

Soap:

5_Save_Example_Scene

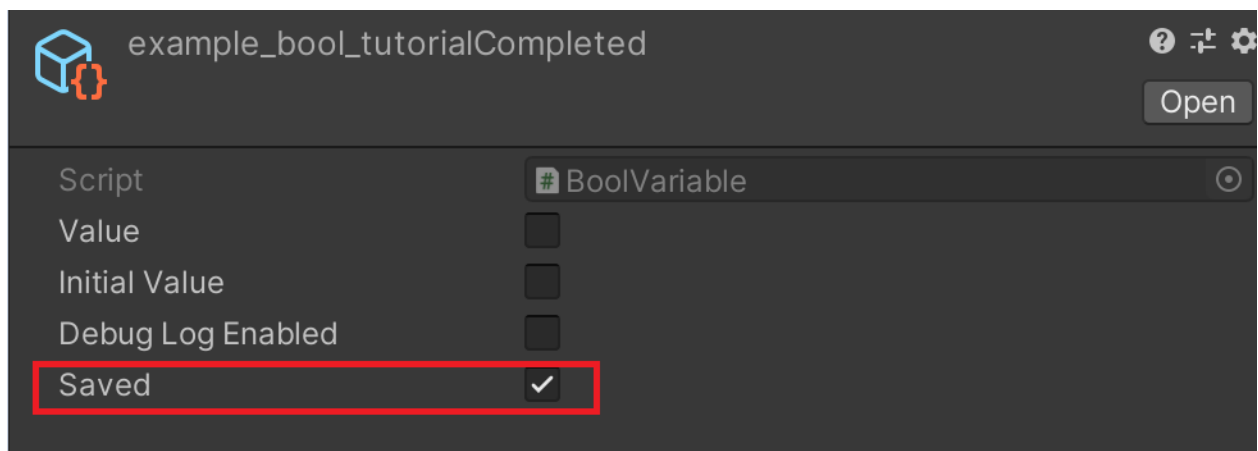
Table of Contents

Single click save.....	2
Clearing Save.....	3
Exercise	4

Single click save

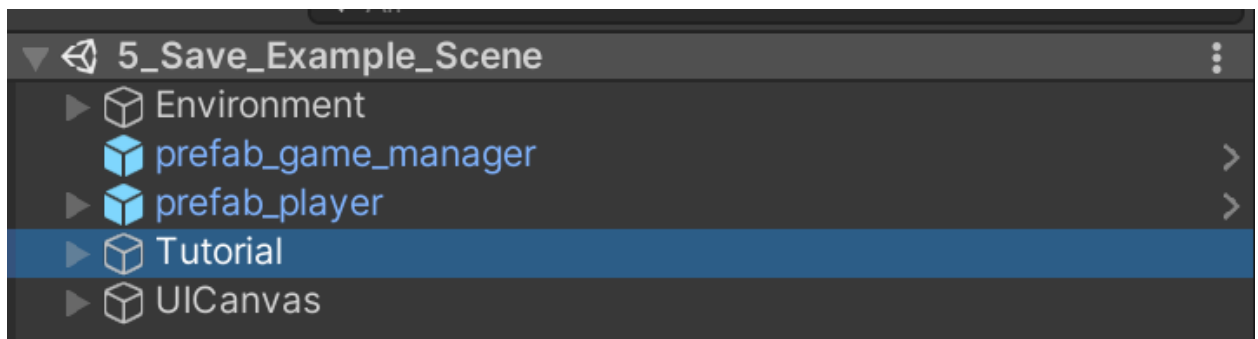
Let's start by testing the scene. If you enter play mode, and press the button “**End Tutorial**”, you will see that the **Tutorial** object will disable itself.

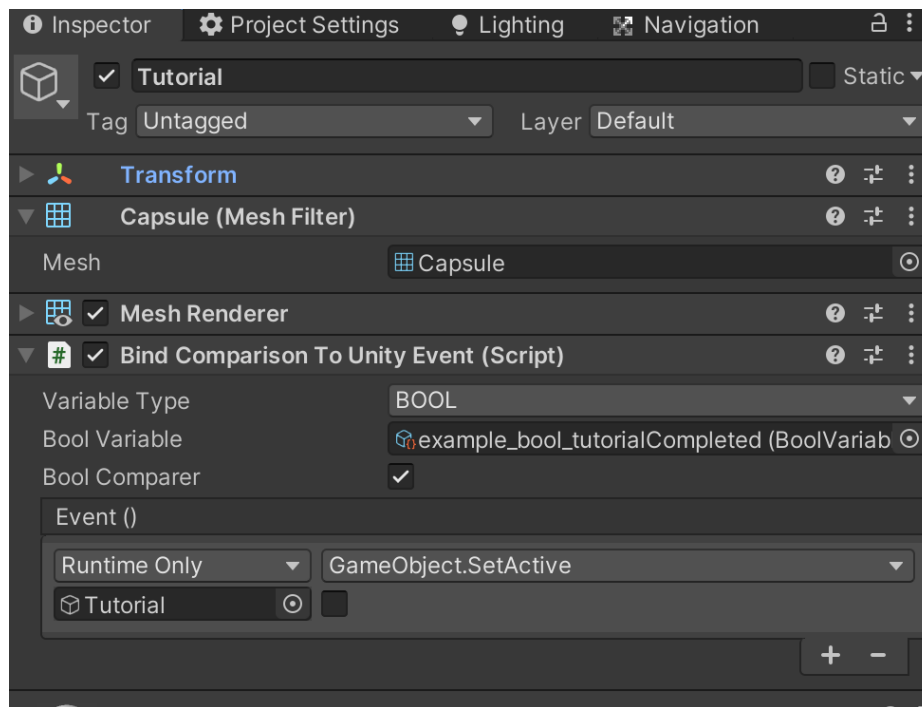
Now exit play mode and play again. You can see that the changes have persisted. It's because the *example_bool_tutorialCompleted* variable state has been saved to **Player Prefs**. To enable saving on your variable, select the variable and enable the property “**Saved**”. All scriptable variables can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



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 😊.

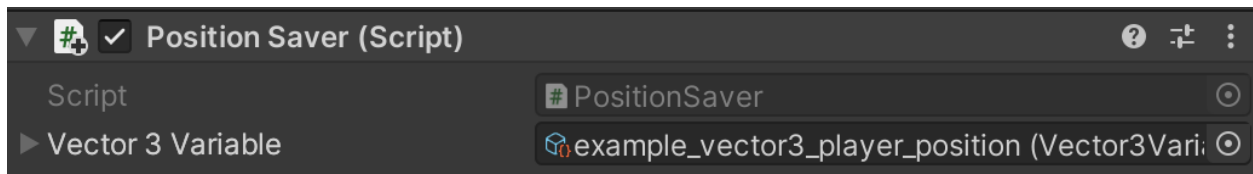
You can use the fact that a variable has been saved in combination with a Binding scripts. If you select the **Tutorial** in the scene, you can see that its component `BindComparisonToUnityEvent` will disable the object if the condition is true.





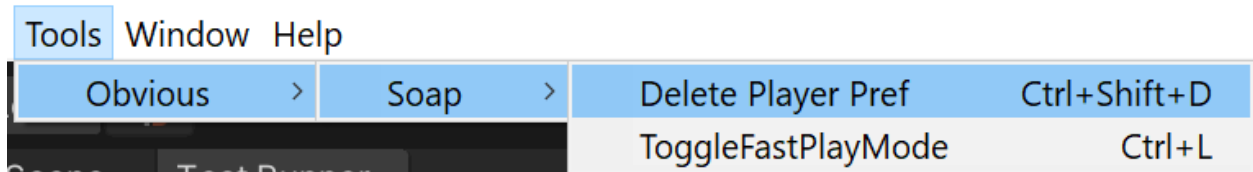
Saving is useful for things like level index, player score, coins, tutorial completed, etc.

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



Clearing Save

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

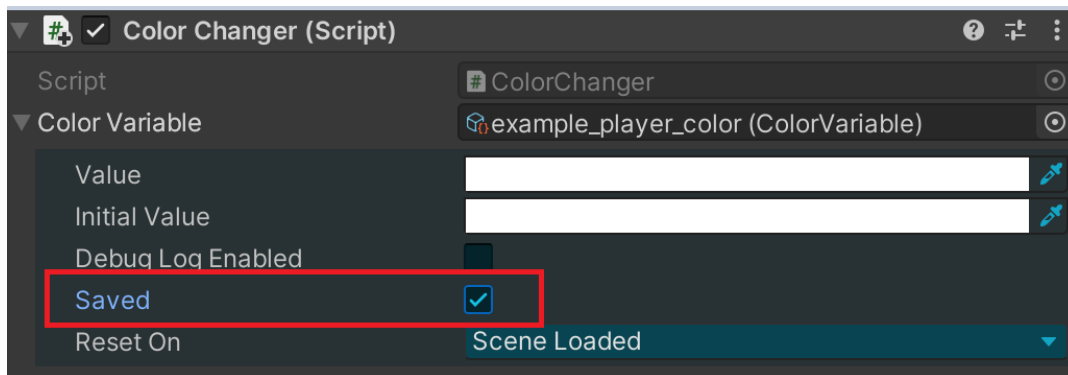


Exercise

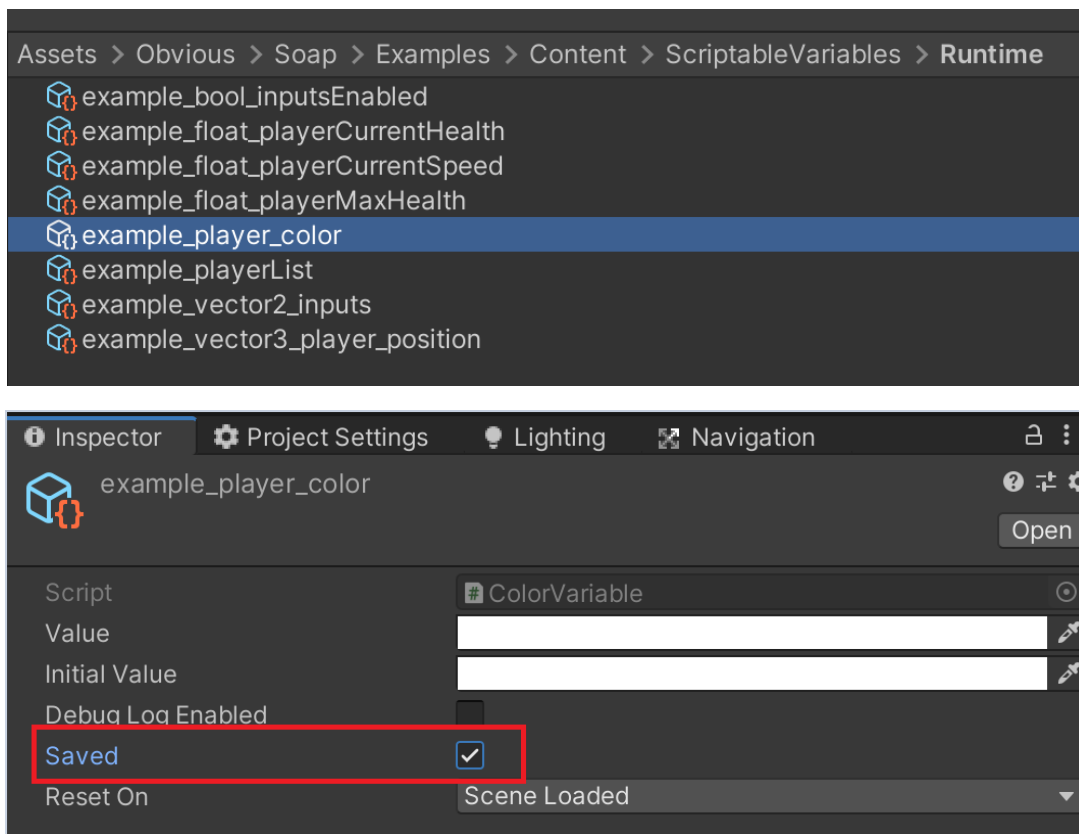
You can set a random color to the player by clicking on the “Random Color” button on the bottom of the screen at runtime. Currently, the color is not saved. **Can you find how to save the player color?**

See the solutions on the next page ->

Solution 1: find the **ColorChanger.cs** component on the prefab_player, expand the Scriptable Variable and enable the “Saved” property.



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



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.