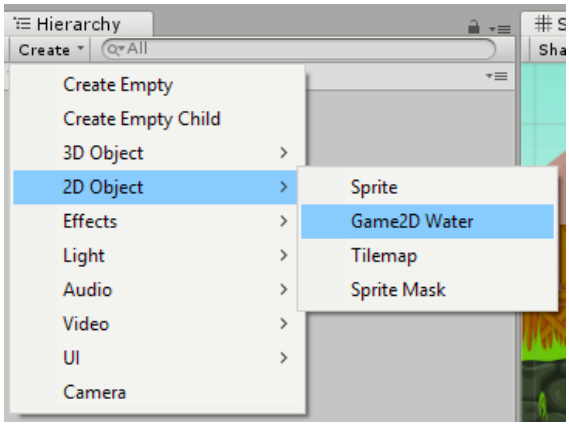
Game2D Water Kit

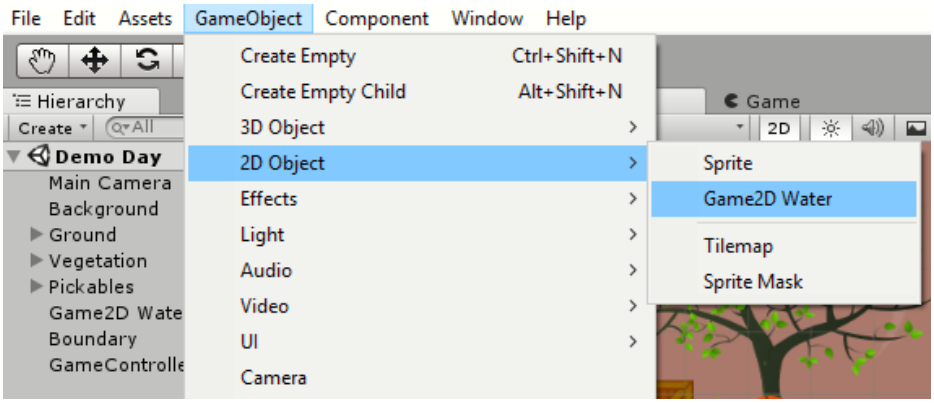
Getting Started:

To create Game2D Water GameObject:

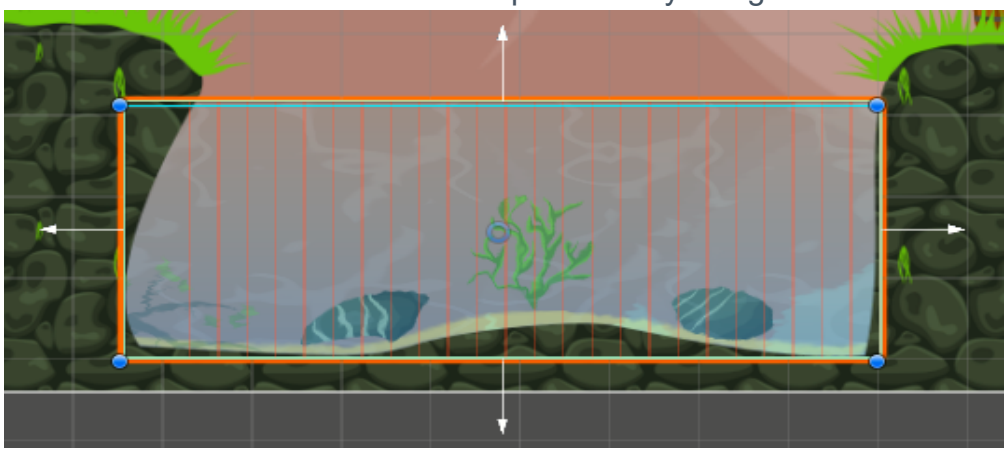
- From the hierarchy window: **Create → 2D Object → Game2D Water**



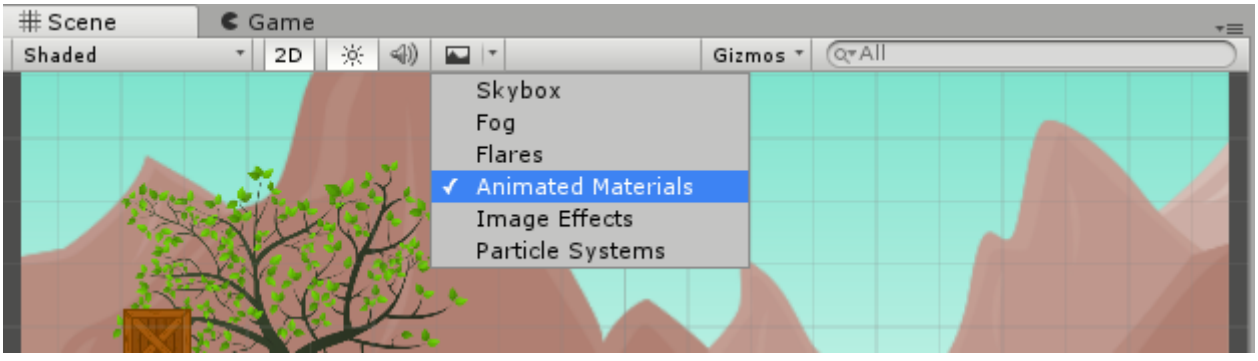
- From the top menu: **GameObject → 2D Object → Game2D Water**



Initially the water has no refraction or reflection enabled. Those can be activated in the material editor. Changing the water size can be done either in the inspector or by using the handles in the scene view.

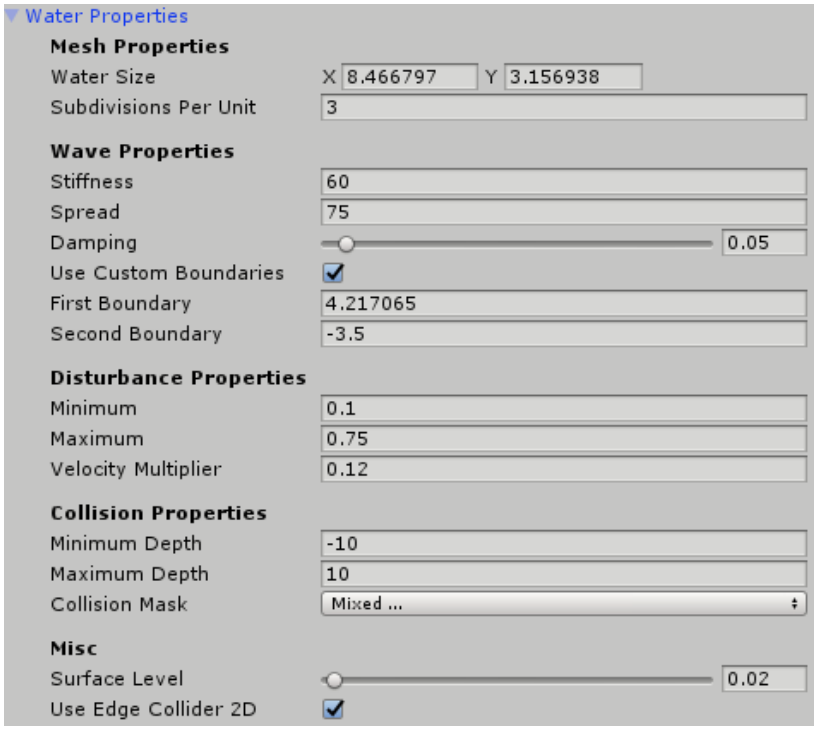


Make sure that “Animated Materials” is checked in order to visualize the water distortion effects in the scene view.



Game2D Water Script:

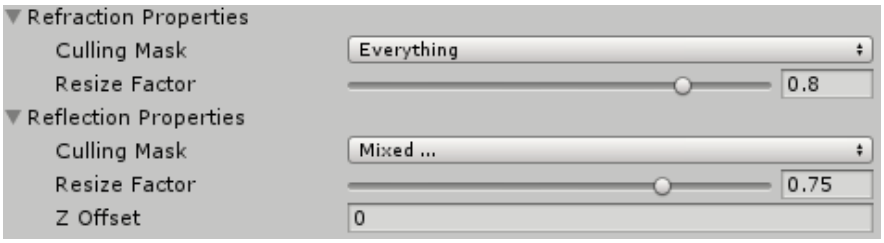
★ Water Properties



- **Mesh Properties:**
  - **Water Size :** Sets the water size. X represents the width and Y represents the height.
  - **Subdivisions Per Unit :** Sets the number of water's surface vertices within one unit.
- **Wave Properties:**
  - **Stiffness:** Controls the frequency of wave vibration. A low value will make waves oscillate slowly, while a high value will make waves oscillate quickly.
  - **Spread:** Controls how fast the waves spread.
  - **Damping:** Controls how fast the waves decay. A low value will make waves oscillate for a long time, while a high value will make waves oscillate for a short time.
  - **Use Custom Boundaries:** Enable/Disable using custom wave boundaries. When waves reach a boundary, they bounce back.
    - **First Boundary:** The location of the first boundary.
    - **Second Boundary:** The location of the second boundary.
- **Disturbance Properties:**
  - **Minimum:** The minimum displacement of water's surface when a GameObject falls into water.
  - **Maximum:** The maximum displacement of water's surface when a GameObject falls into water.
  - **Velocity Multiplier:** When a rigidbody falls into water, the amount of water's surface displacement is determined by multiplying the rigidbody velocity by this factor.
- **Collision Properties:**
  - **Collision Mask:** Only GameObjects on these layers will disturb the water's surface (produce waves) when they fall into water.
  - **Minimum Depth:** Only GameObjects with Z coordinate (depth) greater than or equal to this value will disturb the water's surface when they fall into water.
  - **Maximum Depth:** Only GameObjects with Z coordinate (depth) less than or equal to this value will disturb the water's surface when they fall into water.

- **Misc:**
  - **Surface Level:** Sets the surface location of the buoyancy fluid. When a GameObject is above this line, no buoyancy forces are applied. When a GameObject is intersecting or completely below this line, buoyancy forces are applied.
  - **Use Edge Collider 2D:** Adds/Removes an EdgeCollider2D component. The points of the edge collider are automatically updated whenever the water size changes.

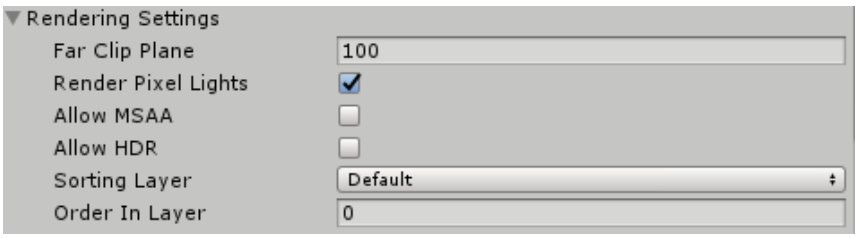
★ Refraction & Reflection Properties:



- **Culling Mask:** Only GameObjects on these layers will be rendered.
- **Resize Factor:** Specifies how much the RenderTexture used to render refraction and/or reflection is resized. Decreasing this value lowers the RenderTexture resolution and thus improves performance at the expense of visual quality.
- **Z Offset:** (Reflection Only) Controls where to start rendering reflection relative to the water GameObject position.

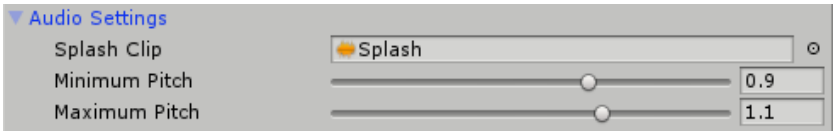
Note: *When rendering refraction and/or reflection, only the visible part of the water is rendered for the sake of better performance.*

★ Rendering Settings:



- **Far Clip Plane:** Sets the furthest point relative to the water that will be drawn when rendering refraction and/or reflection.
- **Render Pixel Lights:** Controls whether the rendered objects will be affected by pixel lights. Disabling this parameter could increase performance at the expense of visual fidelity.
- **Allow MSAA:** Allow multisample antialiasing rendering.
- **Allow HDR:** Allow high dynamic range rendering.
- **Sorting Layer:** The name of the water mesh renderer sorting layer.
- **Order In Layer:** The water mesh renderer order within a sorting layer.

★ Audio Settings:



- **Splash Clip:** The AudioClip asset to play when a GameObject falls into water.

- **Minimum Pitch:** Sets the minimum frequency of the splash clip.
- **Maximum Pitch:** Sets the maximum frequency of the splash clip.

Note: *The AudioSource pitch (playback speed) is linearly interpolated between the minimum pitch and the maximum pitch. When a GameObject falls into water, the higher its velocity, the lower the pitch value is.*

★ Important notes:

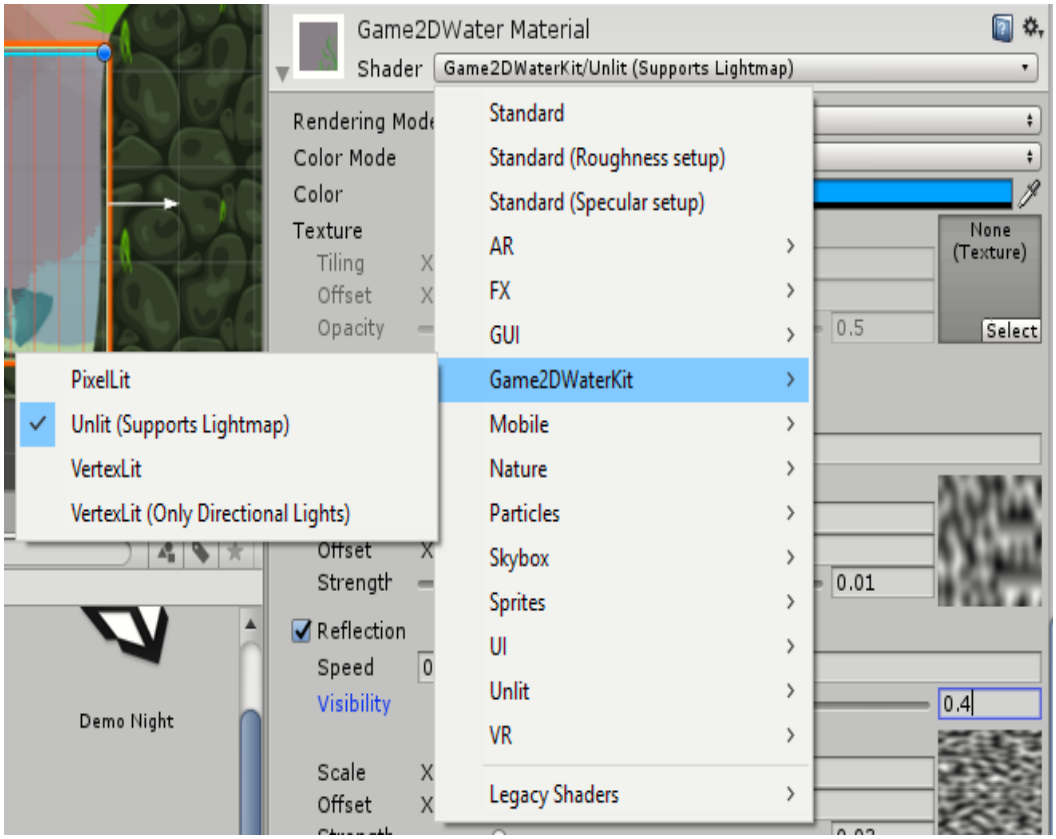
- ❏ Refraction and reflection only work with orthographic cameras.
- ❏ Skyboxes are not rendered when rendering refraction and/or reflection.
- ❏ Lens Flares are not visible when rendering refraction and/or reflection.
- ❏ Refraction and reflection may not work when the water GameObject is marked as Batching Static.

Game2D Water Material Editor:

★ An overview:

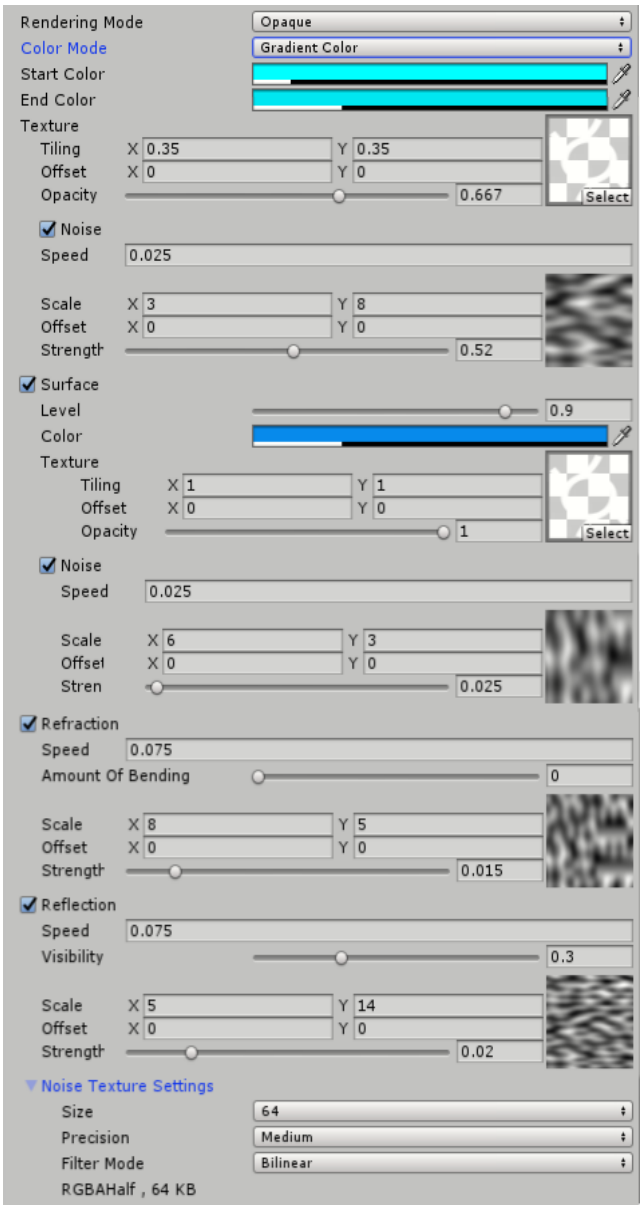
This asset comes with four optimized “Uber” shaders:

- Unlit (Supports Lightmaps)
- VertexLit (Only Directional Lights)
- VertexLit
- PixelLit



- The Unlit shader is the fastest, no realtime lighting calculations are done.
- The VertexLit shaders are cheaper than the PixelLit shader.
- When using VertexLit shaders, all lights are rendered in a single pass and calculated at water mesh vertices only.
- When using PixelLit shader, light is calculated at each pixel that is drawn and each light is rendered in its own pass. Thus, the water GameObject may have to be drawn more than once.
- The PixelLit shader comes at performance cost and it’s not recommended to use it on mobile platforms.
- These shaders come with various features that are enabled or disabled by simply using or not using the texture slots and parameters in the material editor, so the unused shader functions are discarded at runtime automatically with no extra computing overhead.

★ Material Editor:



- **Rendering Mode:** Sets the rendering mode to either Opaque or Transparent. Refraction and reflection are not available when the rendering mode is set to Transparent.
- **Color Mode:** Controls whether the water is tinted with a Solid Color or a Gradient Color.
    - **Color:** Sets the water color when Color Mode is set to Solid Color.
    - **Start Color:** Sets the first color used to generate the gradient color when Color Mode is set to Gradient Color.
    - **End Color:** Sets the second color used to generate the gradient color when Color Mode is set to Gradient Color.
- **Texture:** The texture image applied over the water mesh.
- **Surface:** Enables/Disables rendering the water's surface line.
    - **Level:** Sets where to start rendering the water's surface line.
    - **Color:** Sets the color of the water's surface line.
    - **Texture:** The texture image applied over the water's surface line.
- **Refraction:** Enables/Disables water refraction.
    - **Speed**: Sets the refraction noise texture scroll speed.
    - **Amount Of Bending:** Controls how much the portion of the GameObject above the water is shifted relative to the image viewed under the water.
- **Reflection:** Enables/Disables water reflection.
    - **Speed:** Sets the reflection noise texture scroll speed.

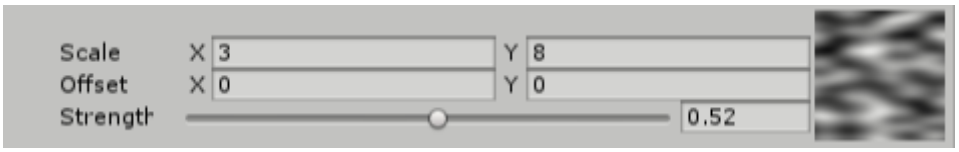
- **Visibility:** Controls the visibility of the reflection when both reflection and refraction are enabled.

Distortion effect could be applied to:

- The water texture.
- The water's surface line texture.
- Refraction render texture.
- Reflection render texture.

The distortion effect is achieved using a Perlin noise texture.

Perlin noise texture generation parameters:

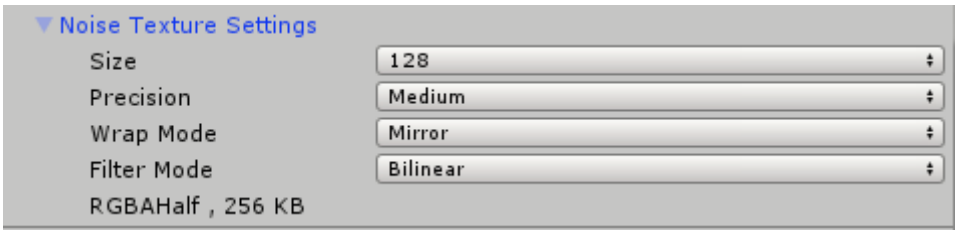


- **Scale:** Sets the noise scale in x and y axes.
- **Offset:** Sets the noise offset in x and y axes.
- **Strength:** Sets the strength of the distortion.

To save memory bandwidth, only one Perlin noise texture is generated:

- The alpha color channel stores the distortion information of the water texture.
- The green color channel stores the distortion information of the water's surface line texture.
- The blue color channel stores the distortion information of the reflection render texture.
- The red color channel stores the distortion information of the refraction render texture.

Noise texture settings:



- **Size:** Sets the noise texture size.
- **Precision:** Sets the noise texture data type precision.
  - **Low:** 8 bits per channel (Fixed).
  - **Medium:** 16 bits per channel (Half).
  - **High:** 32 bits per channel (Float).
- **Wrap Mode:** Sets the noise texture wrap mode to Repeat or Mirror.
- **Filter Mode:** Sets the noise texture filter mode to Point, Bilinear or Trilinear.

Note: *The Wrap Mode field is only available in Unity version 2017.x or newer. Unity versions prior to 2017.x don't support setting texture wrap mode to Mirror, and thus the noise texture wrap mode is always set to Repeat.*