

# Mini-Game Pack - Sliding Puzzle

## Documentation

Hydrolysis Studio

2022/02/06

Contact

[HydrolysisStudio@Outlook.com](mailto:HydrolysisStudio@Outlook.com)

## Contents

Introduction.....	3
What is this pack and what can I do with it?.....	3
What are the features of this system?.....	3
What assets are included in this pack?.....	3
Scripts.....	4
PuzzleBox.cs.....	4
Puzzle.cs.....	4
GameState.cs.....	5
Q&A.....	7
Where to find video tutorials for this pack?.....	7
How to quickly setup a custom sliding puzzle and test it?.....	7
Releases.....	8
v1.0 build 220206.....	8

# Introduction

## What is this pack and what can I do with it?

This mini-game pack contains an intellectual puzzle game named sliding puzzle, it is really simple and funny to play. Player will start with a messy puzzle, the last piece of the puzzle will be removed, then player will move only one piece near the empty place, the final goal of this game is to restore the pattern. With this pack, you can quickly build a sliding puzzle game, almost all fundamental core scripts are in place, it will save your valuable time to make better games.

## What are the features of this system?

- Starter **friendly**, easy to setup and use, you can setup your first sliding puzzle within a few minutes by following the instruction video.
- **Customizable** with several parameters, like puzzle size, image, sound, speed, etc.
- Contains **shuffle algorithm** for puzzle boxes, you don't need to implement the shuffle logic by yourself.
- Use mouse click to move puzzle boxes with **smooth movement**.
- **Simple UI** layout, with texts to show game time and moves, and a button to restart the game.
- Provide **documentation** to explain all scripts, including all variables and functions.
- Provide **tutorial videos**, like how to use and quick setup.
- Scripts have **detailed comments**, almost each line of the code is annotated.

## What assets are included in this pack?

- AudioClip x1
- Script x3
- Prefab x2
- Texture x5
- Material x5
- Scene x1

# Scripts

## [PuzzleBox.cs](#)

This script is created for puzzle box prefab. It contains several basic parameters and four functions.

### Variables

<b>index</b>	int	Index of the puzzle box.
<b>row</b>	int	Row in coordinate for the puzzle box. [0, size-1]
<b>column</b>	int	Column in coordinate for the puzzle box. [0, size-1]
<b>smooth</b>	bool	Indicates whether to use smooth movement.
<b>duration</b>	float	Time used during smooth movement.
<b>velocity</b>	Vector3	Velocity during smooth movement.
<b>click</b>	Action<int, int>	A delegate reference to click event.

### Functions

<b>Init</b>	This function is used to initialize the puzzle box by setting all variables and move it to the corresponding local position. It requires several parameters as input, an index, a coordinate (row and column), a sprite reference, and a delegate reference for click event. It won't return anything. For more details, check the PuzzleBox.cs file and you will see more detailed annotation for each line of the scripts there.
<b>Move</b>	This function is used to move the puzzle box to a specific coordinate with or without smooth movement. It requires two parameters, row and column. It won't return anything. Check the scripts for more details.
<b>OnMouseDown</b>	This is a built-in function and is used to handle mouse click. It will call the delegate function named HandleClick in Puzzle.cs via the reference variable named click.
<b>Update</b>	This is a built-in function and is used to update the local position of the puzzle box smoothly when it is clicked and is moving.

## [Puzzle.cs](#)

This script is created for puzzle prefab. It contains all fundamental core variables and functions for the sliding puzzle game. I made detailed annotation in the scripts, so make sure to use this document with the code.

### Variables

<b>boxes</b>	PuzzleBox[.]	Array for all puzzle boxes.
<b>empty</b>	PuzzleBox	Reference to the empty puzzle box.
<b>container</b>	Transform	Reference to the child GameObject named container to hold all puzzle boxes.
<b>scriptGameState</b>	GameState	Reference to the GameState.cs script.
<b>audioSource</b>	AudioSource	Reference to the AudioSource component.

<b>prefab</b>	PuzzleBox	Prefab for puzzle boxes.
<b>size</b>	int	The size of the puzzle.
<b>material</b>	Material	Material for reference image plane.
<b>sprites</b>	Sprite[]	Array of sprites for puzzle boxes.
<b>delay</b>	float	Delay time after puzzle finished to show the last one.
<b>audioClip</b>	AudioClip	Sound that will be played when any puzzle box is clicked.

### Functions

<b>Start</b>	This is a built-in function and is used to call Init() once game starts.
<b>Init</b>	This functions is used to setup puzzle prefab and game. It will setup all variables, call Generate() to spawn all puzzle boxes, call Shuffle() to randomly arrange the puzzle boxes, etc. Check the scripts for more details, annotation is written for each line of the code.
<b>Generate</b>	This function is used to spawn all puzzle boxes and initialize all of them.
<b>Shuffle</b>	This function contains several algorithms to shuffle puzzle boxes.
<b>SetSmoothMovement</b>	This function is used to enable or disable smooth movement for all puzzle boxes. It requires a bool as input.
<b>SetClickDelegate</b>	This function is used to set click delegate for all puzzle boxes. It requires a Actions<int, int> as input.
<b>InBound</b>	This function is used to check whether a specific coordinate is in bound. If the row or column is out of the range, it will return false.
<b>IsEmpty</b>	This function is used to check whether a puzzle box with specific coordinate is the empty one.
<b>Swap</b>	This function is used to swap two puzzle boxes by applying coordinates and smooth movement. It requires both row and column as input.
<b>HandleClick</b>	This function is used to handle the click event when players use their mouse to click the puzzle box. Check the scripts for more details.
<b>CheckEnd</b>	This function will be called every time when HandleClick() is called. It will check whether the puzzle is finished and return the result as a bool parameter.
<b>GameOver</b>	This function is called when the puzzle is finished. It will stop the timer and remove all delegates, then show the sprite of the last puzzle box after several seconds.
<b>Restart</b>	This function is used to restart the puzzle game. You can click the button on the screen to call this function during your game time.

### GameState.cs

This script is used to handle game information like timer and moves, and also includes UIs texts.

### Variables

<b>txtTime</b>	Text	Reference to the UI text for time.
<b>txtMoves</b>	Text	Reference to the UI text for moves.
<b>btnRestart</b>	Button	Reference to the UI button for restart.

<b>timer</b>	Coroutine	Reference to the timer coroutine.
<b>time</b>	int	Used to record game time, the unit for time variable is seconds.
<b>moves</b>	int	Used to record clicks and moves for puzzle boxes.

### Functions

<b>Start</b>	This is a built-in function and is used to call Init() once game starts.
<b>Init</b>	This function is used to initialize UI texts for time and moves, button for restart game.
<b>Countdown</b>	This function will be called when the timer is activated.
<b>StartCountdown</b>	This function is used to start the countdown timer.
<b>StopCountdown</b>	This function is used to stop the countdown timer.
<b>IncreaseTime</b>	This function will add one second to the time variable.
<b>ClearTime</b>	This function will set the time variable back to 0.
<b>GetTime</b>	This function will return the current game time.
<b>IncreaseMoves</b>	This function will add one step to the moves variable.
<b>ClearMoves</b>	This function will set the moves variable back to 0.
<b>GetMoves</b>	This function will return the current game moves.

# Q&A

Where to find video tutorials for this pack?

I would like to provide all instructions in a shared folder in Google Drive, you can find the link [here](#).

How to quickly setup a custom sliding puzzle and test it?

I uploaded a instruction video to the shared folder, it would be more clear by following the steps with a tutorial video than text instructions.

# Releases

[v1.0 build 220206](#)

- The first released version.