

---

## Quick Start Guide – clicker-item Prefab

---

As part of your quest, you have been asked to create your own design and interpretation of the provided clicker-item prefab.

However, the provided clicker-item prefab is quite an involved piece of UI consisting of many different parts, all of which need to be connected to the ClickerItem.cs script to get things to work (unless you fancy doing some code edits!).

To help you get started, please read the following guide, which will take you through some need-to-know things regarding both the provided clicker-item prefab and the ClickerItem.cs script.

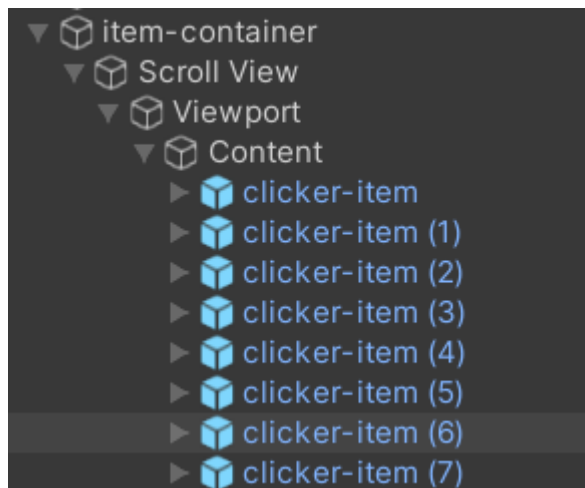
---

## Notes About ClickerItem.cs

---

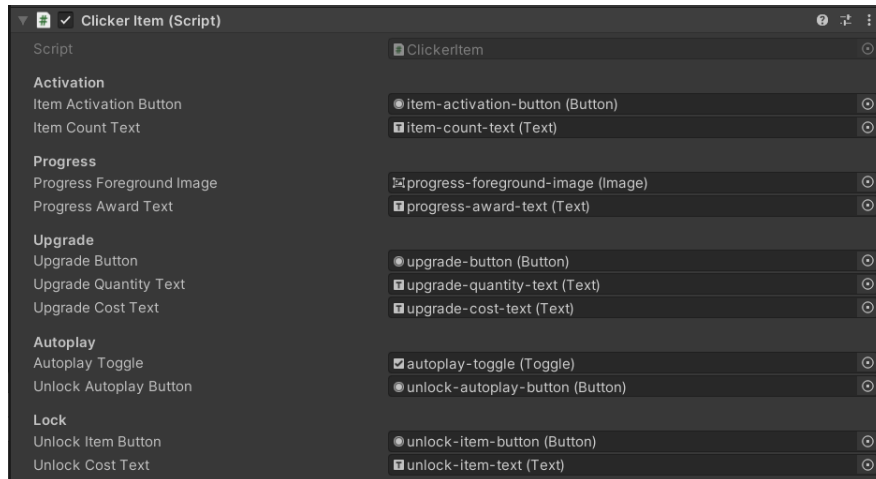
1. The ClickerItem.cs script needs to be attached to every clicker item in the game.
2. It should be attached to the root of your clicker-item prefab to make life a little easier.
3. The script expects all of your clicker-item prefabs to be placed inside their own shared container, with no other elements present.

As an example, here you can see them all sitting in the “Content” folder within a “Scroll View” UI element...

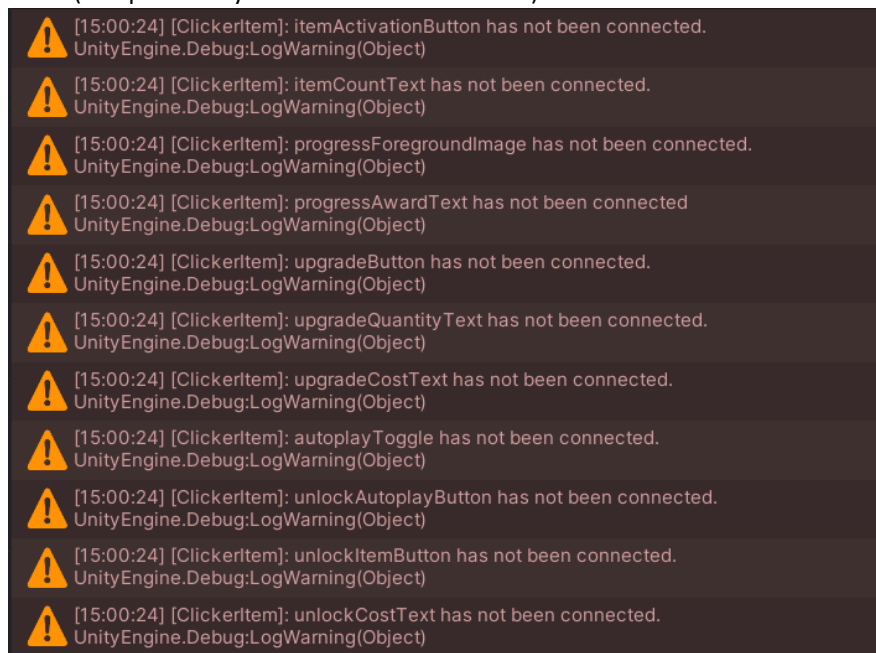


4. Adding multiple clicker-item prefabs to a container will cause their associated values to be changed. This affects things like; unlock cost, upgrade cost, and reward amount.
5. These changes are based on their sibling index within the parent object. You can check the InitValues() method if you fancy playing around with these or just seeing how they work.

6. The WorkRoutine() coroutine is currently missing some code to make the progress bars work, which you will need to fix as part of your quest, so clicking on the item-activation-button will initially do nothing.
7. The script expects a number of fields to be connected for everything to work correctly. You can name these however you like, just make sure they're the correct type of UI element (image, text, button, etc.)



8. If you fail to connect all of the exposed fields then you'll receive a warning in the console to let you know (and potentially some other related errors)

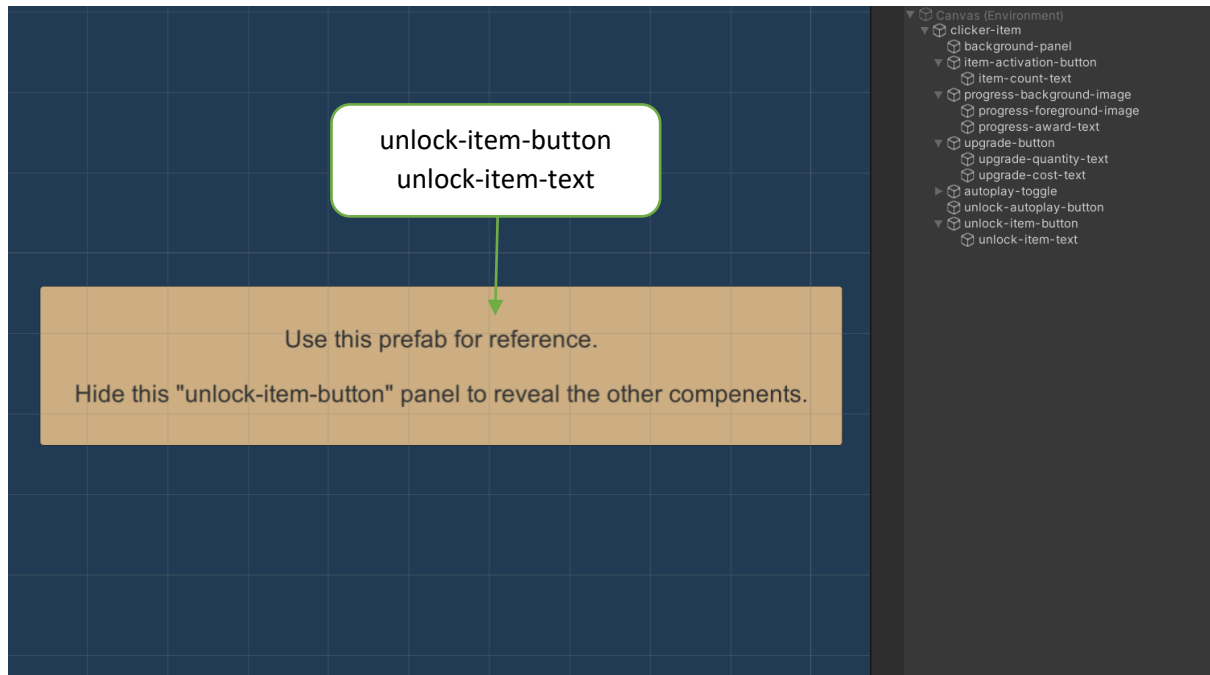


---

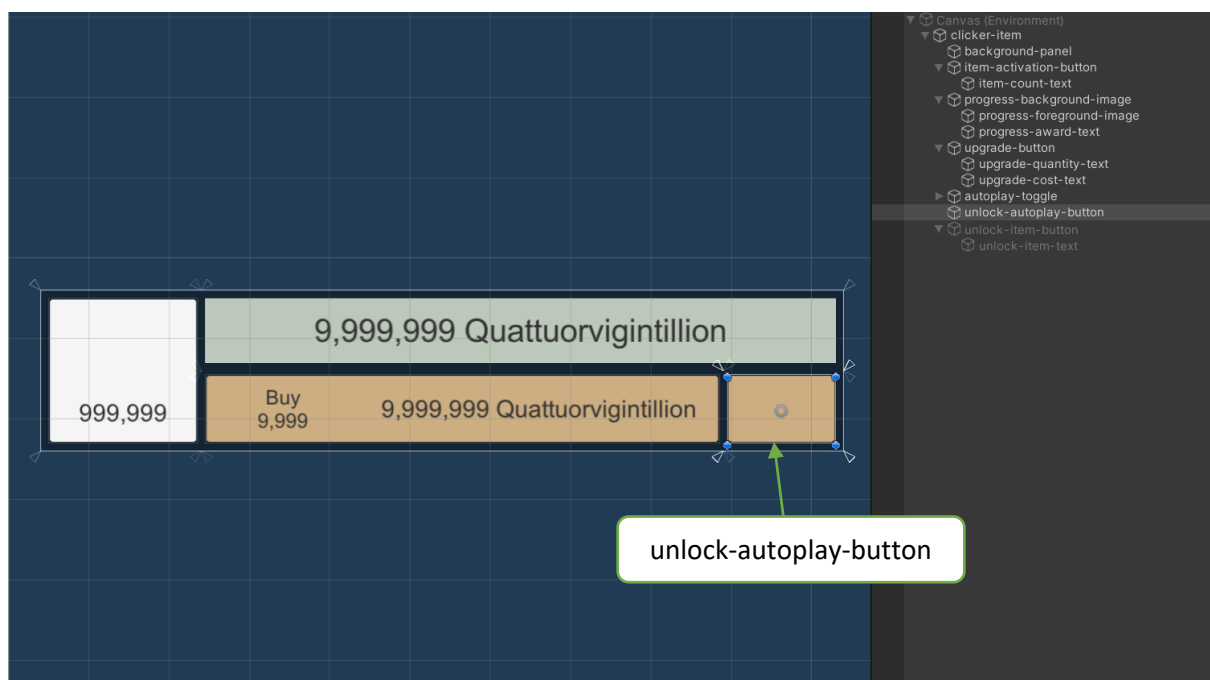
### Notes About the clicker-item Prefab:

---

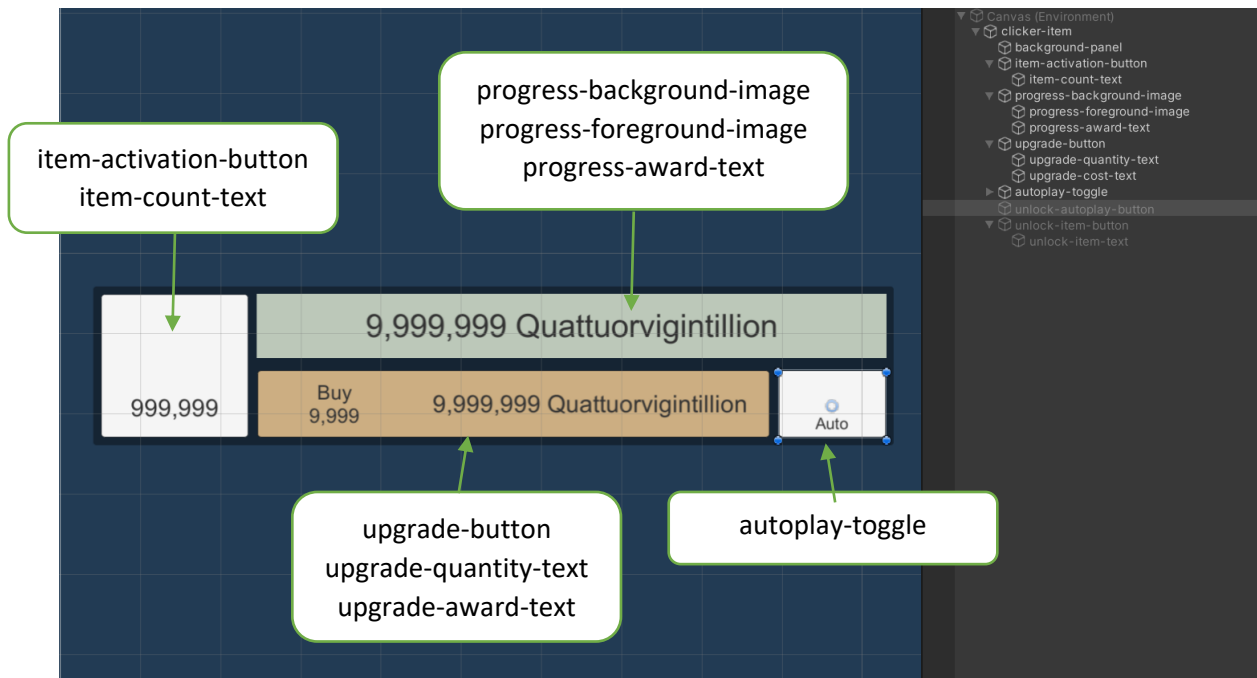
When you first open the prefab, the unlock-item-button will hide all of the other elements.  
In play mode, the unlock-item-text will display the cost for unlocking this item and the player must have enough money in their current balance to unlock it.



Hiding the unlock-item-button will reveal additional elements but the autoplay toggle is still being hidden by the unlock-autoplay-button, since this is a purchasable upgrade.  
Again, the player will need enough money in the bank to unlock this upgrade.



With the unlock-autoplay-button hidden, you can now see all of the final elements



#### Item-activation-button:

Clicking on this triggers ClickerItem.Work()

#### Progress bar (progress-background-image & progress-foreground-image):

This is used to show the progress towards the reward being credited and should fill up over time, once ClickerItem.Work() has been called.

To begin with, this method is incomplete so will do nothing.

#### Upgrade-button:

Clicking on this triggers ClickerItem.PurchaseUpgrade()

This increases the; item count, reward amount to be credited, and cost of the next upgrade.

#### Autoplay-toggle:

Clicking on this triggers ClickerItem.ToggleAutoplay()

This essentially auto-clicks the item-activation-button for you.