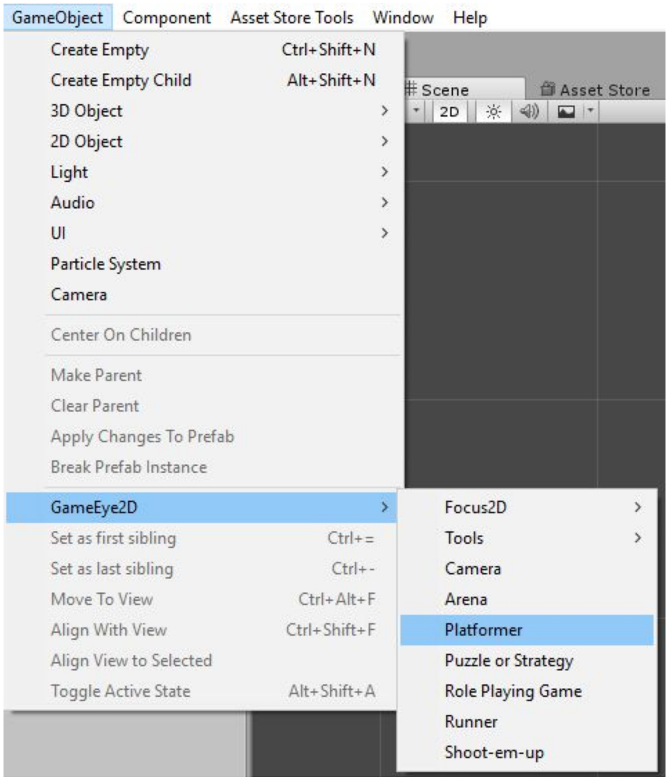
For Unity 5.4.1+
Current Version 1.0
Store Page
Support
Website

Table of Contents

Quick Start

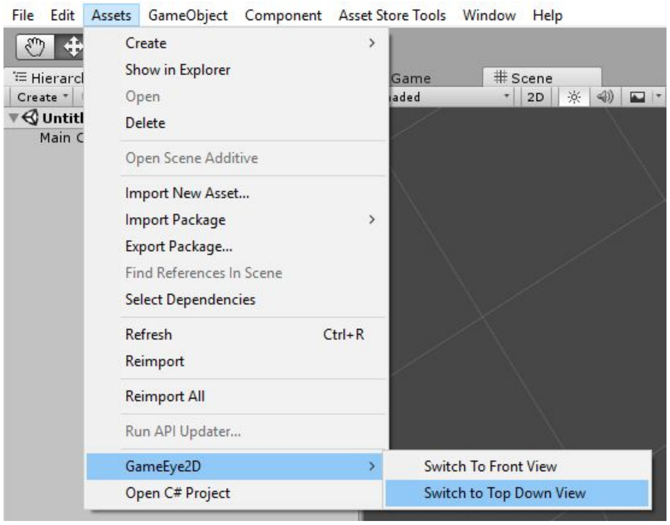
Game Eye Menu

When installed, the following menu is added which you can use to quickly set up a camera with some behaviors and focuses for your type of game.



Top Down Mode

For a top down view, you can change to use the XZ axis instead of XY using the following menu.

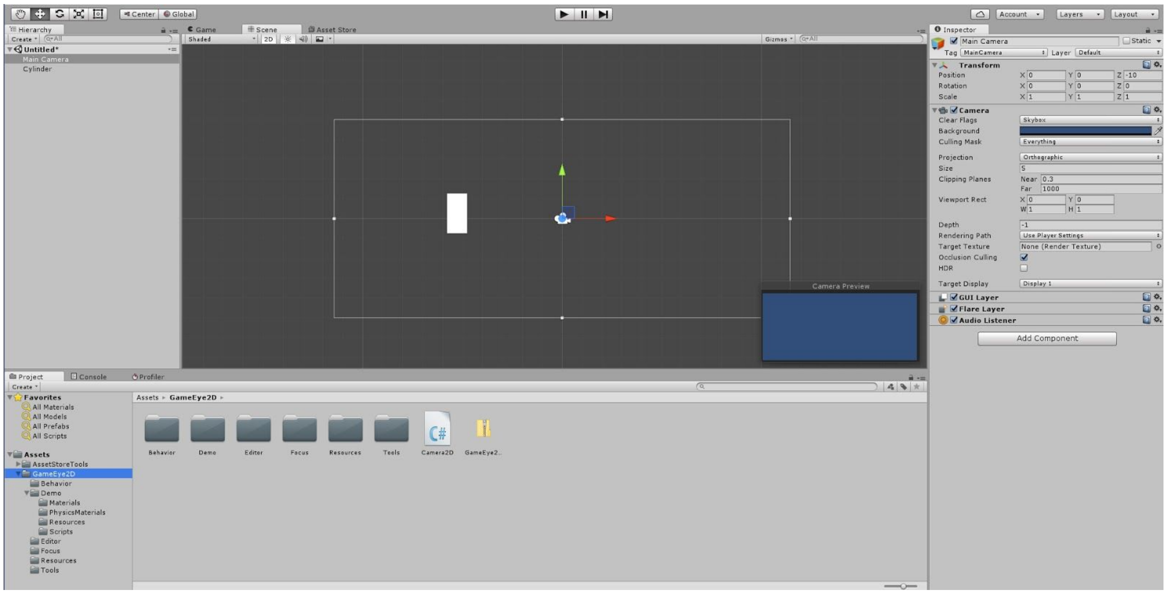


Custom Setup

If you have any trouble, Camera2D and each focus has a Draw Debug toggle while in the editor that will draw information related to the script.

Step 1 - Make a Camera

Create a camera in your scene.



Inspector

☑ Main Camera □ Static

Tag MainCamera Layer Default

Transform

Position X 0 Y 0 Z -10

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

☑ Camera

Clear Flags Skybox

Background

Culling Mask Everything

Projection Orthographic

Size 5

Clipping Planes Near 0.3 Far 1000

Viewport Rect X 0 Y 0 W 1 H 1

Depth -1

Rendering Path Use Player Settings

Target Texture None (Render Texture)

Occlusion Culling ☑

HDR □

Target Display Display 1

☑ GUI Layer

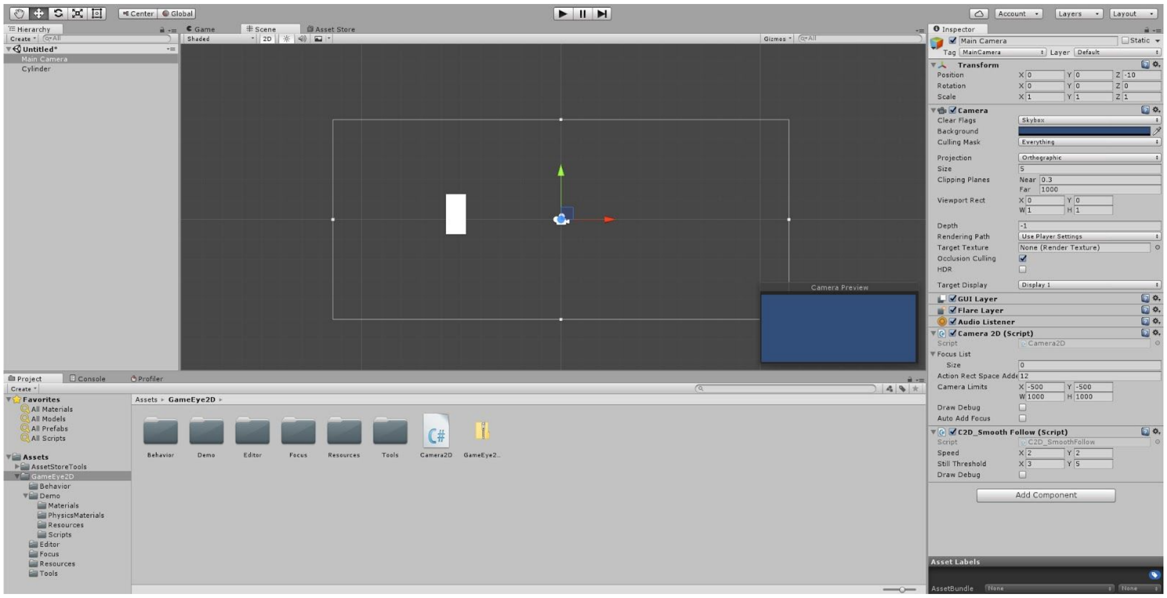
☑ Flare Layer

☑ Audio Listener

Add Component

Step 2 - Add a Camera Behavior

Add Smooth Follow or Fixed Follow to the camera. Camera2D will automatically be added.



Inspector

☑ Main Camera □ Static

Tag MainCamera Layer Default

Transform

Position X 0 Y 0 Z -10

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

☑ Camera

Clear Flags Skybox

Background

Culling Mask Everything

Projection Orthographic

Size 5

Clipping Planes Near 0.3 Far 1000

Viewport Rect X 0 Y 0 W 1 H 1

Depth -1

Rendering Path Use Player Settings

Target Texture None (Render Texture)

Occlusion Culling ☑

HDR □

Target Display Display 1

☑ GUI Layer

☑ Flare Layer

☑ Audio Listener

☑ Camera 2D (Script)

Script Camera2D

Focus List

Size 0

Action Rect Space Adde 12

Camera Limits X -500 Y -500 W 1000 H 1000

Draw Debug □

Auto Add Focus □

☑ C2D_Smooth Follow (Script)

Script C2D_SmoothFollow

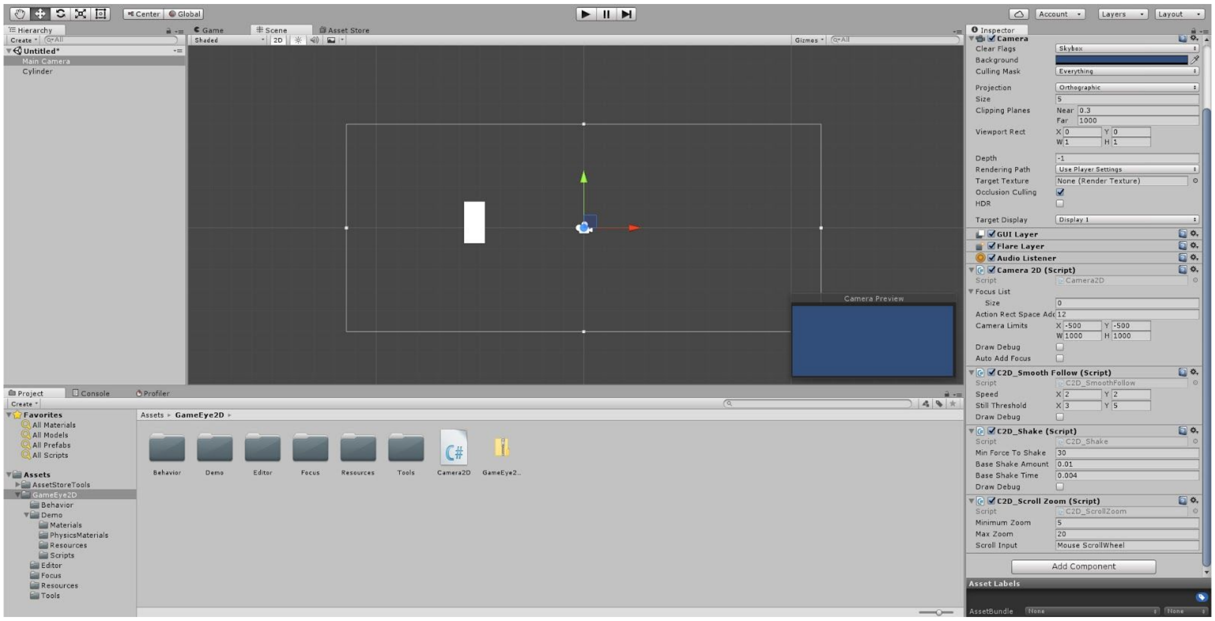
Speed X 2 Y 2

Still Threshold X 3 Y 5

Draw Debug □

Add Component

Additionally you may add Zoom To Fit or Scroll Zoom to allow the camera to zoom in or out, or Shake to allow you to call for the camera to shake through script.



Camera
Clear Flags: Skybox
Background
Culling Mask: Everything
Projection: Orthographic
Size: 5
Clipping Planes: Near 0.3, Far 1000
Viewport Rect: X 0, Y 0, W 1, H 1
Depth: -1
Rendering Path: Use Player Settings
Target Texture: None (Render Texture)
Occlusion Culling: ☑
HDR: ☐
Target Display: Display 1

GUI Layer
Flare Layer
Audio Listener

Camera 2D (Script)
Script: Camera2D
Focus List
Size: 0
Action Rect Space Add: 12
Camera Limits: X -500, Y -500, W 1000, H 1000
Draw Debug: ☐
Auto Add Focus: ☐

C2D_Smooth Follow (Script)
Script: C2D_SmoothFollow
Speed: X 2, Y 2
Still Threshold: X 3, Y 5
Draw Debug: ☐

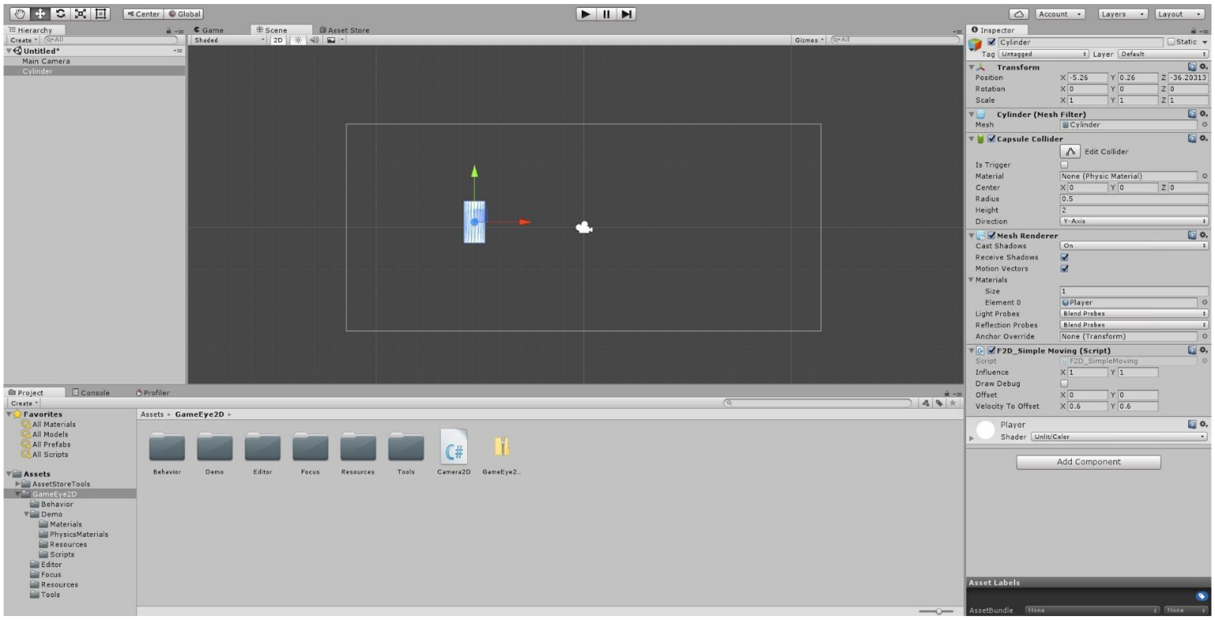
C2D_Shake (Script)
Script: C2D_Shake
Min Force To Shake: 30
Base Shake Amount: 0.01
Base Shake Time: 0.004
Draw Debug: ☐

C2D_Scroll Zoom (Script)
Script: C2D_ScrollZoom
Minimum Zoom: 5
Max Zoom: 20
Scroll Input: Mouse ScrollWheel

Step 3 - Set a Focus

Add any of the following focuses to a gameobject you want the camera to follow.

Transform, Simple Moving, Rigid Body, or Rigid Body 2D for the camera to follow the position of a game object.



☑ Cylinder ☐ Static
Tag: Untagged Layer: Default

Transform
Position: X -5.26, Y 0.26, Z -36.20313
Rotation: X 0, Y 0, Z 0
Scale: X 1, Y 1, Z 1

Cylinder (Mesh Filter)
Mesh: Cylinder

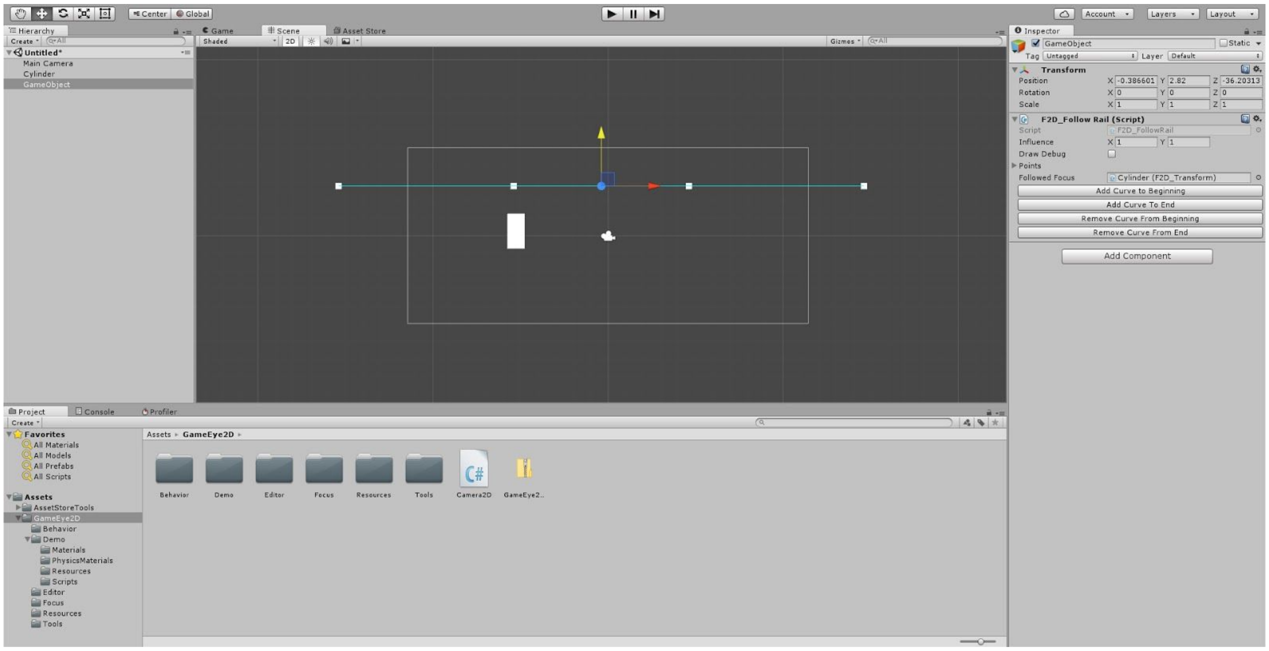
Capsule Collider
Is Trigger: ☐
Material: None (Physic Material)
Center: X 0, Y 0, Z 0
Radius: 0.5
Height: 2
Direction: Y-Axis

Mesh Renderer
Cast Shadows: On
Receive Shadows: ☑
Motion Vectors: ☑
Materials
Size: 1
Element 0: Player
Light Probes: Blend Probes
Reflection Probes: Blend Probes
Anchor Override: None (Transform)

F2D_Simple Moving (Script)
Script: F2D_SimpleMoving
Influence: X 1, Y 1
Draw Debug: ☐
Offset: X 0, Y 0
Velocity To Offset: X 0.6, Y 0.6

Player
Shader: Unlit/Color

Follow Rail, Follow Rail X, Follow Rail Y, or Timed Rail for the camera to follow a rail. Follow rails must have another focus provided to them to follow.

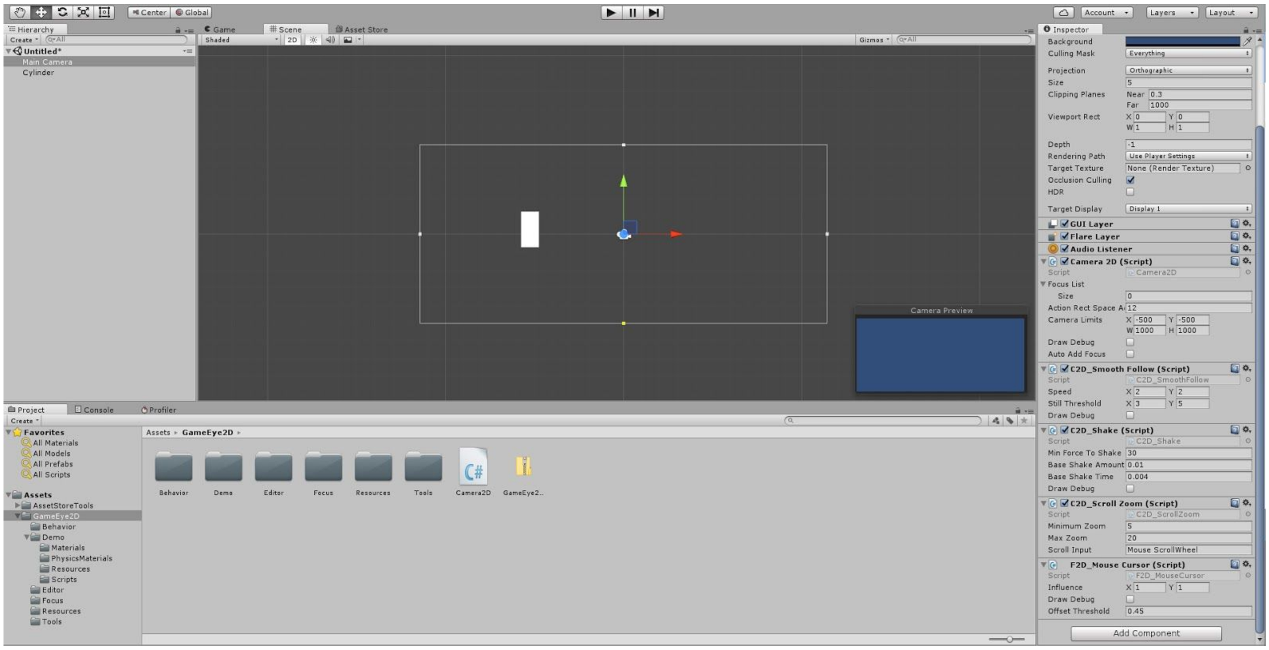


GameObject ☑ □ Static
Tag: Untagged Layer: Default

Transform
Position X -0.386601 Y 2.82 Z -36.20313
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

F2D_Follow Rail (Script)
Script: F2D_FollowRail
Influence X 1 Y 1
Draw Debug □
Points
Followed Focus: Cylinder (F2D_Transform)

Mouse Cursor, or Mouse Cursor Limited (these must on the camera object) for the camera to follow the position of the mouse cursor.



Background
Culling Mask: Everything
Projection: Orthographic
Size: 5
Clipping Planes: Near 0.3, Far 1000
Viewport Rect: X 0 Y 0 W 1 H 1
Depth: -1
Rendering Path: Use Player Settings
Target Texture: None (Render Texture)
Occlusion Culling ☑
HDR □
Target Display: Display 1

☑ GUI Layer
☑ Flare Layer
☑ Audio Listener

☑ Camera 2D (Script)
Script: Camera2D
Focus List
Size: 0
Action Rect Space A: 12
Camera Limits: X -500 Y -500 W 1000 H 1000
Draw Debug □
Auto Add Focus □

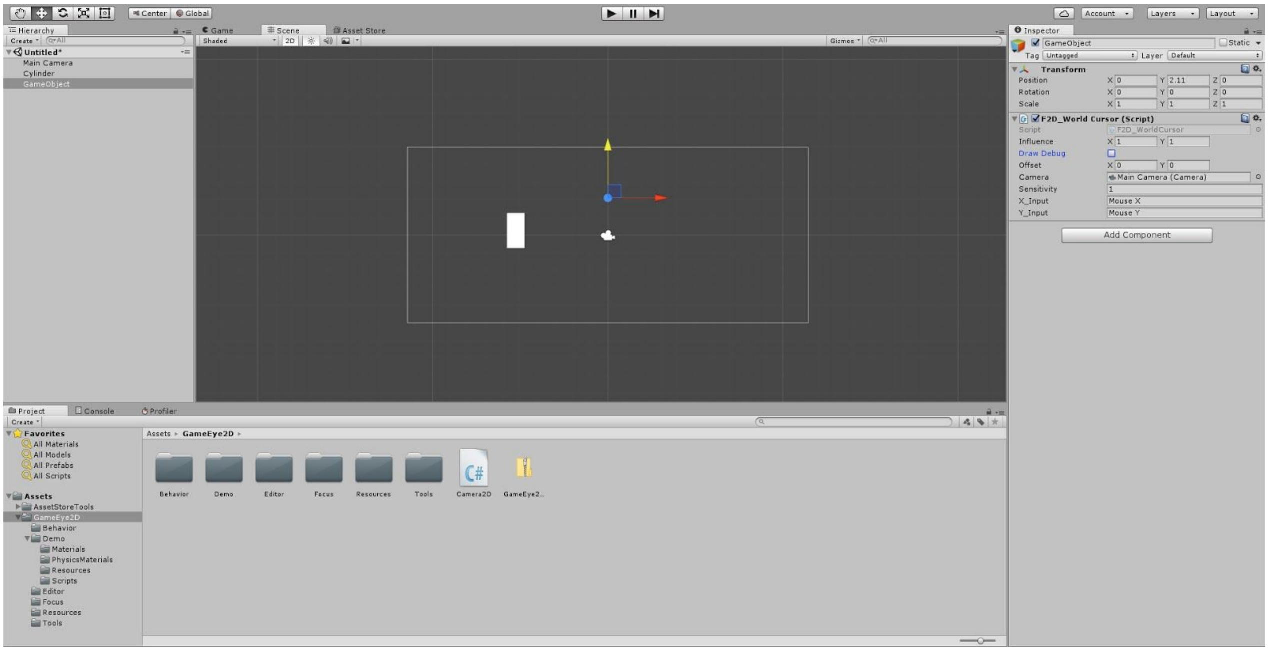
☑ C2D_Smooth Follow (Script)
Script: C2D_SmoothFollow
Speed X 2 Y 2
Still Threshold X 3 Y 5
Draw Debug □

☑ C2D_Shake (Script)
Script: C2D_Shake
Min Force To Shake: 30
Base Shake Amount: 0.01
Base Shake Time: 0.004
Draw Debug □

☑ C2D_Scroll Zoom (Script)
Script: C2D_ScrollZoom
Minimum Zoom: 5
Max Zoom: 20
Scroll Input: Mouse ScrollWheel

F2D_Mouse Cursor (Script)
Script: F2D_MouseCursor
Influence X 1 Y 1
Draw Debug □
Offset Threshold: 0.45

World Cursor, or World Cursor Limited for the camera to follow an object that is moved by the mouse’s movement.



☑ GameObject □ Static
Tag Untagged Layer Default

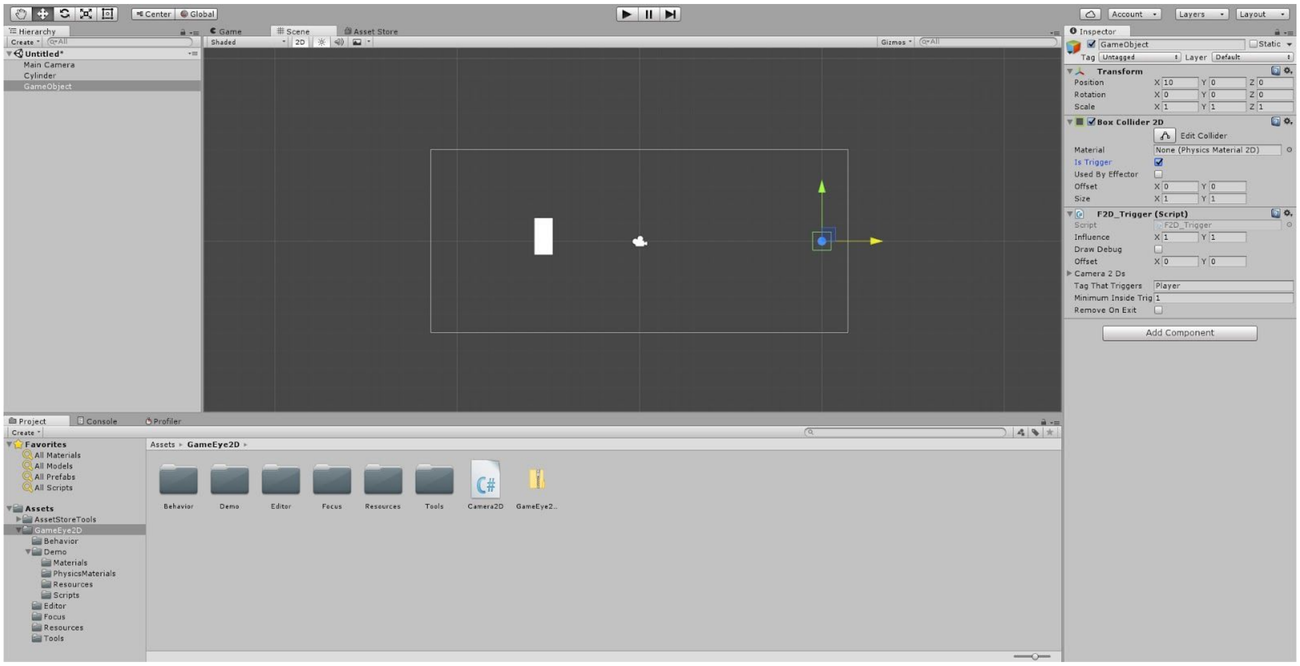
Transform

Position	X	0	Y	2.11	Z	0
Rotation	X	0	Y	0	Z	0
Scale	X	1	Y	1	Z	1

☑ F2D_World Cursor (Script)

Script	F2D_WorldCursor
Influence	X 1 Y 1
Draw Debug	□
Offset	X 0 Y 0
Camera	Main Camera (Camera)
Sensitivity	1
X_Input	Mouse X
Y_Input	Mouse Y

Trigger, or Timed Trigger for the camera to follow the position of a game object when a trigger collider is entered.



☑ GameObject □ Static
Tag Untagged Layer Default

Transform

Position	X	10	Y	0	Z	0
Rotation	X	0	Y	0	Z	0
Scale	X	1	Y	1	Z	1

☑ Box Collider 2D

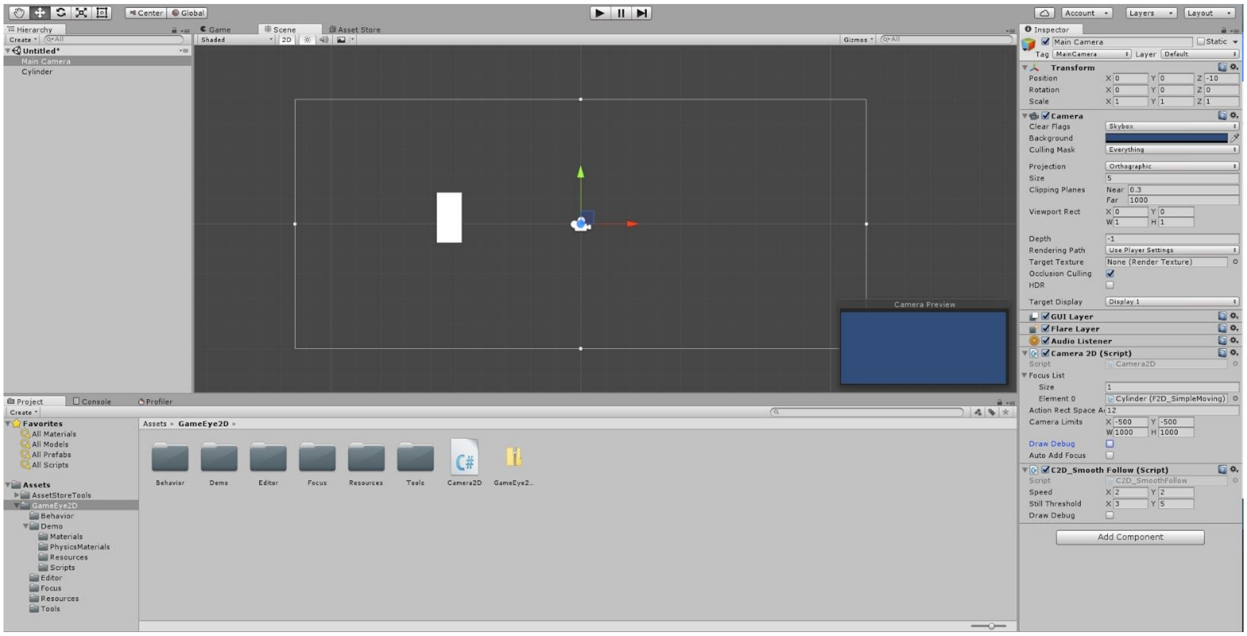
Material	None (Physics Material 2D)
Is Trigger	☑
Used By Effector	□
Offset	X 0 Y 0
Size	X 1 Y 1

F2D_Trigger (Script)

Script	F2D_Trigger
Influence	X 1 Y 1
Draw Debug	□
Offset	X 0 Y 0
▸ Camera 2 Ds	
Tag That Triggers	Player
Minimum Inside Trig	1
Remove On Exit	□

Step 4 - Give camera the focus

Add the focus to Camera2D’s list of followed focuses. When you hit play the camera should now follow the focus.



Inspector
☑ Main Camera □ Static
Tag MainCamera Layer Default

Transform
Position X 0 Y 0 Z -10
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

☑ Camera
Clear Flags Skybox
Background
Culling Mask Everything
Projection Orthographic
Size 5
Clipping Planes Near 0.3
Far 1000
Viewport Rect X 0 Y 0
W 1 H 1
Depth -1
Rendering Path Use Player Settings
Target Texture None (Render Texture)
Occlusion Culling ☑
HDR □
Target Display Display 1

☑ GUI Layer
☑ Flare Layer
☑ Audio Listener

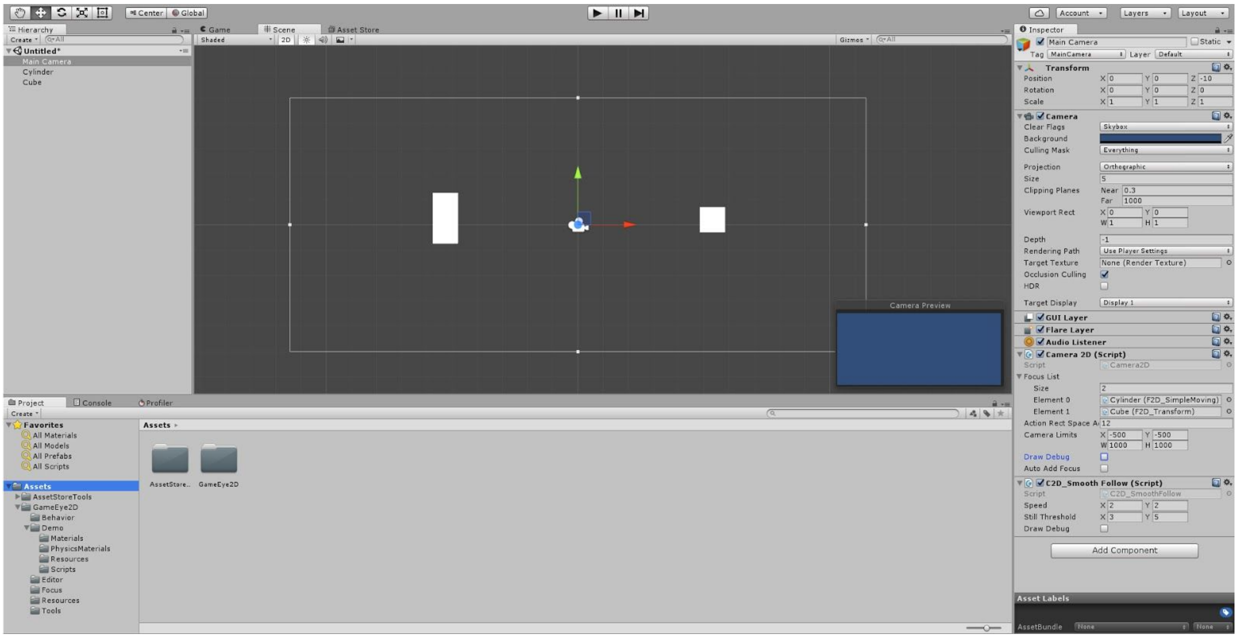
☑ Camera 2D (Script)
Script Camera2D
Focus List
Size 1
Element 0 Cylinder (F2D_SimpleMoving)
Action Rect Space A 12
Camera Limits X -500 Y -500
W 1000 H 1000
Draw Debug □
Auto Add Focus □

☑ C2D_Smooth Follow (Script)
Script C2D_SmoothFollow
Speed X 2 Y 2
Still Threshold X 3 Y 5
Draw Debug □

Add Component

Step 5 - Add More Focuses (optional)

Feel free to repeat step 3 and 4 until the camera follows every object that is important to be on camera.



Inspector
☑ Main Camera □ Static
Tag MainCamera Layer Default

Transform
Position X 0 Y 0 Z -10
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

☑ Camera
Clear Flags Skybox
Background
Culling Mask Everything
Projection Orthographic
Size 5
Clipping Planes Near 0.3
Far 1000
Viewport Rect X 0 Y 0
W 1 H 1
Depth -1
Rendering Path Use Player Settings
Target Texture None (Render Texture)
Occlusion Culling ☑
HDR □
Target Display Display 1

☑ GUI Layer
☑ Flare Layer
☑ Audio Listener

☑ Camera 2D (Script)
Script Camera2D
Focus List
Size 2
Element 0 Cylinder (F2D_SimpleMoving)
Element 1 Cube (F2D_Transform)
Action Rect Space A 12
Camera Limits X -500 Y -500
W 1000 H 1000
Draw Debug □
Auto Add Focus □

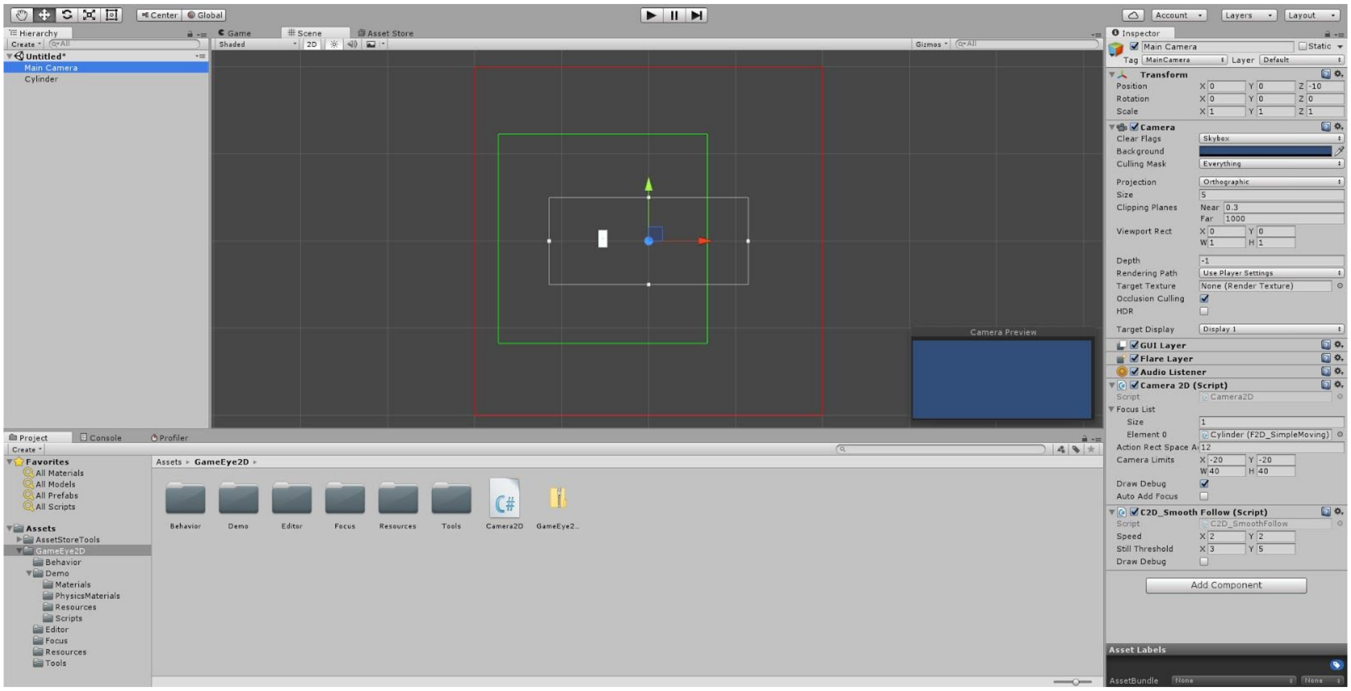
☑ C2D_Smooth Follow (Script)
Script C2D_SmoothFollow
Speed X 2 Y 2
Still Threshold X 3 Y 5
Draw Debug □

Add Component

Asset Labels
AssetBundle None None

Step 6 - Set Camera Limits (optional)

In the inspector for Camera2D you can set limits that the camera view will not leave. I highly suggest toggling on Draw Debug in the inspector so you can see the Camera Limits you are setting (this is the red rectangle in the following image).



☑ Main Camera ☐ Static

Tag MainCamera Layer Default

Transform

Position	X	0	Y	0	Z	-10
Rotation	X	0	Y	0	Z	0
Scale	X	1	Y	1	Z	1

☑ Camera

Clear Flags	Skybox
Background	
Culling Mask	Everything
Projection	Orthographic
Size	5
Clipping Planes	Near 0.3 Far 1000
Viewport Rect	X 0 Y 0 W 1 H 1
Depth	-1
Rendering Path	Use Player Settings
Target Texture	None (Render Texture)
Occlusion Culling	☑
HDR	☐
Target Display	Display 1

☑ GUI Layer

☑ Flare Layer

☑ Audio Listener

☑ Camera 2D (Script)

Script	Camera2D
Focus List	
Size	1
Element 0	Cylinder (F2D_SimpleMoving)
Action Rect Space A	12
Camera Limits	X -20 Y -20 W 40 H 40
Draw Debug	☑
Auto Add Focus	☐

☑ C2D_Smooth Follow (Script)

Script	C2D_SmoothFollow
Speed	X 2 Y 2
Still Threshold	X 3 Y 5
Draw Debug	☐

Add Component

AssetBundle None None

Step 7 - Change Camera and Focus Values (optional)

In the inspector window the options listed above have ways to customize each component. This allows you to set up camera movement that is specially tailored for your game.

Summary

GameEye2D is component based, and so the functionality is separated into 4 sections.

Camera2D

Camera2D is an orthographic camera extension that is uses the focuses in your world to calculate the optimal camera view. It is used by camera behaviors to move and zoom the camera.

Camera Behaviors

Behaviors move and zoom the camera either using the input of your player or the optimal view provided by Camera2D. You can choose which behaviors to add to a camera to determine how the camera will move. For example C2D_SmoothFollow will smoothly transition the camera's position to the center of the optimal view. All behaviors begin with the prefix C2D_

Fixed Follow

When attached to a transform with a Camera2D, causes the camera's position to be exactly at the center of the action. The behavior can be paused by disabling it.

Scroll Zoom

When attached to a transform with a Camera2D, causes the camera's orthographic size to change based on the input of the player. The most common use is to allow the mouse wheel to scroll the camera in and out. The behavior can be paused by disabling it.

Shake

When attached to a transform with a Camera2D, provides public functions that cause the camera's position to change as if a heavy force was applied. The behavior can be paused by disabling it.

Smooth Follow

When attached to a transform with a Camera2D, smoothly moves the camera's position towards the center of action. The behavior can be paused by disabling it.

Zoom To Fit

When attached to a transform with a Camera2D, smoothly zooms the camera’s view in or out to contain the action rect calculated by Camera2D. The behavior can be paused by disabling it.

Focuses

A focus calculates the point of interest of any gameobject they are attached to. Camera2D uses these points to determine the optimal view. You can choose what focus you want to add to change what the optimal view is considered to be. All focus2Ds begin with the prefix F2D_

Below are the Focus2D’s available in this package.

Follow Rail

Creates a rail system when attached to an object that follows another focus. The focus point of the rail is the nearest position on the rail to that focus.

Follow Rail X

Creates a rail system when attached to an object that follows another focus. The focus point of the rail is the nearest position on the rail to the X coordinate of that focus.

Follow Rail Y

Creates a rail system when attached to an object that follows another focus. The focus point of the rail is the nearest position on the rail to the Y coordinate of that focus.

Mouse Cursor

When attached to a camera, create a focus point that is at the position of the mouse cursor.

Mouse Cursor Limited

When attached to a camera, create a focus point that is equal to the position of the mouse cursor, clamped within a certain range of the transform this focus is attached to. This range can be set through the inspector.

Rigid Body

When attached to a transform with a rigidbody, creates a focus point that is at the position of the transform. If the rigidbody has a velocity, the focus point leads ahead of the transform in the direction of the velocity. The conversion from velocity to offset can be set through the inspector.

Rigid Body 2D

When attached to a transform with a rigidbody2D, creates a focus point that is at the position of the transform. If the rigidbody2D has a velocity, the focus point leads ahead of the transform in the direction of the velocity. The conversion from velocity to offset can be set through the inspector.

Simple Moving

When attached to a transform, creates a focus point that is at the position of the transform. If the transform moves, the focus point leads ahead of it. The conversion from velocity to offset can be set through the inspector.

Timed Rail

Creates a rail system when attached to an object. The focus point of the rail starts at the front and moves along the rail until it reaches the end. If looping is enabled through the inspector then the rail will start over once it reaches the end.

Timed Trigger

When attached to a transform with a trigger collider or collider2D, creates a focus that adds itself to the focus list of camera2Ds when an object with a certain tag enters the collider after a delay. There is an option for the camera2Ds to remove the focus when the trigger collider is exited.

Transform

When attached to a transform, creates a focus point that is at the position of the transform. This point can be offset through the inspector.

Trigger

When attached to a transform with a trigger collider or collider2D, creates a focus that adds itself to the focus list of camera2Ds when an object with a certain tag enters the collider. There is an option for the camera2Ds to remove the focus when the trigger collider is exited.

World Cursor

When added to a transform, creatures a focus that moves that transform based on the movement of two input axis (by default mouse X and Y). The normal unity cursor becomes hidden during gameplay. The axis that moves the transform can be set through the inspector.

World Cursor Limited

When added to a transform, creatures a focus that moves that transform based on the movement of two input axis (by default mouse X and Y) and leashes it to not be farther away from the transform's parent than the leash distance. The normal unity cursor becomes hidden during gameplay. The axis that moves the transform can be set through the inspector.

Tools

Tools are simple scripts that I would likely use for a game using GameEye2D.

ClampToScreen

When attached to a transform with a renderer, the transform's position will become clamped to remain within an orthographic camera view. For smooth movement it is suggested to have this script execute after camera movement has completed using script execution order settings.

Parallax

When attached to a transform, the transform's position will move to follow the camera. How much influence the camera's movement has on the transform can be changed to give the parallax effect.

Trigger Camera Limits

When attached to a transform with a trigger collider, when that collider is entered the limits of a Camera2D will change. The transition can be instant or over time.

Trigger Cameras

When attached to a transform with a trigger collider, when that collider is entered certain camera's will be enabled, and others disabled. You can choose which cameras are enabled or disabled.

Trigger Focus List

When attached to a transform with a trigger collider, when that collider is entered the focus list of a Camera2D is changed to a new list.

Trigger Zoom

When attached to a transform with a trigger collider, when that collider is entered the camera will begin to change orthographic sizes, zooming the camera in or out.

Scripting Reference

Camera2D

Public class Camera2D : MonoBehavior

Summary

When added to an orthographic camera, provides helper functions and collects information for making camera behaviors.

The main function of camera2D is the ability to track where the action of the game is and an orthographic view that would optimally view the action, known as the action rect. You can see the action rect in your scene window by toggling on Draw Debug in the inspector window.

Through the inspector you may add a focus2D component to a gameobject. A focus's main function is to calculate how that object's position wants to be tracked by Camera2D. Then you may then drag the focus into the camera2D's focus list. All focus's in the list are used to calculate the action rect.

Among others, the most used helper function provided by camera2D is known as the position2D property. When used to set the camera position on the 2D axis position2D will clamp the position of the camera within the camera limits you have set through the camera2D's inspector window.

Public Properties

gameCamera	Camera	The camera attached to the Camera2D.
focusArray	Focus2D[]	The focus's used to calculate the action rect.
position2D	Vector2	The position of the camera. When set, the position the camera will be clamped to not view outside of the camera limits.
localPosition2D	Vector2	The local position of the camera. When set, the position the camera will be clamped to not view outside of the camera limits.
Zoom	float	The orthographic size of the camera. When set, the position the camera will be clamped to not view outside of the camera limits.
actionRect	Rect	A rectangle that is update each frame it is called to be centered on the focus's in the focus array. The rectangle is large enough to contain every focus in

		the array, with the addition of space added on each side determined by the addedActionRectSpace.
addedActionRectSpace	float	How much space is added to the top, bottom, left, and right of the action rectangle. The total width and height is increased by double this amount.
cameraLimits	Rect	When setting position2D, localPosition2D, or zoom the view of the camera will be clamped to be within this rectangle.

Public Functions

AddFocus (Focus2D focus) AddFocus(Focus2D[] focus)	void	Add the focus(s) to the list.

RemoveFocus(Focus2D focus)
RemoveFocus(Focus2D[] focus)
	void	Removes the focus(s) from the list.

CleanFocusList()	void	Removes all null focus's from the list.
ClearFocusList()	void	Removes all focus's from the list.
ClampToCameraLimits (Vector2 position) ClampToCameraLimits (Vector3 position) ClampToCameraLimits (Rect rect) ClampToCameraLimits (Bounds bounds)	Vector2 Vector3 Vector2 Vector3	Clamps the given position, rect, or bounds to the camera limits.
ClampToCameraView (Vector2 position) ClampToCameraView (Vector3 position) ClampToCameraView (Rect rect) ClampToCameraView (Bounds bounds)	Vector2 Vector3 Vector2 Vector3	Clamps the given position, rect, or bounds to the camera view.
FocusListContains (Focus2D focus)	bool	Returns if the focus is in the focus array.
MaximumZoom (Vector2 size)	float	Calculates the maximum orthographic size of the camera that would keep the view inside a rectangle of the given size.
UpdateActionRect ()	void	If the action rect has already updated

		on this frame, this function causes the action rect to update again.
WorldToZoom(Vector2 size)	float	Calculates the maximum orthographic size of the camera that would keep a rect of the given size inside the camera's view.

Static Functions

ClampToCameraView (Camera camera, Vector2 position) ClampToCameraView (Camera camera, Vector3 position) ClampToCameraView (Camera camera, Rect rect) ClampToCameraView (Camera camera, Bounds bounds)	Vector2 Vector3 Vector2 Vector3	Clamps the given position, rect, or bounds to a camera view and returns the new center position.
GetViewRectAsWorldSpace (Camera camera)	Rect	Calculates the view of the given camera as a rectangle in worldspace coordinates.
MaximumZoom (Vector2 size, float aspectRatio)	float	Calculates the maximum orthographic size of a view of the given aspect ratio that would keep the view inside a rectangle of the given size.
WorldToZoom (Vector2 size, float aspectRatio)	float	Calculates the maximum orthographic size that would keep a rect of the given size inside the view of the given aspect ratio.

Inspector Variables

The following variables are available in the inspector window.

Draw Debug	bool	When toggled on and the object is selected, the action rect and camera limits will be drawn in the scene window.
Auto Add Focus	bool	When toggled on, the camera2D will automatically add a focus2D that is added to the scene.

Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

Clean Focus List	Removes all null focus's from the focus list.
Clear Focus List	Removes all focus's from the focus list.
Add All Focus In Scene To List	Adds every focus in the scene to the focus list.
Set Position To Action Center	Sets the position of the camera to the center of the action rect.
Log Action Rect	Logs the current action rect in the console.
Log Maximum Zoom	Logs the maximum zoom possible with the current camera limits in the console.
Log View Rect As World Space	Logs the current action rect in the console.

Public Editor Properties

These only work in the scene editor. Do not call them while the game is running.

autoAddFocus	When toggled on, the camera2D will automatically add a focus2D that is added to the scene.

Camera Behaviours

FixedFollow

Public class GameEye2D.Behaviour.FixedFollow : MonoBehavior

Summary

When attached to a transform with a Camera2D, causes the camera's position to be exactly at the center of the action. The behavior can be paused by disabling it.

Public Properties

camera2D	Camera2D	The Camera2D used by this behavior.

ScrollZoom

Public class GameEye2D.Behaviour.ScrollZoom : MonoBehavior

Summary

When attached to a transform with a Camera2D, causes the camera's orthographic size to change based on the input of the player. The most common use is to allow the mouse wheel to scroll the camera in and out. The behavior can be paused by disabling it.

Public Properties

camera2D	Camera2D	The Camera2D used by this behavior.
minimumZoom	float	The minimum orthographic size possible by scrolling the camera.

maximumZoom	float	The maximum orthographic size possible by scrolling the camera.
scrollInput	string	The input used to scroll the camera.

Shake

Public class GameEye2D.Behaviour.Shake : MonoBehavior

Summary

When attached to a transform with a Camera2D, provides public functions that cause the camera's position to change as if a heavy force was applied. The behavior can be paused by disabling it.

Public Properties

camera2D	Camera2D	The Camera2D used by this behavior.
minimumForceToShake	float	The minimum force required to shake the camera. This amount is subtracted from all shake forces.
isShaking	bool	If the camera is shaking.
shakeOffset	Vector2	How much the camera is offset due to the camera shaking.

Public Functions

ShakeCamera (float magnitude) ShakeCamera (Vector2 force)	void	Causes the camera to shake.
EndShake ()	void	If the camera is currently shaking, it stops shaking.

Inspector Variables

The following variables are available in the inspector window.

Draw Debug	bool	When toggled on, a line is drawn from the camera's offset position to where it would be otherwise.

Inspector Variables

The following variables are available in the inspector window.

Draw Debug	bool	When toggled on and the object is selected, a line is drawn from the camera’s position to the position the camera is moving to.

SmoothFollow

Public class GameEye2D.Behaviour.SmoothFollow : MonoBehavior

Summary

When attached to a transform with a Camera2D, smoothly moves the camera's position towards the center of action. The behavior can be paused by disabling it.

Public Properties

camera2D	Camera2D	The Camera2D used by this behavior.
speed	Vector2	How fast the camera's position changes to follow the action rect.
stillThreshold	Vector2	How much the action rect must move in order to update the desired position to move to. The X axis of the threshold is altered based on the aspect ratio of the camera.

Inspector Variables

The following variables are available in the inspector window.

Speed	Vector2	How fast the position changes to follow the action.
Still Threshold	Vector2	How much the action rect must move in order to update the desired position to move to. The X axis of the threshold is altered based on the aspect ratio of the camera.
Draw Debug	bool	When toggled on, a line is drawn from the camera’s position to the position the camera is moving to.

SmoothScrollZoom

Public class GameEye2D.Behaviour.SmoothScrollZoom : C2D_ScrollZoom

Summary

When attached to a transform with a Camera2D, causes the camera's orthographic size to smoothly change based on the input of the player. The most common use is to allow the mouse wheel to scroll the camera in and out. The behavior can be paused by disabling it.

Public Properties

camera2D	Camera2D	The Camera2D used by this behavior.
minimumZoom	float	The minimum orthographic size possible by scrolling the camera.
maximumZoom	float	The maximum orthographic size possible by scrolling the camera.
scrollInput	string	The input used to scroll the camera.
speed	float	How fast the camera zooms in and out to follow the input scroll.

Inspector Variables

The following variables are available in the inspector window.

Speed	float	How fast the camera zooms in and out to follow the input scroll.

ZoomToFit

Public class GameEye2D.Behaviour.ZoomToFit : MonoBehavior

Summary

When attached to a transform with a Camera2D, smoothly zooms the camera's view in or out to contain the action rect calculated by Camera2D. The behavior can be paused by disabling it.

Public Properties

camera2D	The Camera2D used by this behavior.
speed	How fast the camera zooms in or out.
stillThreshold	How much the action rect calculated by Camera2D must expand or shrink in order to update the desired zoom.
minimumZoom	The lowest that ZoomToFit will update the camera's orthographic size to.

Inspector Variables

The following variables are available in the inspector window.

Speed	float	How fast the camera's position changes to follow the action rect.
Still Threshold	float	How much the action rect must move in order to update the desired position to move to. The X axis of the threshold is altered based on the aspect ratio of the camera.
Minimum Zoom	float	The lowest that ZoomToFit will update the camera's orthographic size to.
Draw Debug	bool	When toggled on, 2 rectangles are drawn representing how much the action rect must change to update the desired orthographic size.

Focuses

Focus2D

Public abstract class GameEye2D.Focus.Focus2D : MonoBehavior

Summary

An abstract class that focuses inherit from, allowing the camera to use inheriting classes as a focus when calculating the action rect.

Focus2Ds calculate the point of interest of any gameobject they are attached to. Camera2D uses these points to determine the optimal view.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.

Public Abstract Functions

GetFocusPoint ()	Vector2	A public function for the camera to use to find this object's point of interest. When making your own focus's you must implement this function.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

FollowRail

Public class GameEye2D.Focus.FollowRail : F2D_Rail

Summary

Creates a rail system when attached to an object that follows another focus. The focus point of the rail is the nearest position on the rail to that focus.

Public Properties

position2D	Vector2	The position of the focus as a vector2.
localPosition2D	Vector2	The local position of the focus as a vector2.
Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
followedFocus	Focus2D	The focus the rail is following.

Public Functions

GetFocusPoint ()	Vector2	Calculates the focus point along the rail.
ClampToCurve (Vector2 position)	Vector2	Calculates the nearest Vector2 position along the curve to the given position.

Inspector Variables

The following variables are available in the inspector window.

Points	Vector2[]	The points that make the rail.
Followed Focus	Focus2D	The focus the rail is following.
Draw Debug	bool	When toggled on, the focus point is drawn in the scene window as a cyan ball.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

Protected Functions

These only work in the scene editor. Do not call them while the game is running.

ValidatePoints ()	void	If there are not 4, 7, 10, etc... points on the rail this function will add enough points to complete a curve.

FollowRailX

Public class GameEye2D.Focus.FollowRailX : Focus2DFollowRail

Summary

Creates a rail system when attached to an object that follows another focus. The focus point of the rail is the nearest position on the rail to the X coordinate of that focus.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
followedFocus	Focus2D	The focus the rail is following.

Public Functions

GetFocusPoint ()	Vector2	Calculates the focus point along the rail.
ClampToCurveX (float position)	Vector2	Calculates the nearest position along the curve to the given X coordinate.

Inspector Variables

The following variables are available in the inspector window.

Points	Vector2[]	The points that make the rail.
Followed Focus	Focus2D	The focus the rail is following.
Draw Debug	bool	When toggled on, the focus point is drawn in the scene window as a cyan ball.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

Protected Functions

These only work in the scene editor. Do not call them while the game is running.

ValidatePoints ()	void	If there are not 4, 7, 10, etc... points on the rail this function will add enough points to complete a curve.

FollowRailY

Public class GameEye2D.Focus.FollowRailY : Focus2DFollowRail

Summary

Creates a rail system when attached to an object that follows another focus. The focus point of the rail is the nearest position on the rail to the Y coordinate of that focus.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
followedFocus	Focus2D	The focus the rail is following.

Public Functions

GetFocusPoint ()	Vector2	Calculates the focus point along the rail.
ClampToCurveY (float position)	Vector2	Calculates the nearest position along the curve to the given Y coordinate.

Inspector Variables

The following variables are available in the inspector window.

Points	Vector2[]	The points that make the rail.
Followed Focus	Focus2D	The focus the rail is following.
Draw Debug	bool	When toggled on, the focus point is drawn in the scene window as a cyan ball.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

Protected Functions

These only work in the scene editor. Do not call them while the game is running.

ValidatePoints ()	void	If there are not 4, 7, 10, etc... points on the rail this function will add enough points to complete a curve.

MouseCursor

Public class GameEye2D.Focus.MouseCursor : Focus2D

Summary

When attached to a camera, create a focus point that is at the position of the mouse cursor.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
gameCamera	Camera	The camera the focus is attached to.

Public Functions

GetFocusPoint ()	Vector2	Calculates the focus point position of the mouse cursor.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on, the focus point is drawn in the scene window as a cyan ball.
Offset Threshold	float	How far the mouse must be from the center of the screen (0 - 0.5) to return a value that is not the center of the screen.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

Rail

Public abstract class GameEye2D.Focus.Rail : Focus2D

Summary

Creates a rail system when attached to an object. The class is abstract, and can be inherited from to create rail-based focus's.

Public Properties

position2D	Vector2	The position of the focus as a vector2.
localPosition2D	Vector2	The local position of the focus as a vector2.
Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
points	Vector2[]	The points that are used to calculate the rail's bezier curve.
curveCount	int	The number of curves in the rail.
numberOfPoints	int	The number of points in the rail.

Public Abstract Functions

GetFocusPoint ()	Vector2	When overridden by inheriting classes, Camera2D can use this function to determine a Vector2 point that is should be in view of the camera.

Public Functions

GetPoint (float interpolation, int curve)	Vector 2	Gets a point along a curve in the rail.

Inspector Variables

The following variables are available in the inspector window.

Points	Vector2[]	The points that make the rail.
Draw Debug	bool	When toggled on, the focus point is drawn in the scene window as a cyan ball.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

Protected Functions

These only work in the scene editor. Do not call them while the game is running.

ValidatePoints ()	void	If there are not 4, 7, 10, etc... points on the rail this function will add enough points to complete a curve.

F_RigidBody

Public class GameEye2D.Focus.F_RigidBody : F2D_Transform

Summary

When attached to a transform with a rigidbody, creates a focus point that is at the position of the transform. If the rigidbody has a velocity, the focus point leads ahead of the transform in the direction of the velocity. The conversion from velocity to offset can be set through the inspector.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
body	RigidBody	The rigidbody of the focus.

Public Functions

GetFocusPoint ()	Vector2	Calculates a position that is ahead of the focus's transform, in the direction the rigidybody it moving.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector2	An offset added to the position of the focus point.
Velocity To	Vector	How far the focus point will lead the movements of the

Offset	2	rigidbody.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

F_RigidBody2D

Public class GameEye2D.Focus.F_RigidBody2D : F2D_Transform

Summary

When attached to a transform with a rigidbody2D, creates a focus point that is at the position of the transform. If the rigidbody2D has a velocity, the focus point leads ahead of the transform in the direction of the velocity. The conversion from velocity to offset can be set through the inspector.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
body2D	RigidBody2D	The rigidbody2D of the focus.

Public Functions

GetFocusPoint ()	Vector2	Calculates a position that is ahead of the focus's transform, in the direction the rigidybody2D it moving.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector2	An offset added to the position of the focus point.

Velocity To Offset	Vector 2	How far the focus point will lead the movements of the rigidbody2D.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

SimpleMoving

Public class GameEye2D.Focus.SimpleMoving : F2D_Transform

Summary

When attached to a transform, creates a focus point that is at the position of the transform. If the transform moves, the focus point leads ahead of it. The conversion from velocity to offset can be set through the inspector.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.

Public Functions

GetFocusPoint ()	Vector2	Calculates a position that is ahead of the focus's transform, in the direction the transform it moving.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector 2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector 2	An offset added to the position of the focus point.
Velocity To Offset	Vector 2	How far the focus point will lead the movements of the transform.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

F_Transform

Public class GameEye2D.Focus.F_Transform : Focus2D

Summary

When attached to a transform, creates a focus point that is at the position of the transform. This point can be offset through the inspector.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.

Public Functions

GetFocusPoint ()	Vector2	Returns a position that is at the transform's position.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector 2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector 2	An offset added to the position of the focus point.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

TimedTrigger

Public class GameEye2D.Focus.TimedTrigger : F2D_Trigger

Summary

When attached to a transform with a trigger collider or collider2D, creates a focus that adds itself to the focus list of camera2Ds when an object with a certain tag enters the collider after a delay. There is an option for the camera2Ds to remove the focus when the trigger collider is exited.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
camera2Ds	Camera2D[]	The camera2Ds that will add and remove the focus from a focus list.

Public Functions

GetFocusPoint ()	Vector2	Returns the position of the focus's transform.
Activate()	void	Adds the focus to the cameras it would be added to by the trigger collider.
Undo()	void	Removes the focus from the cameras it has been added to by the trigger collider.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector2	An offset added to the position of the focus point.
Camera2Ds	Camera2D[]	The Camera2Ds that will add and remove the focus from a focus list.
Tag That Triggers	string	An object setting off the trigger must have this tag.
Minimum Inside Trigger	int	The minimum number of objects with the correct tag that must be within the trigger for the trigger to be set off.
Remove On Exit	bool	If the focus is removed from the Camera2Ds when the trigger is exited.
Time To Activate	float	How long it takes for the trigger to activate.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

F_Transform

Public class GameEye2D.Focus.F_Transform : Focus2D

Summary

When attached to a transform, creates a focus point that is at the position of the transform. This point can be offset through the inspector.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.

Public Functions

GetFocusPoint ()	Vector2	Returns a position that is at the transform's position.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector 2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector 2	An offset added to the position of the focus point.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

Trigger

Public class GameEye2D.Focus.Trigger : F2D_Transform

Summary

When attached to a transform with a trigger collider or collider2D, creates a focus that adds itself to the focus list of camera2Ds when an object with a certain tag enters the collider. There is an option for the camera2Ds to remove the focus when the trigger collider is exited.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
camera2Ds	Camera2D[]	The camera2D's that will add and remove the focus from a focus list.

Public Functions

GetFocusPoint ()	Vector2	Returns the position of the focus's transform.
Activate()	void	Adds the focus to the cameras it would be added to by the trigger collider.
Undo()	void	Removes the focus from the cameras it has been added to by the trigger collider.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector2	An offset added to the position of the focus point.
Camera2Ds	Camera2D[]	The camera2Ds that will add and remove the focus from a focus list.
Tag That Triggers	string	An object setting off the trigger must have this tag.
Minimum Inside Trigger	int	The minimum number of objects with the correct tag that must be within the trigger for the trigger to be set off.
Remove On Exit	bool	If the focus is removed from the camera2Ds when the trigger is exited.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

WorldCursor

Public class GameEye2D.Focus.WorldCursor : F2D_Transform

Summary

When added to a transform, creatures a focus that moves that transform based on the movement of two input axis (by default mouse X and Y). The normal unity cursor becomes hidden during gameplay. The axis that moves the transform can be set through the inspector.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
sensitivity	float	How much the transform of the focus moves to follow the input.

Public Functions

GetFocusPoint ()	Vector2	Returns the position of the world cursor’s transform.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector2	An offset added to the position of the focus point.
Camera	Camera	The camera the cursor’s initial position is based off of.
Sensitivity	float	How much the transform of the focus moves to follow the input.
X Input	string	The input that moves the cursor along the X axis.
Y Input	string	The input that moves the cursor along the Y axis.

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

WorldCursorLimited

Public class GameEye2D.Focus.WorldCursorLimited : F2D_WorldCursor

Summary

When added to a transform, creatures a focus that moves that transform based on the movement of two input axis (by default mouse X and Y) and leashes it to not be farther away from the transform's parent than the leash distance. The normal unity cursor becomes hidden during gameplay. The axis that moves the transform can be set through the inspector.

Public Properties

position2D	Vector2	The position of the focus.
localPosition2D	Vector2	The local position of the focus.
influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
sensitivity	float	How much the transform of the focus moves to follow the input.

Public Functions

GetFocusPoint ()	Vector2	Returns the position of the world cursor's transform.

Inspector Variables

The following variables are available in the inspector window.

Influence	Vector2	How much the action rect calculated by Camera2D is influenced by this focus.
Draw Debug	bool	When toggled on and the object is selected, the focus point is drawn in the scene window as a cyan ball.
Offset	Vector2	An offset added to the position of the focus point.
Camera	Camera	The camera the cursor's initial position is based off of.
Sensitivity	float	How much the transform of the focus moves to follow the input.
X Input	string	The input that moves the cursor along the X axis.
Y Input	string	The input that moves the cursor along the Y axis.

Leash Distance	float	How far the cursor can be from this transform's parent

Protected Inspector Functions

The following functions are available in the inspector window by right clicking on the component.

These only work in the scene editor. Do not call them while the game is running.

Add Focus To Every Camera2D	Adds this focus to every Camera2D in the scene.
Add Focus To Every Auto Add Camera2D	Adds this focus to every Camera2D in the scene with auto add focus enabled.
Remove Focus From Every Camera2D	Remove this focus from every Camera2D in the scene.
Remove Focus From Every Auto Add Camera2D	Remove this focus from every Camera2D in the scene with auto add focus enabled.
Set Origin to (0, 0)	Sets the local 2D position of the focus to (0,0).
Set Rotation to Identity	Sets the local rotation of the focus to (0,0,0).

Tools

ClampToScreen

Public class ClampToScreen : MonoBehavior

Summary

When attached to a transform with a renderer, the transform’s position will become clamped to remain within an orthographic camera view. For smooth movement it is suggested to have this script execute after camera movement has completed using script execution order settings.

Public Properties

gameCamera	Camera	The camera that the object will be clamped to.

Inspector Variables

The following variables are available in the inspector window.

Camera	Camera	The camera that the object will be clamped to.

Parallax

Public class Parallax : MonoBehavior

Summary

When attached to a transform, the transform's position will move to follow the camera. How much influence the camera's movement has on the transform can be changed to give the parallax effect.

Public Properties

cameraTransform	Transform	The transform of the camera that will cause the parallax object to move.
influence	Vector2	How much the transform will be moved by the movement of the camera.
parallaxPosition	Vector3	Where the object will be according to the current camera position.

Inspector Variables

The following variables are available in the inspector window.

Camera Transform	Transform	The camera that the transform is moving in accordance to.
Influence	Vector2	How much the transform will be moved by the movement of the camera.

TriggerCameraLimits

Public abstract class TriggerCameraLimits : MonoBehavior

Summary

When attached to a transform with a trigger collider, when that collider is entered the limits of a Camera2D will change. The transition can be instant or over time.

Public Functions

Activate	void	Causes the camera limits to begin to change.
ResetTrigger	void	If already triggered, reenables the trigger.

Inspector Variables

The following variables are available in the inspector window.

Camera2D	Camera2D	The Camera2D that will have its camera limits changed.
Tag That Triggers	string	An object setting off the trigger must have this tag.
Camera Limits	Rect	The new limits being set.
Transition Time	float	How long it takes for the limits to change. It can be instant or over time.
Draw Debug	bool	When toggled on, the limits of the Camera2D and the limits they will change to are drawn in the scene window.

TriggerCameras

Public class TriggerCameras : MonoBehavior

Summary

When attached to a transform with a trigger collider, when that collider is entered certain camera’s will be enabled, and others disabled. You can choose which cameras are enabled or disabled.

Public Functions

Activate	void	Causes the cameras to change.

Inspector Variables

The following variables are available in the inspector window.

EnabledCameras	Camera[]	The cameras that are enabled when the trigger is entered.
DisabledCameras	Camera[]	The cameras that are disabled when the trigger is entered.
Tag That Triggers	string	An object setting off the trigger must have this tag.

TriggerFocusList

Public class TriggerFocusList : MonoBehavior

Summary

When attached to a transform with a trigger collider, when that collider is entered the focus list of a Camera2D is changed to a new list.

Public Functions

Activate	void	Causes the camera limits to begin to change.

Inspector Variables

The following variables are available in the inspector window.

Camera2D	Camera2D	The Camera2D that will have its camera limits changed.
New Focus List	Focus2D[]	The focus list that the Camera2D will change to.
Tag That Triggers	string	An object setting off the trigger must have this tag.

TriggerZoom

Public class TriggerZoom : MonoBehavior

Summary

When attached to a transform with a trigger collider, when that collider is entered the camera will begin to change orthographic sizes, zooming the camera in or out.

Public Functions

Activate	void	Causes the orthographics size of a camera to begin to change.

Inspector Variables

The following variables are available in the inspector window.

Camera2D	Camera2D	The Camera2D that will have its orthographics size changed.
Tag That Triggers	string	An object setting off the trigger must have this tag.
EnterZoom	float	The new orthographics size being set when the trigger is entered.
ExitZoom	float	The new orthographics size being set when the trigger is exited.
Transition Time	float	How long it takes for the orthographics size to change. It can be instant or over time.
Draw Debug	bool	When toggled on, the enter zoom and the exit zoom are drawn as rectangles in the scene window.

Building an Extension

Camera2D is component based, so you can add and replace parts with ease. It is built in C#.

Make a Camera Behavior

Your custom camera behaviors will work with Camera2D. Camera2D offers a number of helper functions that will help your custom behavior.

Let's start with a simple behavior. One that causes the camera to follow a focus in its list. First we need a camera, Camera2D, and our custom behavior attached to the object through the inspector.

Then through script we can get get the Camera2D to help us move the camera.

```
using UnityEngine;

[ RequireComponent( typeof( Camera2D )) ]
public class CustomBehavior : MonoBehaviour
{
	Camera2D m_Camera2D;

	void Start ()
	{
		m_Camera2D = GetComponent<Camera2D>();
	}
}
```

Then we can grab where the focus's are through the actionRect property. The actionRect is a rectangle calculated by Camera2D that is centered on every focus in its list, and is large enough to contain every focus in the list. So if we set Camera2D's position to the center of the action rect, the position of the camera is changed to be at the center of the action.

```
void LateUpdate()
{
	Camera2D.position2D = Camera2D.actionRect.center;
}
```

Why not use transform.position you may ask? Well the position2D property of Camera2D automatically clamps any position it is set to not look outside of the camera limits. So by using Camera2D.position2D we are keeping the camera from looking outside of our game.

This is what the final code looks like for a simple action following behavior.

```
using UnityEngine;

[ RequireComponent( typeof( Camera2D )) ]
public class CustomBehavior : MonoBehaviour
{
	Camera2D m_Camera2D;

	void Start ()
	{
		m_Camera2D = GetComponent<Camera2D>();
	}

	void LateUpdate()
	{
		Camera2D.position2D = Camera2D.actionRect.center;
	}
}
```

Of course for any big game you may find yourself wanting to smooth out the camera's movement. I suggest taking a look at SmoothFollow as it is a behavior I provided that does that already.

Make a Camera Focus

To the top of your script add the following inclusion.

```
using Camera2D.Focus;
```

It should be found with the other inclusions of your script so it will look at follows.

```
using UnityEngine;
using System.Collections;
using Camera2D.Focus;
```

Now all you need to do to make a custom focus is to inherit from Focus2D, and implement the following function.

```
public override Vector2 GetFocusPoint ()
{
	return Vector2.zero;
}
```

This is how Camera2D gets the position of the focus. The code combined will look something like this.

```
using UnityEngine;
using System.Collections;
using Camera2D.Focus;

public class CustomFocus : Focus2D
{
        public override Vector2 GetFocusPoint ()
        {
                return Vector2.zero;
        }
}
```

You should now be able to drag your focus into the Camera2D's focus list through the inspector. However it only ever returns Vector2.zero, how about we have it follow the transform instead? Focus2D already has a property for its transform's position converted to a Vector2, so we'll use that.

```
public override Vector2 GetFocusPoint ()
{
        return position2D;
}
```

And done. The final code looks like this.

```
using UnityEngine;
using System.Collections;
using Camera2D.Focus;

public class CustomFocus : Focus2D
{
        public override Vector2 GetFocusPoint ()
        {
                return position2D;
        }
}
```

That is the basics, however more complex objects in your scene may need to be followed. For moving objects I suggest looking at SimpleMoving as it leads the transform towards where the transform is moving, for mouse movement either MouseCursor or WorldCursor will give you a good idea of how to proceed, and for rail systems I suggest looking at the abstract class Rail and inheriting from it (you will still need to implement the GetFocusPoint () function).

Changelog

1.0

January 1st, 2017

- Initial Document