

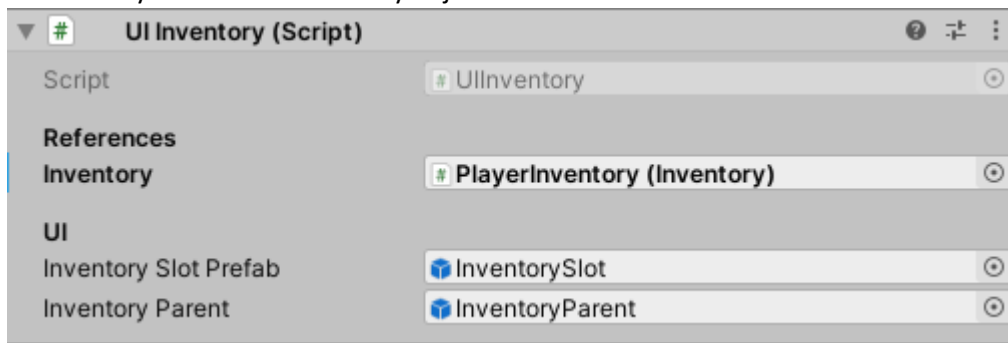
QuickStart Guide:

Functionality:

1. Attach the Inventory.cs script to your desired object (Player, Building, etc)
2. Set your desired properties:



3. Create a UIObject for this inventory → Add the “UIInventory” prefab to the scene.
4. Reference your desired Inventory object to the UI:

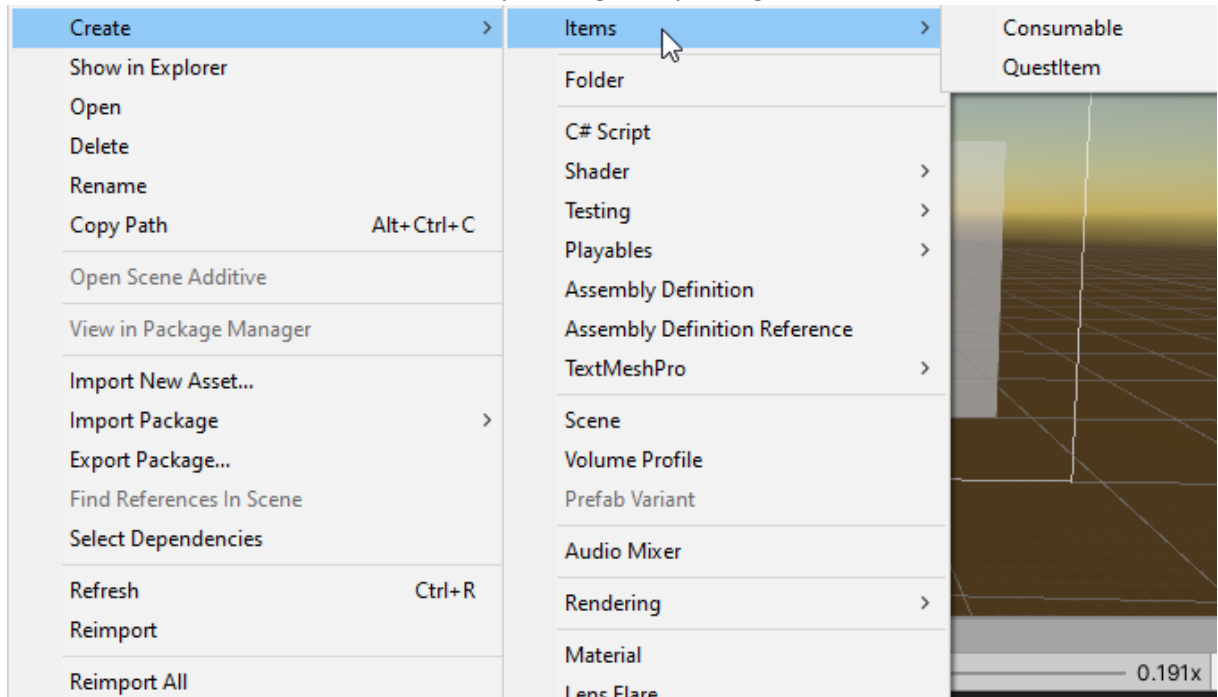


5. You can now create different item types in the “Scripts/Inventory/DataModel” folder by duplicating an existing one (ConsumableItem / QuestItem) and modify its properties to your liking.
6. If you want to customize your use methods of your items, go to the respective item type

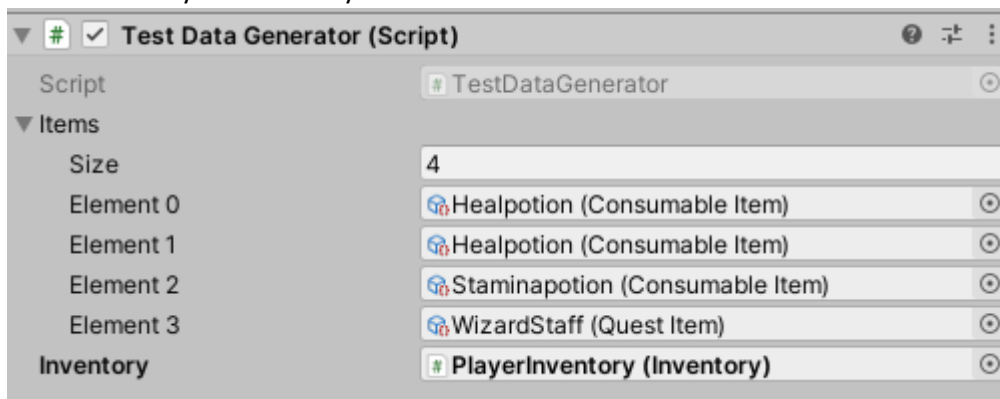
(ConsumableItem / QuestItem) and edit the sample code that is existing there.

```
3 references
public override void Use()
{
    base.Use();
    /* SAMPLE CODE
     * if(HealAmount > 0)
     * {
     *     FindObjectOfType<Player>().Heal(HealAmount);
     * }
     * if(StaminaHealAmount > 0)
     * {
     *     FindObjectOfType<Player>().HealStamina(StaminaHealAmount);
     * }
     */
}
```

7. Create a new Item in the folder "Items" by clicking with your right mouse button:



8. Add new Item icons in the folder "Icons".
9. If you want to test your setup. You can drag the TestDataGenerator prefab into your scene and reference your inventory:



10. If you want to control the way decreasing works, go to "InventorySlot.cs" and change following method:

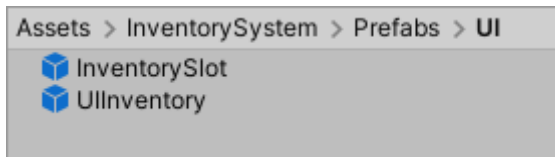
```

1  /// <summary>
2  /// Call the "Use" function on the inventory item, if the item is not null
3  /// </summary>
4  /// reference
5  public void UseItem()
6  {
7      if (this.Item != null)
8      {
9          this.Item.Use();
10
11          //Implement decreasing of items here if needed
12          if (this.Item is ConsumableItem)
13          {
14              Inventory.DecreaseAmount(this.SlotNr);
15          }
16      }
17  }

```

Design:

To change the layout and the design, you can open the “InventorySlot” and the “UIInventory” prefabs and design them accordingly to your needs.



Otherwise you can rebuild the whole prefabs with your own controls and just add the needed scripts and components.