Procedural Book

version 1.8.0

Scripting API

Introduction

This document provides information about the main classes and functions. If you haven't read the **Manual** document yet, please do so first.

Topics

Getting Started

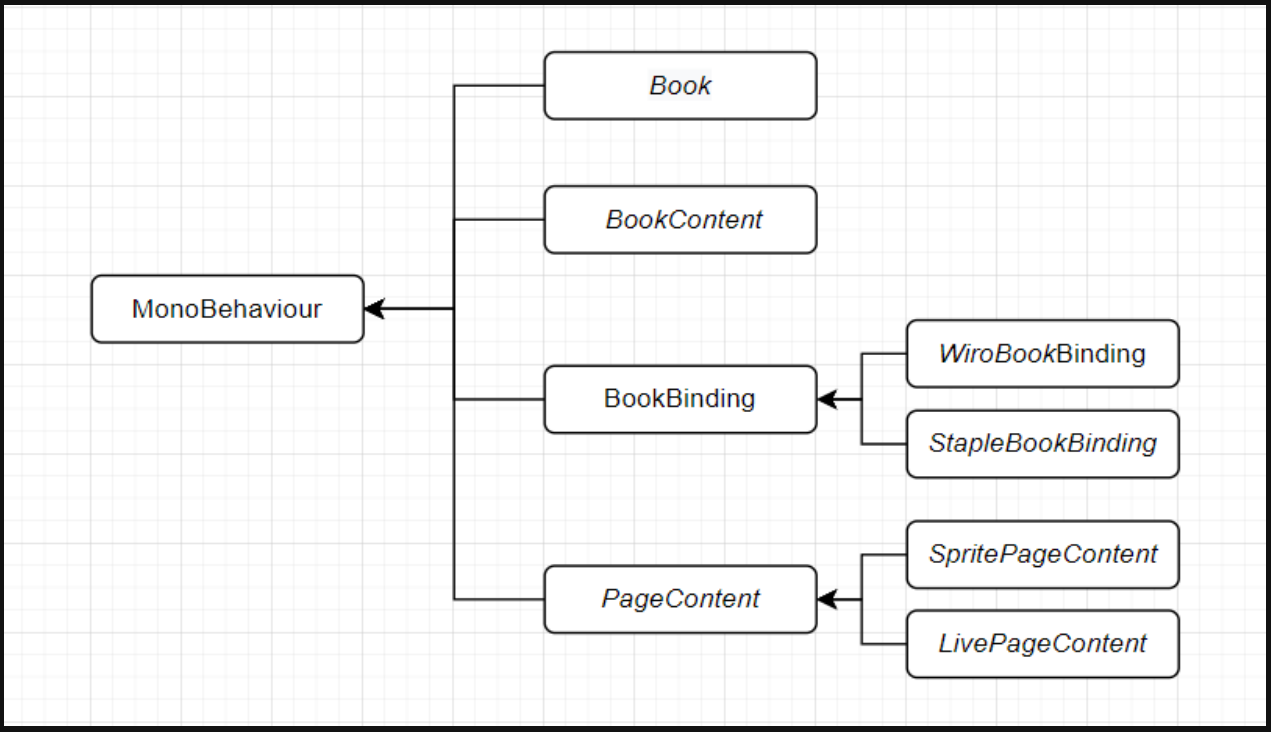
Namespace

ScriptBoy.ProceduralBook

Classes

Book
BookContent
BookBinding
WiroBookBinding
StapleBookBinding
PageContent
SpritePageContent
LivePageContent
LivePageCanvasRaycaster

Hierarchy



How to Turn a Page?

You can control a page turning animation using the following methods and properties of the Book class:

bool StartTurning(Ray ray)
void UpdateTurning(Ray ray)
void StopTurning()
bool isTurning {get;}

Here's a simple example:

```
using UnityEngine;
using ScriptBoy.ProceduralBook;

public class Player : MonoBehaviour
{
    [SerializeField] Book m_Book;

    void Update()
    {
        // 1. Get the current mouse position on the screen
        Vector2 m = Input.mousePosition;
        // 2. Create a ray from the camera through the mouse position
        Ray ray = Camera.main.ScreenPointToRay(m);

        // 3. Check if the book is currently in the process of turning a page
        if (m_Book.isTurning)
        {
            // 5. While the left mouse button is held down, continue updating the turning
            if (Input.GetMouseButton(0))
            {
                m_Book.UpdateTurning(ray);
            }
            // 6. When the left mouse button is released, stop the turning
            else
            {
                m_Book.StopTurning();
            }
        }
        // 4. If the book is not currently turning,
        // start the turning process when the left mouse button is pressed
        else if (Input.GetMouseButtonDown(0))
        {
            m_Book.StartTurning(ray)
        }
    }
}
```

What Is Auto Page Turning?

There is a second solution for turning pages that does not require ray input.

You can control an auto page turning animation using the following methods and properties of the Book class:

bool StartAutoTurning (AutoTurnDirection direction, AutoTurnSettings settings)
or (AutoTurnDirection direction, AutoTurnSettings settings, int turnCount, float delyPerTurn)
or (AutoTurnDirection direction, AutoTurnSettings settings, int turnCount, AutoTurnSetting delyPerTurn)
void CancelPendingAutoTurns()
bool isAutoTurning { get; }
bool hasPendingAutoTurns { get; }
float autoTurningEndTime { get; }

Here's a simple example:

```
using UnityEngine;
using ScriptBoy.ProceduralBook;

public sealed class Player : MonoBehaviour
{
    [SerializeField] Book m_Book;
    [SerializeField] AutoTurnSettings m_AutoTurnSettings;

    public void Update()
    {
        if (Input.GetKeyDown(KeyCode.D))
        {
           m_Book.StartAutoTurning(AutoTurnDirection.Next, m_AutoTurnSettings);
        }

        if (Input.GetKeyDown(KeyCode.A))
        {
           m_Book.StartAutoTurning(AutoTurnDirection.Back, m_AutoTurnSettings);
        }
    }
}
```

Book

Methods

bool StartTurning(Ray ray)
Starts the page turning animation if the specified ray hits the page.
(Returns true if it does not fail.)

void UpdateTurning(Ray ray)
Updates the page turning animation based on the specified ray.

void StopTurning()
Stops the page turning animation.

void GetActivePageIndices(List<int> indices)
Retrieves a list of indices of active pages.

bool StartAutoTurning (AutoTurnDirection direction, AutoTurnSettings settings)
or (AutoTurnDirection direction, AutoTurnSettings settings, int turnCount, float delyPerTurn)
or (AutoTurnDirection direction, AutoTurnSettings settings, int turnCount, AutoTurnSetting delyPerTurn)
Starts the auto page turning animation.
(Returns true if it does not fail.)

void CancelPendingAutoTurns()
Cancels any pending auto turns that have not started yet.

void Build()
Builds the book.

Properties

bool isTurning { get; }
Indicates whether any page is currently turning (not including auto turning).

bool isAutoTurning { get; }
Indicates whether any page is currently auto turning (not including turning).

bool isFalling { get; }
Indicates whether any page is currently falling.

bool isIdle { get; }
Indicates whether all pages are idle (no turning, no auto turning, and no falling).

bool hasPendingAutoTurns { get; }
Indicates if there are pending auto turns that have not started yet.

float autoTurningEndTime { get; }
Gets the time at which the auto page turning animation will end.

bool isBuilt { get; }
Indicates whether the book has been built and is ready for use.

BookContent content { get; set; }
Gets or sets the content of the book.
Changing the content clears the book and requires calling the Build() method.

BookBinding binding { get; set; }
Gets or sets the binding of the book.
Changing the binding clears the book and requires calling the Build() method.

Static Properties

Book[] instances {get; }
Returns an array of Book objects.

Book Content

Properties

List<Object> covers { get; }
Returns a reference to the Covers list (not a copy).

List<Object> pages { get; }
Returns a reference to the Pages list (not a copy).

Note:
The cover and page elements must be of type Sprite, SpritePageContent, or LivePageContent. Adding elements of any other type is not supported and may cause errors. Additionally, modifying these lists does not affect the built book; you need to rebuild it by calling the book Build() method.

Page Content

Properties

bool isActive { get; }
Indicates whether the page is active, meaning it is visible or is about to be visible.

System.Action onActiveChangedCallback { get; }
The callback that is invoked when the active state of the page changes.

Live Page Canvas Raycaster

Static Properties

Camera camera {get; set;}

The currently active drawing camera that is viewing the book. It is required to handle mouse input for UI elements. If it is null, Camera.main is used instead. If Camera.main is also null, it throws an error. You should either set it or ensure that the tag of your camera is 'MainCamera' in the Inspector window.

Links & Contact Info

https://www.youtube.com/playlist?list=ProceduralBook

ScriptBoyTools@outlook.com

Have Fun!
Script Boy
:)