Soap: 5_Save_Example_Scene

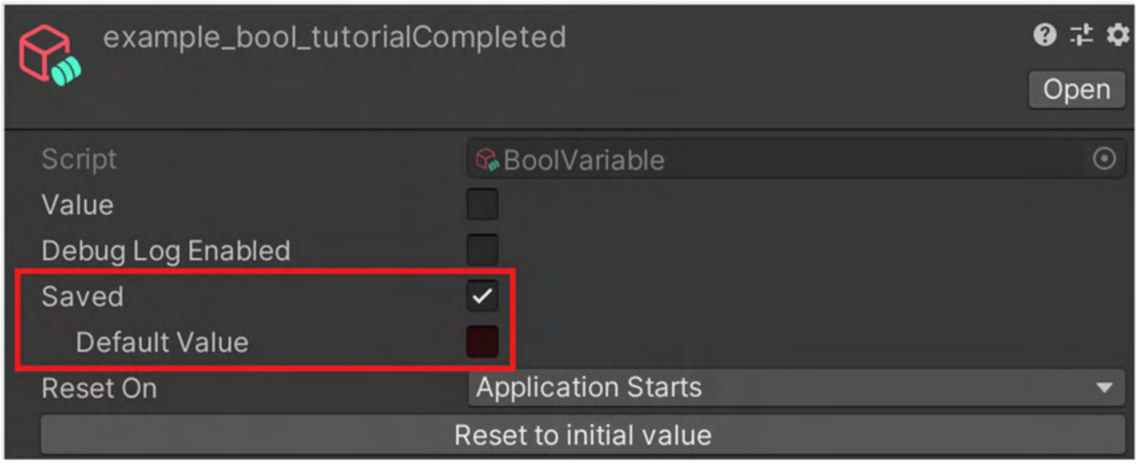
Table of Contents

Single click save

Let's start by testing the scene. If you enter play mode, and press the button "**End Tutorial**", you will see that the **Tutorial** object will disable itself.

Now exit play mode and play again. You can see that the changes have persisted. It's because the *example_bool_tutorialCompleted* variable state has been saved to **Player Prefs**. To enable saving on your variable, select the variable and enable the property "**Saved**". All included scriptable variables can be saved to Player Prefs **with a single click.** (Except the GameObjectVariable).

When loaded for the first time, the variable will be set to the **Default Value**.



example_bool_tutorialCompleted

Script: BoolVariable

Value: ☐

Debug Log Enabled: ☐

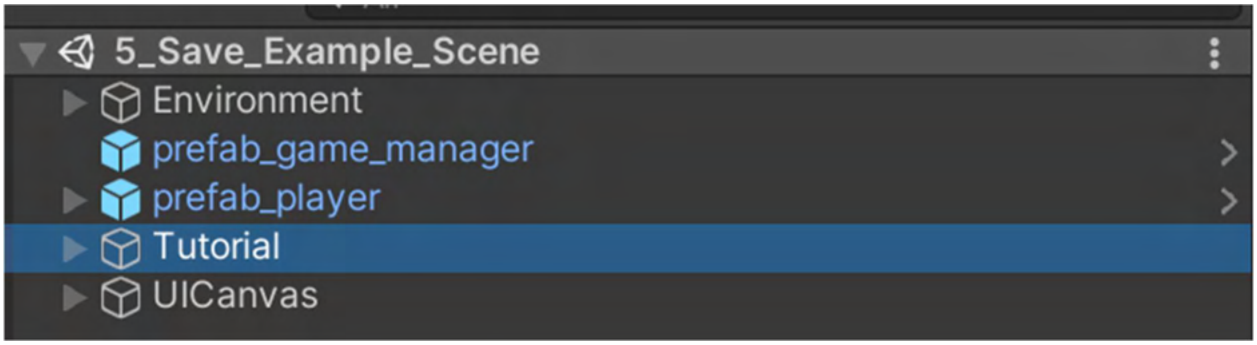
Saved: ☑

Default Value: ☐

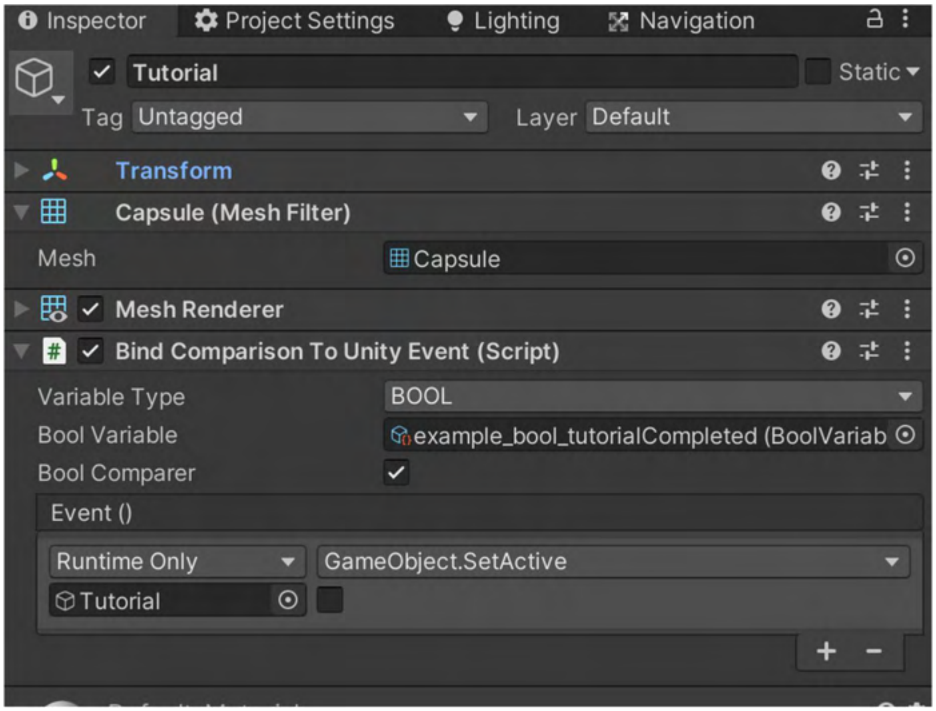
Reset On: Application Starts

Note: The key used to save to Player Prefs is the Guid of the ScriptableVariable. This Guid is generated using the path of the asset, but you can simply forget about it 😊 .

You can use the fact that a variable has been saved in combination with a Binding scripts. If you select the **Tutorial** in the scene, you can see that its component BindComparisonToUnityEvent will disable the object if the condition is true.



5_Save_Example_Scene
Environment
prefab_game_manager
prefab_player
Tutorial
UICanvas



☑ Tutorial □ Static
Tag Untagged Layer Default

Transform

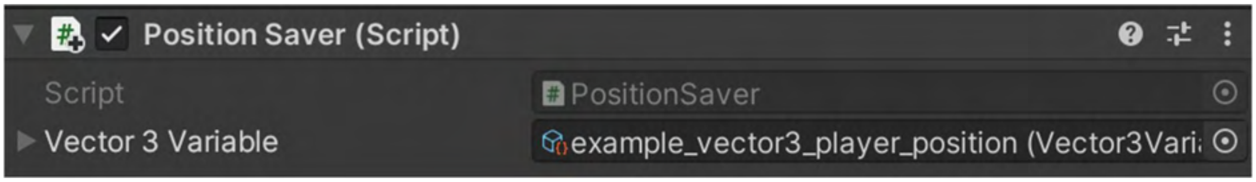
Capsule (Mesh Filter)
Mesh: Capsule

☑ Mesh Renderer

☑ Bind Comparison To Unity Event (Script)
Variable Type: BOOL
Bool Variable: example_bool_tutorialCompleted (BoolVariab
Bool Comparer: ☑
Event ()
Runtime Only | GameObject.SetActive
Tutorial | □

Saving is useful for things like level index, player score, coins, tutorial completed, etc.

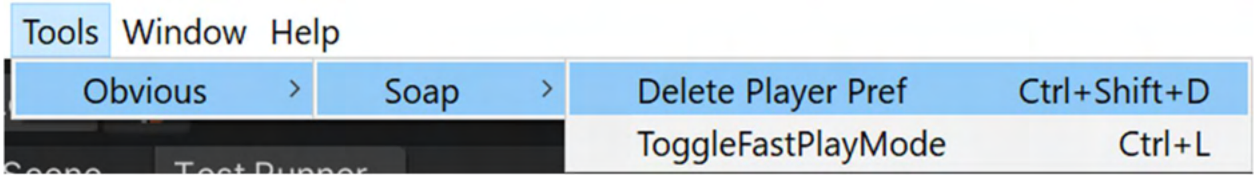
The player position is also saved, check the **PositionSaver.cs** on the **prefab_player** in the scene.



☑ Position Saver (Script)
Script: PositionSaver
Vector 3 Variable: example_vector3_player_position (Vector3Vari

Clearing Save

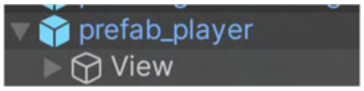
If you want to **clear the save** (deleting all Player Prefs), use the shortcut (CTRL+Shift+D) or the menu item:



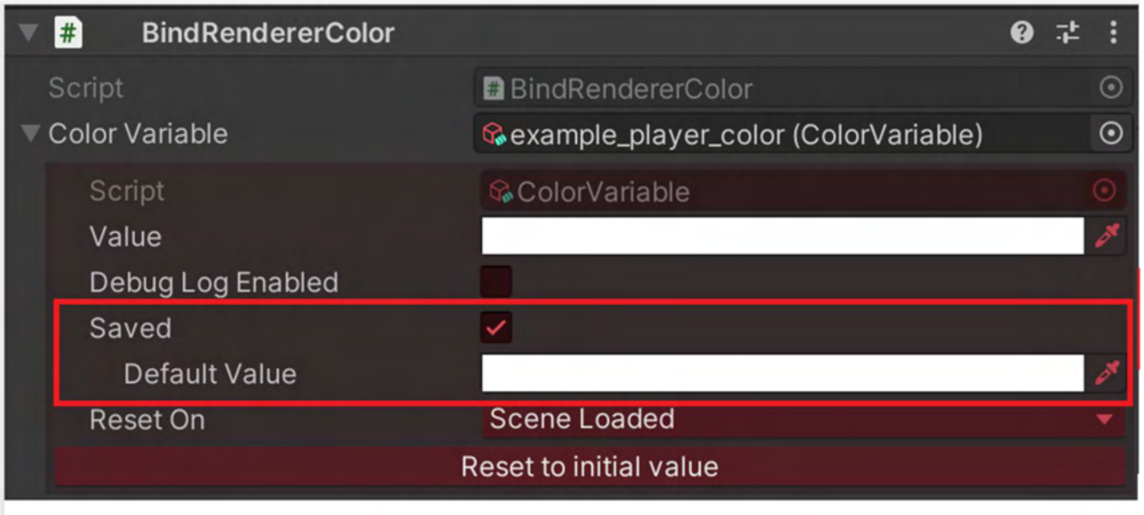
Exercise

You can set a random color to the player by clicking on the "Random Color" button on the bottom of the screen at runtime. Currently, the color is not saved. **Can you find how to save the player color?**

<u>Solution 1</u>: find the **BindRendererColor.cs** component on the prefab_player ->View.



Then, expand the Scriptable Variable, enable the "Saved" property and set the default color to white.



BindRendererColor

Script: BindRendererColor

Color Variable: example_player_color (ColorVariable)

Script: ColorVariable

Value: ⬜

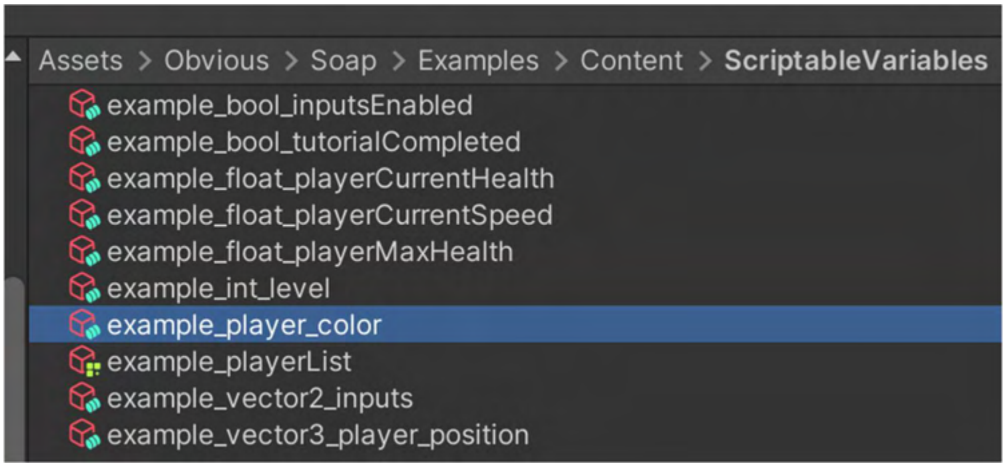
Debug Log Enabled: □

Saved: ☑

Default Value: ⬜

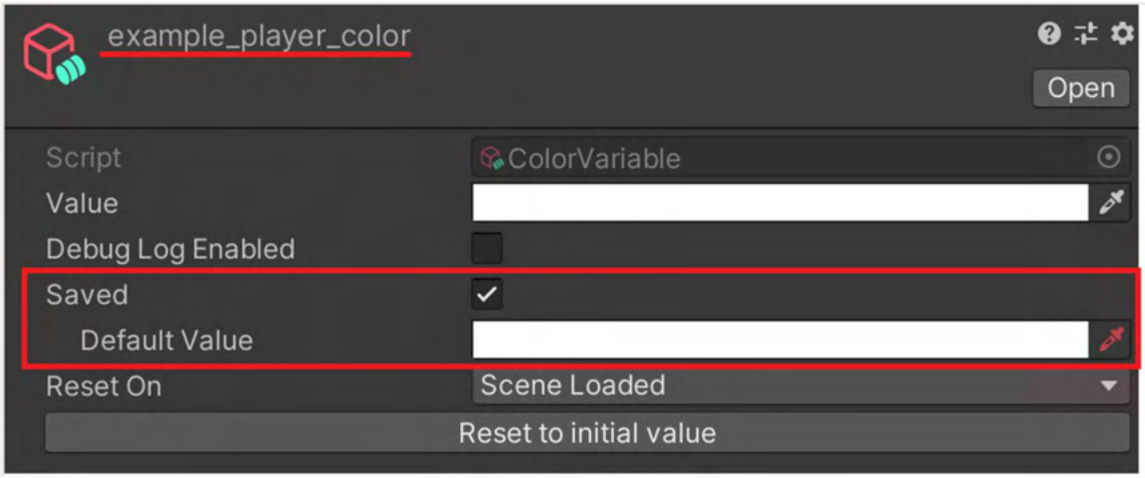
Reset On: Scene Loaded

Solution 2: find the scriptable variable **example_player_color** and enable the "Saved" property.



Assets > Obvious > Soap > Examples > Content > **ScriptableVariables**

- example_bool_inputsEnabled
- example_bool_tutorialCompleted
- example_float_playerCurrentHealth
- example_float_playerCurrentSpeed
- example_float_playerMaxHealth
- example_int_level
- example_player_color
- example_playerList
- example_vector2_inputs
- example_vector3_player_position



example_player_color

Script: ColorVariable

Value

Debug Log Enabled ☐

Saved ☑

Default Value

Reset On: Scene Loaded

Note: when you create a new scriptable variables type, you can override the Save and Load method, and have custom save/load for them using Player Prefs 😊. Check the ColorVariable or Vector2Variable to have a concrete example.