Object. Such layout elements can be Image or Text components, layout groups, or a Layout Element component.
* Grid Layout Group - Unlike other layout groups, the Grid Layout Group ignores the minimum, preferred, and flexible size properties of its contained layout elements and instead assigns a fixed size to all of them which is defined with the Cell Size property of the Grid Layout Group itself.