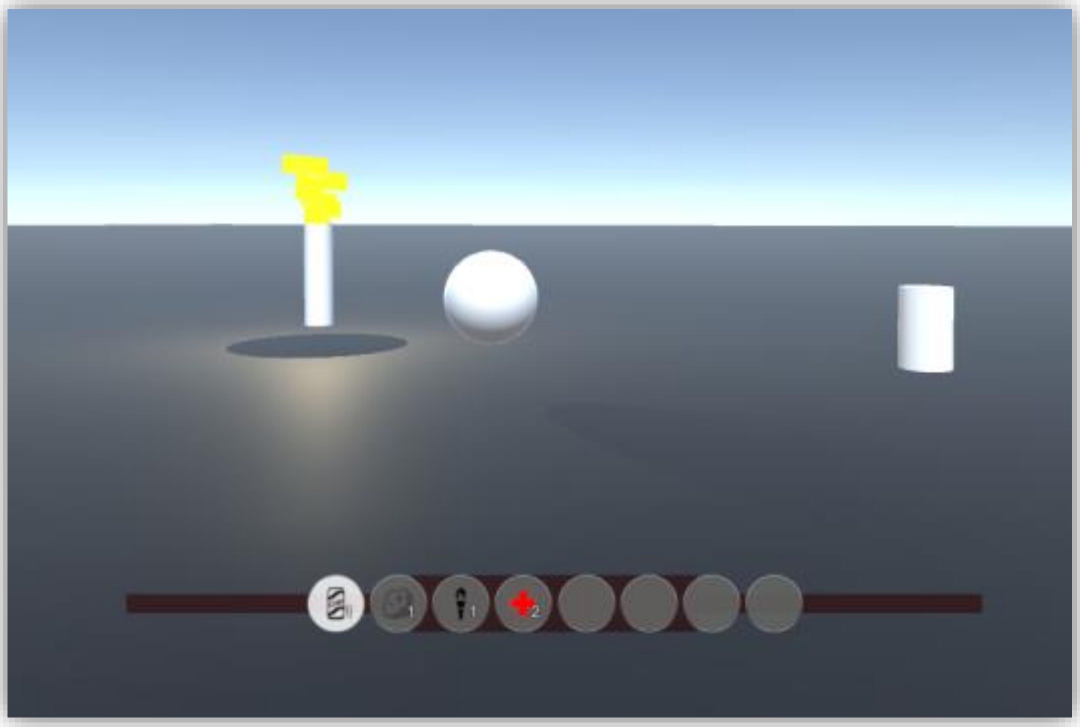
SIMPLE INVENTORY BAR

By Frederic Babord

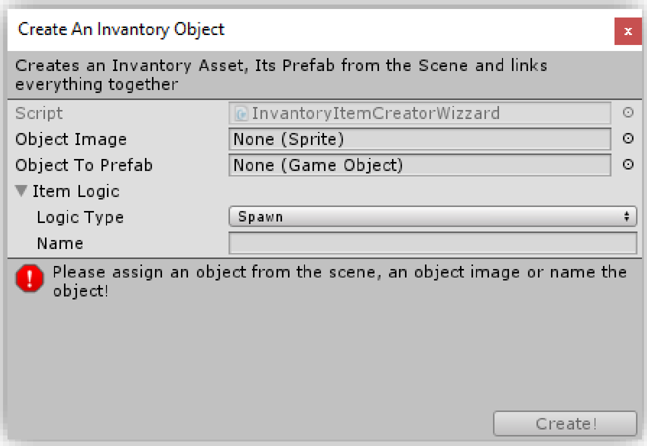


INTEGRATING INTO A CURRENT PROJECT

You will need to drag the Inventory Bar prefab into the scene. Then on your player object add the ***Inventory*** script. The ***Inventory Root*** GameObject is under the Inventory Bar prefab in the scene. Following this, the max number of inventory slots that are made at runtime are set below.

ADDING A NEW OBJECT

Make the object in the scene, so set up the mesh and any properties it's going to need. Then open the inventory asset creator wizard in the Tools > Inventory System menu.



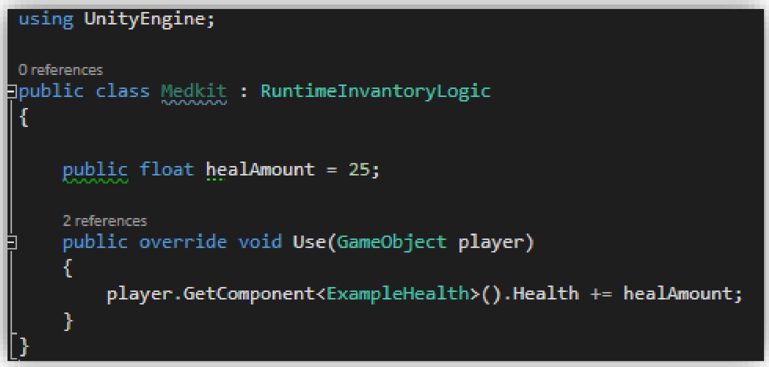
Creates an Invantory Asset, Its Prefab from the Scene and links everything together

Script: InvantoryItemCreatorWizzard
Object Image: None (Sprite)
Object To Prefab: None (Game Object)
▼ Item Logic
Logic Type: Spawn
Name:

Please assign an object from the scene, an object image or name the object!

Attach the image that's going to appear in the inventory bar and the GameObject you made in the scene. If the object only needs to be spawned in the world, then leave the logic type as ***Spawn***, otherwise set it to be ***Other***. Finally give it a sensible name and hit ***Create***.

If the object requires a custom logic type (***Other***), then create a new script which inherits from the ***RuntimeInvantoryLogic*** class. Override the Use function and add your custom logic there. This then needs to be attached to the prefab which was created for you by the inventory object wizard. This is an example of how a med-kit might be used within this system.



```
using UnityEngine;

0 references
public class Medkit : RuntimeInvantoryLogic
{

    public float healAmount = 25;

    2 references
    public override void Use(GameObject player)
    {
        player.GetComponent<ExampleHealth>().Health += healAmount;
    }
}
```

NOTE: Each object has a sphere collider trigger attached to it when it gets created through the wizard. This is needed to add the item when the player walks over / near it. If it doesn't get triggered, then the radius may need to be expanded and then applied on the prefab.

CUSTOMISING THE GRAPHICS

In the ***InventoryBar*** prefab there is an ***InventroyTemplate***. This is what's used as a base for each inventory slot. You'll find an ***InvantorySlot*** script attached which controls how the slot appears in its different states. The ***BaseSlotImage*** is what's show when the object is not highlighted. The ***ActiveSlotImage*** is what's shown when the object is highlighted. The subsequent 2 fields should already be set up but they determine where the preview image is and the item count text are.

For the bar itself the graphic is on the ***InvantoryRoot*** GameObject.

THE DEMO SCENE

When you press play you'll be able to look and walk around the simple scene and object will be added to your inventory as you walk over them. To use the selected object press ***Space / Return***.

LICENCE

CHANGELOG

Version 1.1.1 – 15/05/2017

- Fixed another conflict in the build process
- Added legacy support for Unity 5.0, Unity 5.1, Unity 5.4 & Unity 5.6

Version 1.1.0 – 07/11/2016

- Added mobile (touch and mouse) support
- Added tooltips

Version 1.0.1 – 13/10/2016

- Fixed conflict in build process

Version 1.0.0 – 07/10/2016

- Initial Release

QUESTIONS

If you have any question with how to use the product, then get it touch and I'd be happy to help:

Website: http://freddiebabord.com
Email: freddie.babord@gmail.com