Tutorial Master v1.0 - API Documentation

While Tutorial Master requires almost no code to get it up and running, there are some hidden gem functions that may help you to customize your tutorials even deeper.

In order to access such functions, you must put `using HardCodeLab.TutorialMaster;` at the top of the script

Function	Description	Example
`tutorial.Start((int)tutorial_id, (bool)continue where left?)`	Starts the tutorial of specific ID. Optionally, you can tell whether you want to continue from the last played frame or not	`tutorial.Start(0, true)`
`tutorial.Stop()`	Stops the current tutorial. Command will be ignored if no active tutorial found	`tutorial.Stop()`
`frame.Next()`	Goes to the next frame of the tutorial. Command will be ignored if no active tutorial found	`tutorial.Next()`
`frame.Prev()`	Goes to the previous frame of the tutorial. Command will be ignored if no active tutorial found	`tutorial.Prev()`
`tutorial.Goto((int)frame_id)`	Goes to the frame of specified ID of the tutorial. Command will be ignored if no active tutorial found	`tutorial.Goto(2)`
`data.Reset()`	Deletes PlayerPrefs related to the Tutorial Master	`data.Reset()`

Variables	Description
`data.tutorialsCompleted`	Returns an integer of number of tutorials that has been completed
`data.tutorialsInProgress`	Returns an integer of number of tutorials that has been completed
`frame.id`	Returns the ID of currently played frame. Returns -1 if no active tutorial found
`tutorial.id`	Returns the ID of the currently active tutorial. Returns -1 if no active tutorial found
`tutorial.progress`	Returns a float, representing the progress of the current tutorial between 0 and 1. (1 means completed)