# Advanced Save System

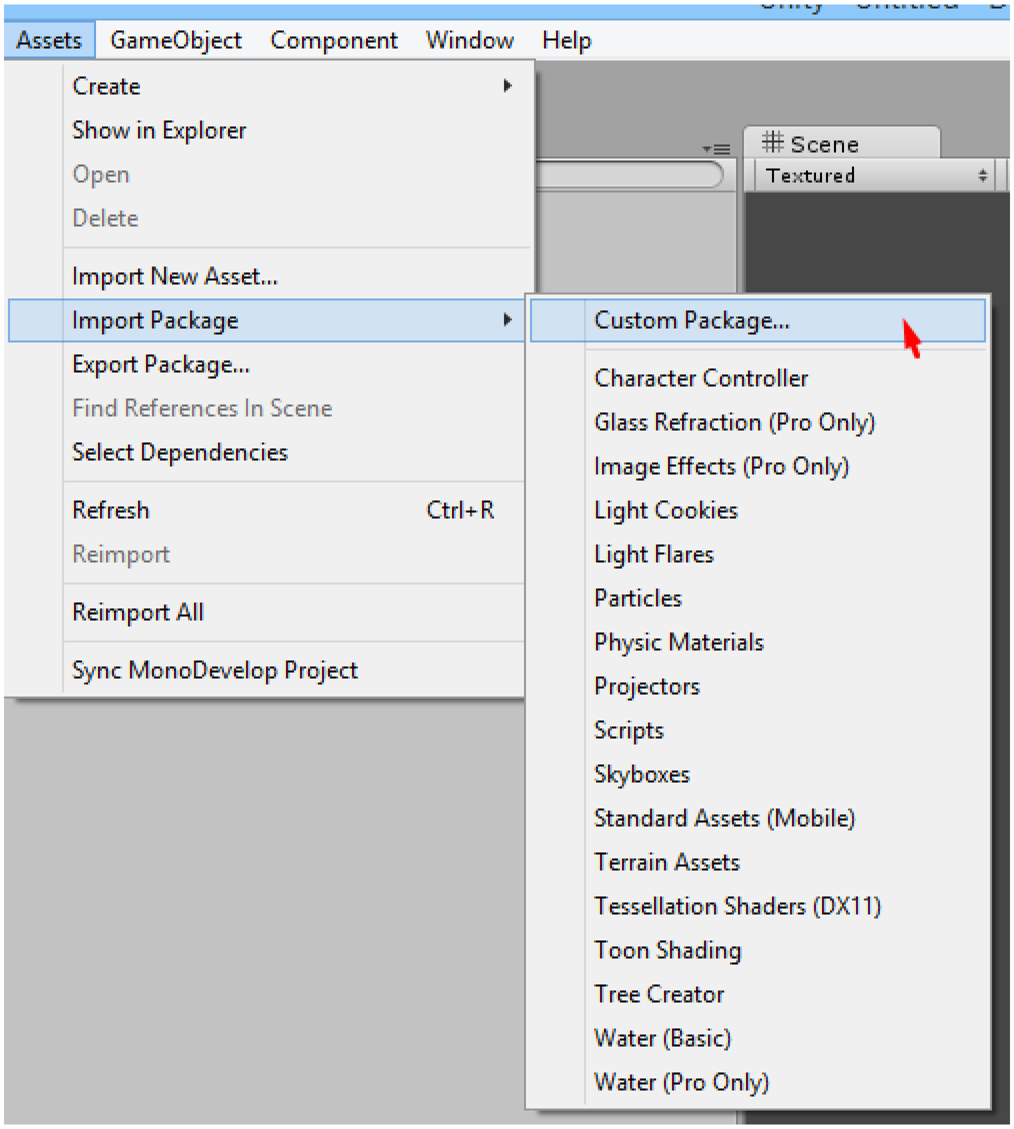
**Documentation Information :**

Version : 1.0 Alpha

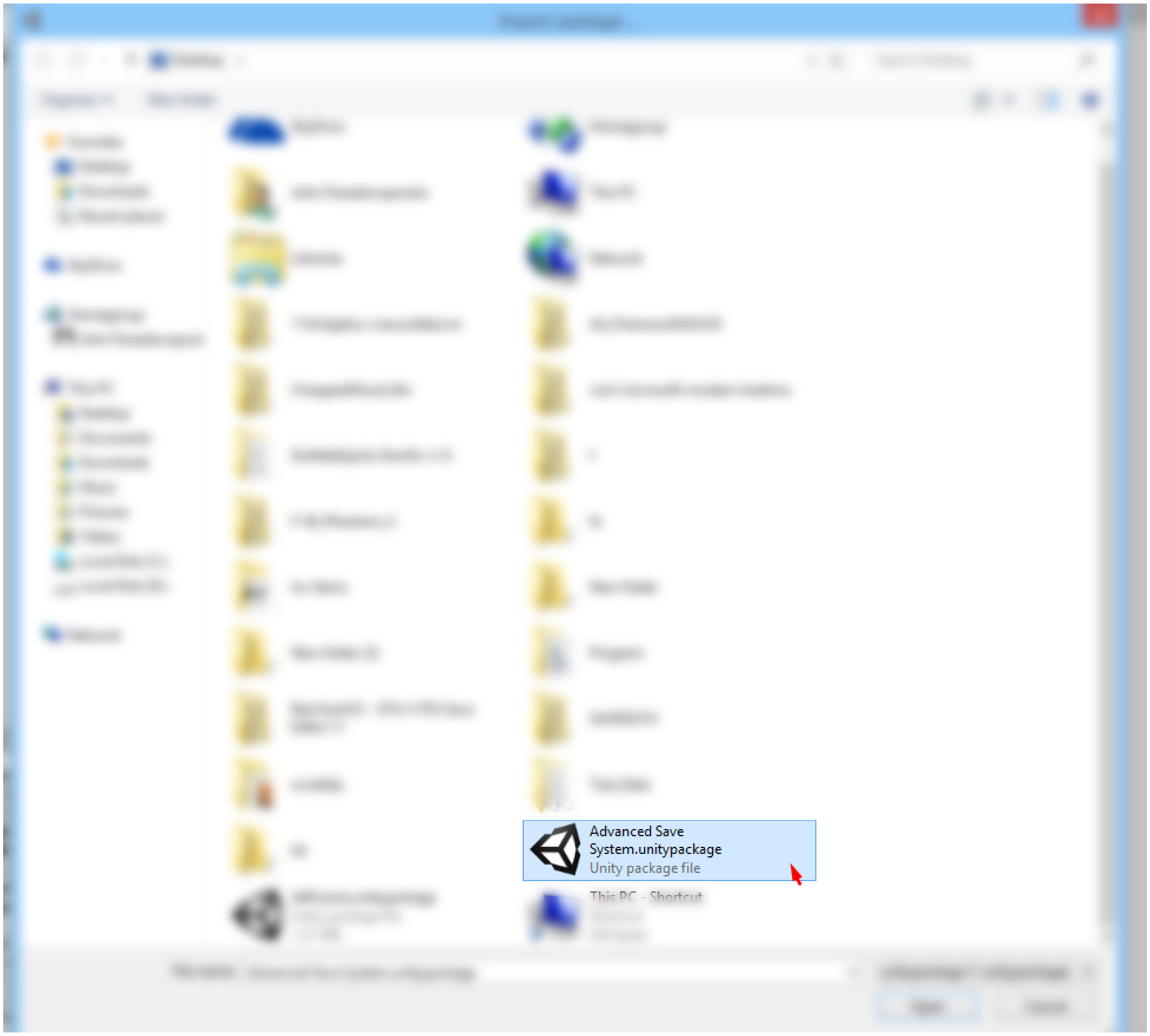
Developer : GreekStudios

Tested on version : 4.3.0f4

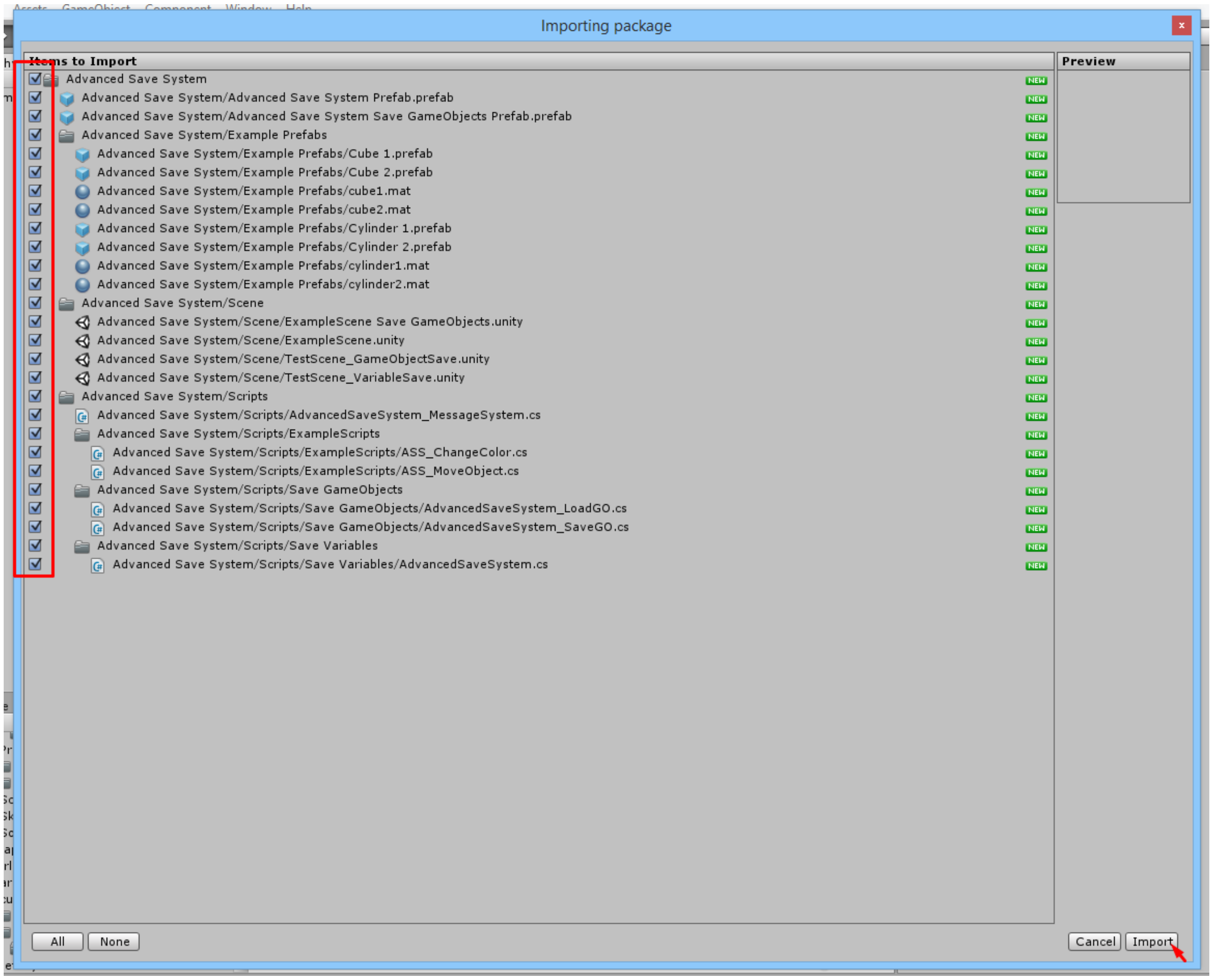
## 1.) Import to Unity



Select the "Advanced Save System.unitypackgage" file.



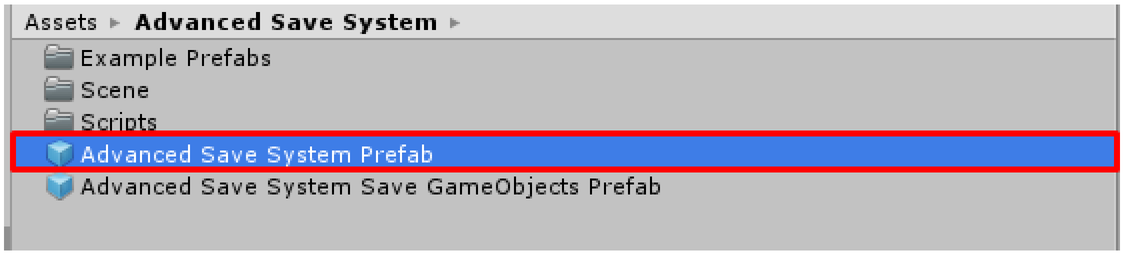
Make sure that all files are selected and hit "Import"



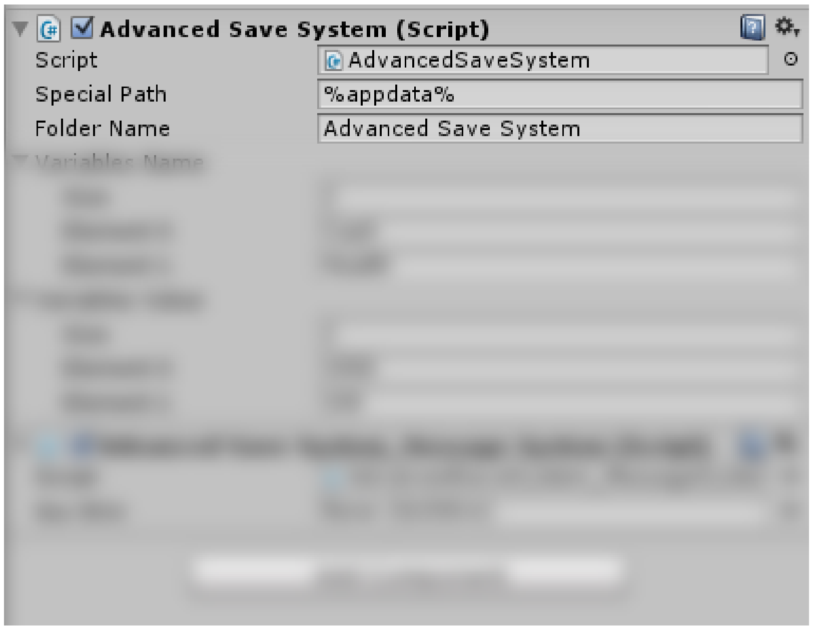
```
Advanced Save System
Advanced Save System/Advanced Save System Prefab.prefab
Advanced Save System/Advanced Save System Save GameObjects Prefab.prefab
Advanced Save System/Example Prefabs
Advanced Save System/Example Prefabs/Cube 1.prefab
Advanced Save System/Example Prefabs/Cube 2.prefab
Advanced Save System/Example Prefabs/cube1.mat
Advanced Save System/Example Prefabs/cube2.mat
Advanced Save System/Example Prefabs/Cylinder 1.prefab
Advanced Save System/Example Prefabs/Cylinder 2.prefab
Advanced Save System/Example Prefabs/cylinder1.mat
Advanced Save System/Example Prefabs/cylinder2.mat
Advanced Save System/Scene
Advanced Save System/Scene/ExampleScene Save GameObjects.unity
Advanced Save System/Scene/ExampleScene.unity
Advanced Save System/Scene/TestScene_GameObjectSave.unity
Advanced Save System/Scene/TestScene_VariableSave.unity
Advanced Save System/Scripts
Advanced Save System/Scripts/AdvancedSaveSystem_MessageSystem.cs
Advanced Save System/Scripts/ExampleScripts
Advanced Save System/Scripts/ExampleScripts/ASS_ChangeColor.cs
Advanced Save System/Scripts/ExampleScripts/ASS_MoveObject.cs
Advanced Save System/Scripts/Save GameObjects
Advanced Save System/Scripts/Save GameObjects/AdvancedSaveSystem_LoadGO.cs
Advanced Save System/Scripts/Save GameObjects/AdvancedSaveSystem_SaveGO.cs
Advanced Save System/Scripts/Save Variables
Advanced Save System/Scripts/Save Variables/AdvancedSaveSystem.cs
```

## 2.)How to set up the script to save your variables

Select the Advanced Save System Prefab and drop it into your scene


Assets ▸ Advanced Save System ▸
Example Prefabs
Scene
Scripts
Advanced Save System Prefab
Advanced Save System Save GameObjects Prefab

You can now see that properties in your inspector.



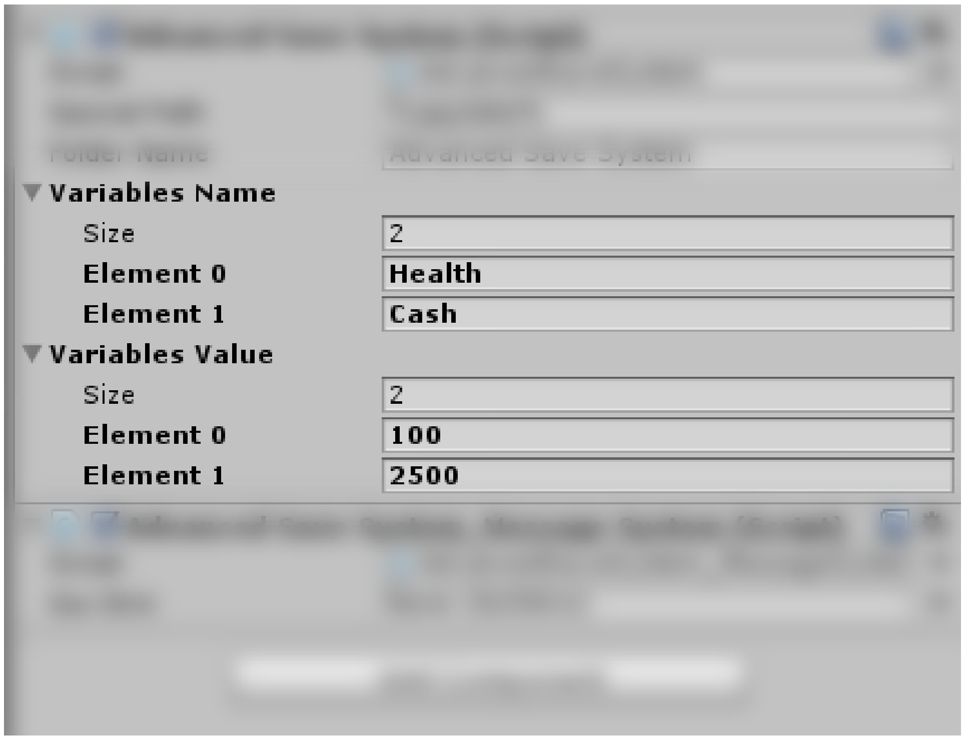
Advanced Save System (Script)

| | |
|---|---|
| Script | AdvancedSaveSystem |
| Special Path | %appdata% |
| Folder Name | Advanced Save System |

At **special path** you can type environment variables between '%'. (eg : %AppData , %LOCALAPPDATA% , %ProgramFiles% , %CommonProgramFiles% , %SystemRoot% , %UserDomain% , %UserProfile% etc..)
If you select a directory that requires administrator privileges then your game must run as administrator in order to work.

At **Folder Name** type a name for the folder you want to be created in the special select.

Now to assign and save the variables you want you have to type the amount of variables you want in Size on both Variables Name and Variables Value arrays.
It's important to have the same size between that two arrays.On Variables Name array type the name of the variable (on our example it would be Health and the second one would be Cash), and on the Variables Value type the values you want to have.



Variables Name

| | |
|---|---|
| Size | 2 |
| Element 0 | Health |
| Element 1 | Cash |

Variables Value

| | |
|---|---|
| Size | 2 |
| Element 0 | 100 |
| Element 1 | 2500 |

The Advanced Save System is now ready to work, but to call it you have to make your own script or use the example. Here is a simple example on how to call it and how to select between the 3 available slots.

First attach your script to the same gameobject (Advanced Save System Prefab).
Now you have to update the values before you save them. To do it just type

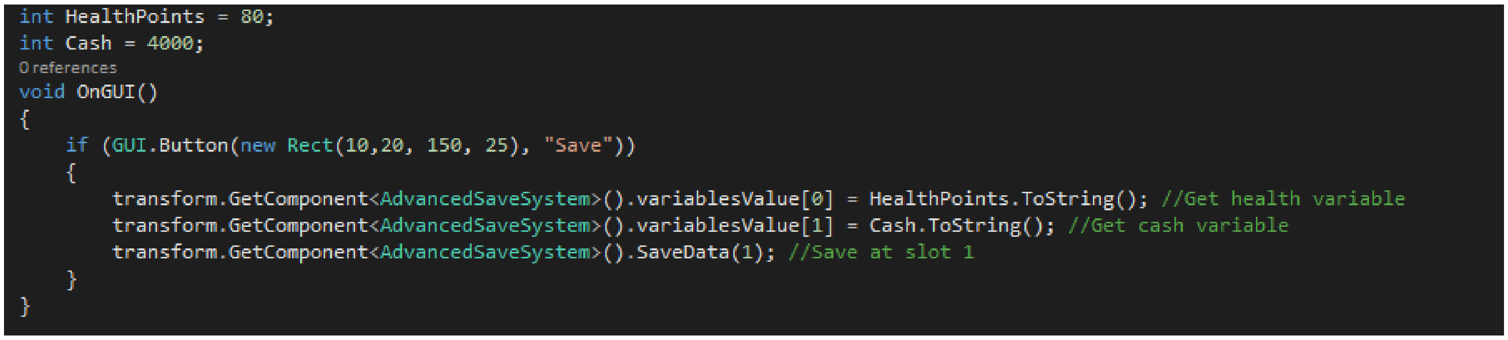
transform.GetComponent**<AdvancedSaveSystem>**().variablesValue[#Element ID#] = #YourVariable#

At #Element ID# type the id (as int) that represent your variable at the arrays.
At #YourVariable# type the variable you like to save.
Then to save type

transform.GetComponent**<AdvancedSaveSystem>**().SaveData(#Slot#);

At #Slot# type the slot you like to save (From 1 to 3) , (as int).

So in our example the script would be :



```
int HealthPoints = 80;
int Cash = 4000;
0 references
void OnGUI()
{
    if (GUI.Button(new Rect(10,20, 150, 25), "Save"))
    {
        transform.GetComponent<AdvancedSaveSystem>().variablesValue[0] = HealthPoints.ToString(); //Get health variable
        transform.GetComponent<AdvancedSaveSystem>().variablesValue[1] = Cash.ToString(); //Get cash variable
        transform.GetComponent<AdvancedSaveSystem>().SaveData(1); //Save at slot 1
    }
}
```

To load you have to type

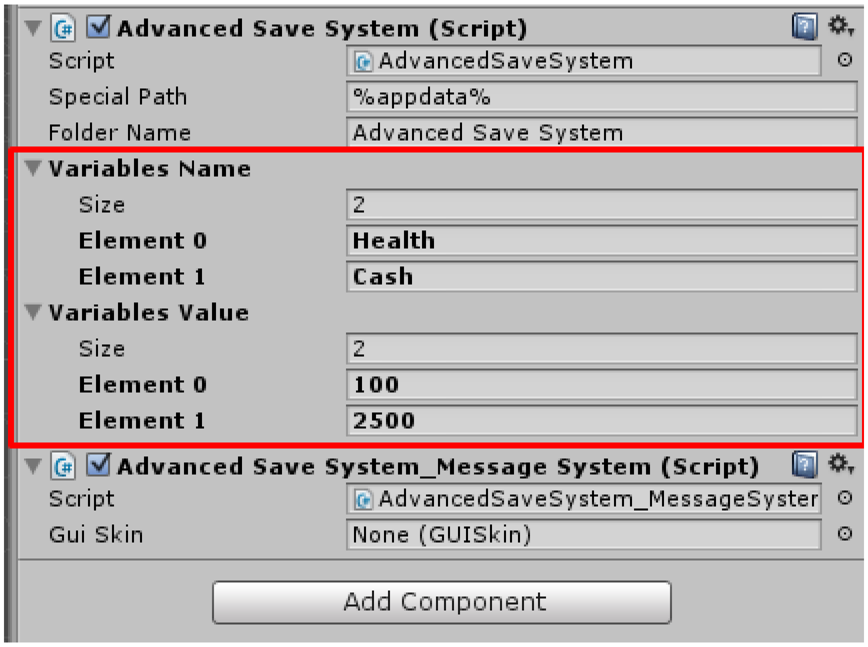
transform.GetComponent**<AdvancedSaveSystem>**().LoadData(#Slot#);

At #Slot# type the slot you want to load (From 1 to 3), (as int).
Then you have to take and set the loaded values to the variables you want.To do that simply type

#YourVariable# = transform.GetComponent**<AdvancedSaveSystem>**().variablesValue[#ElementID#];

At #YourVariable# the variable you want to set the loaded value.
At #ElementID# the int that represent the variable you want from the arrays.

To make sure you save and load the correct variables check your element id from the arrays



Advanced Save System (Script)

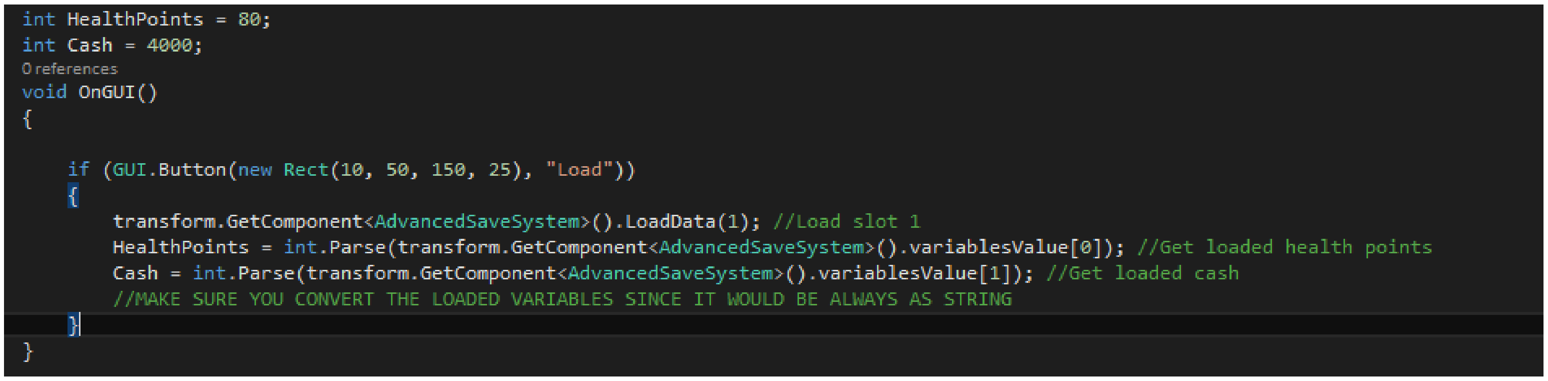
| | |
|---|---|
| Script | AdvancedSaveSystem |
| Special Path | %appdata% |
| Folder Name | Advanced Save System |
| **Variables Name** | |
| Size | 2 |
| **Element 0** | **Health** |
| **Element 1** | **Cash** |
| **Variables Value** | |
| Size | 2 |
| **Element 0** | **100** |
| **Element 1** | **2500** |

Advanced Save System_Message System (Script)

| | |
|---|---|
| Script | AdvancedSaveSystem_MessageSyster |
| Gui Skin | None (GUISkin) |

In our example :
"0" represents Health
"1" represents Cash

So the script would be :

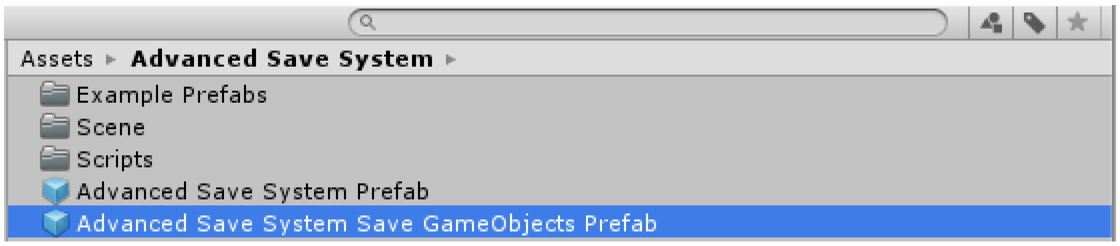


```
int HealthPoints = 80;
int Cash = 4000;
0 references
void OnGUI()
{

    if (GUI.Button(new Rect(10, 50, 150, 25), "Load"))
    {
        transform.GetComponent<AdvancedSaveSystem>().LoadData(1); //Load slot 1
        HealthPoints = int.Parse(transform.GetComponent<AdvancedSaveSystem>().variablesValue[0]); //Get loaded health points
        Cash = int.Parse(transform.GetComponent<AdvancedSaveSystem>().variablesValue[1]); //Get loaded cash
        //MAKE SURE YOU CONVERT THE LOADED VARIABLES SINCE IT WOULD BE ALWAYS AS STRING
    }
}
```

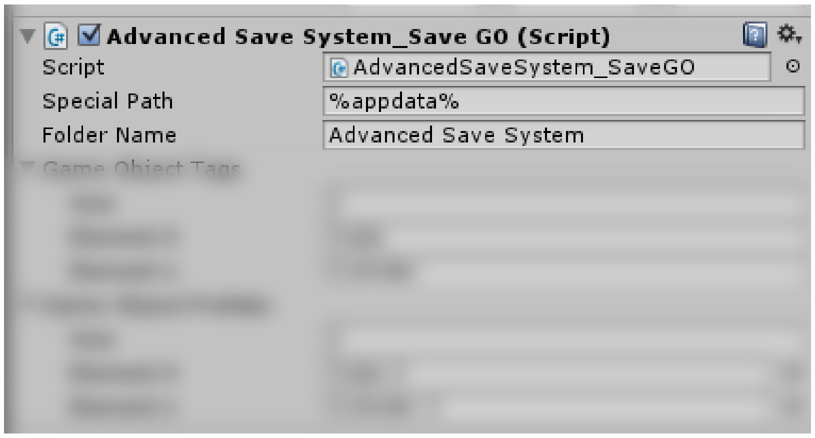
**3.)How to set up the script to save tha gameobjects you want.**

The way this system works is using Tags.So to start on your scene drag and drop the "Advanced Save System Save GameObjects Prefab"



Assets ▸ Advanced Save System ▸
Example Prefabs
Scene
Scripts
Advanced Save System Prefab
Advanced Save System Save GameObjects Prefab

Then select the prefab, now lets take a look at the inspector.

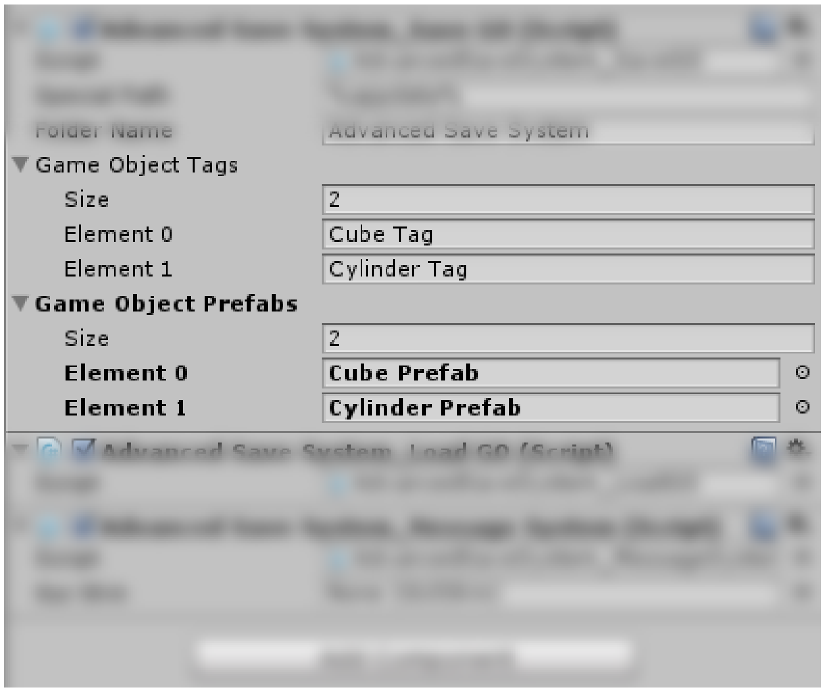


Advanced Save System_Save GO (Script)

| | |
|---|---|
| Script | AdvancedSaveSystem_SaveGO |
| Special Path | %appdata% |
| Folder Name | Advanced Save System |

The two first variables works the same with the other system.Note it's not necessary to use the same directory.

Now, in order to save the gameobjects you like just go to the "Game Object Tags" array and add as many gameobject tags as you like.Then on the second array the "Game Object Prefabs" add the same size with the previous,you have to add a prefab for each tag you add in order to spawn it.



| | |
|---|---|
| Folder Name | Advanced Save System |
| **Game Object Tags** | |
| Size | 2 |
| Element 0 | Cube Tag |
| Element 1 | Cylinder Tag |
| **Game Object Prefabs** | |
| Size | 2 |
| **Element 0** | **Cube Prefab** |
| **Element 1** | **Cylinder Prefab** |

Advanced Save System_Load GO (Script)

Make sure that the prefab that corresponds to the appropriate tag.Like the example the Cube Tag is on '0', and the Cube Prefab is on '0' too.

Now, in order to call the Save and Load functions you have to make a script and attach it into the same gameobject (Advanced Save System Save GameObjects Prefab).Then add the following code in order to save

```
transform.GetComponent<AdvancedSaveSystem_SaveGO>().SaveGameObjects(#Slot#);
```

At #Slot# type the slot you want to load (From 1 to 3) , (as int).

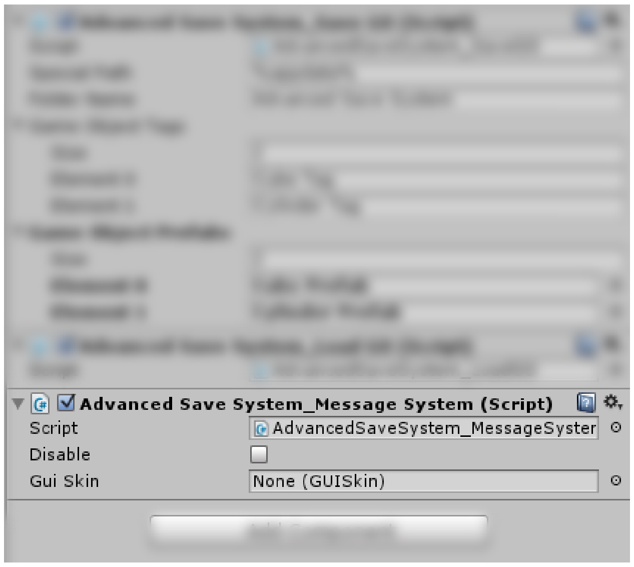
In order to load type the following code

```
transform.GetComponent<AdvancedSaveSystem_LoadGO>().LoadGameObjects(#Slot#);
```

At #Slot# type the slot you want to load (From 1 to 3) , (as int).

## 3.) Messaging function

On both prefabs for saving variables and game objects there is a script attached that would give a message to the top left corner for a short period of time and it would say "Game Saved" or "Game Loaded".To disable it just check the Disable box.You can add your own GUI Skin and if you want you can modify the script to fill your needs.

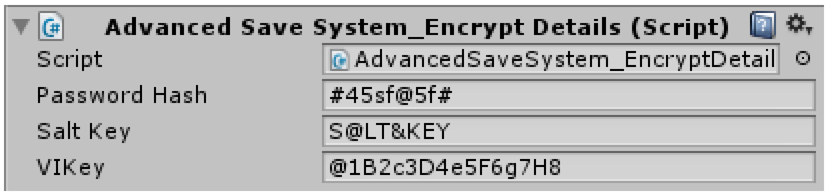


Advanced Save System_Message System (Script)

| | |
|---|---|
| Script | AdvancedSaveSystem_MessageSyster |
| Disable | ☐ |
| Gui Skin | None (GUISkin) |

## 4.)Change  encryption details

If you are not advanced on that things change only the Password Hash and the Salt Key



Advanced Save System_Encrypt Details (Script)

| | |
|---|---|
| Script | AdvancedSaveSystem_EncryptDetail |
| Password Hash | #45sf@5f# |
| Salt Key | S@LT&KEY |
| VIKey | @1B2c3D4e5F6g7H8 |

**That was the Advanced Save System made by GreekStudios.Feel free to ask any question at paraskevlos@yahoo.gr. Answers within 7 days.**