-ZERIN LABS-

# Shader Pack

**Cartoon water**

Welcome!
...and thanks for buying this outstanding shader pack :)

On this small tutorial you will find all the necessary details to configure the shaders and prepare your meshes to create amazing video-game graphics.

For any doubt feel free to contact us at: zerinlabs@gmail.com

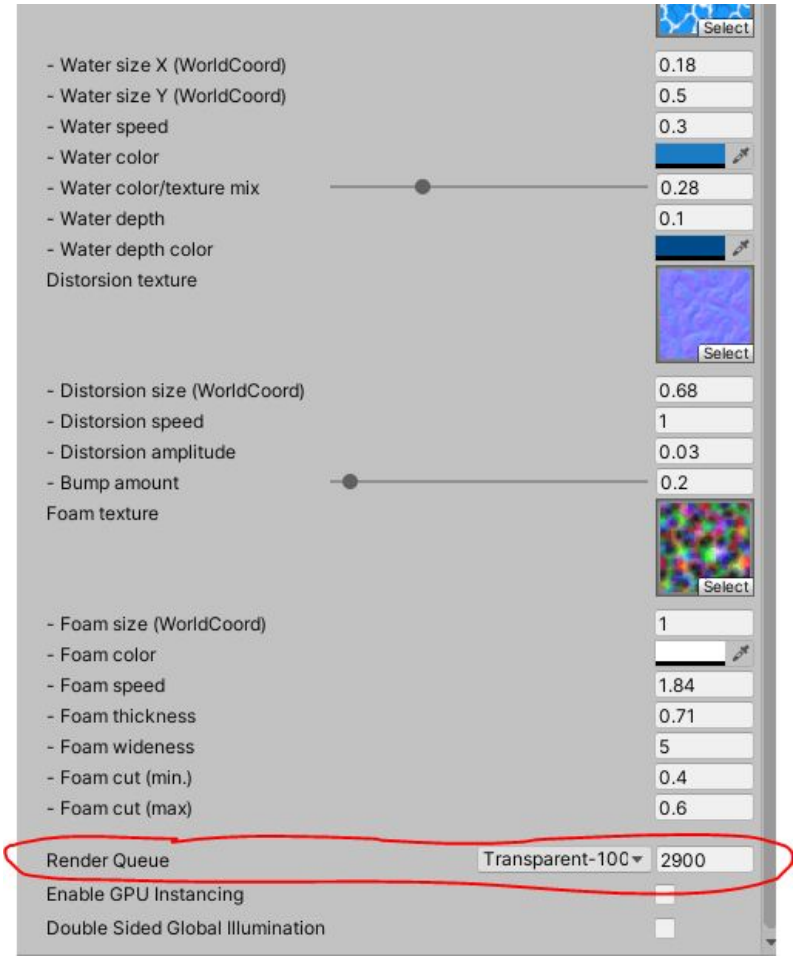
# HOW TO USE THE SHADERS INSIDE UNITY

**IMPORTANT:** In order to display your shader properly it is important for you to **install URP or use the LWRP/HDRP** as your main render pipeline.

**Optionally** you may install **Shader Graph** in case you would like to tweak or modify the shaders (but it is actually not necessary)

Due to the special nature of the shaders it may be required that you change the “rendering Queue” of your material in order to display some of the water shaders properly and avoid possible artifacts.

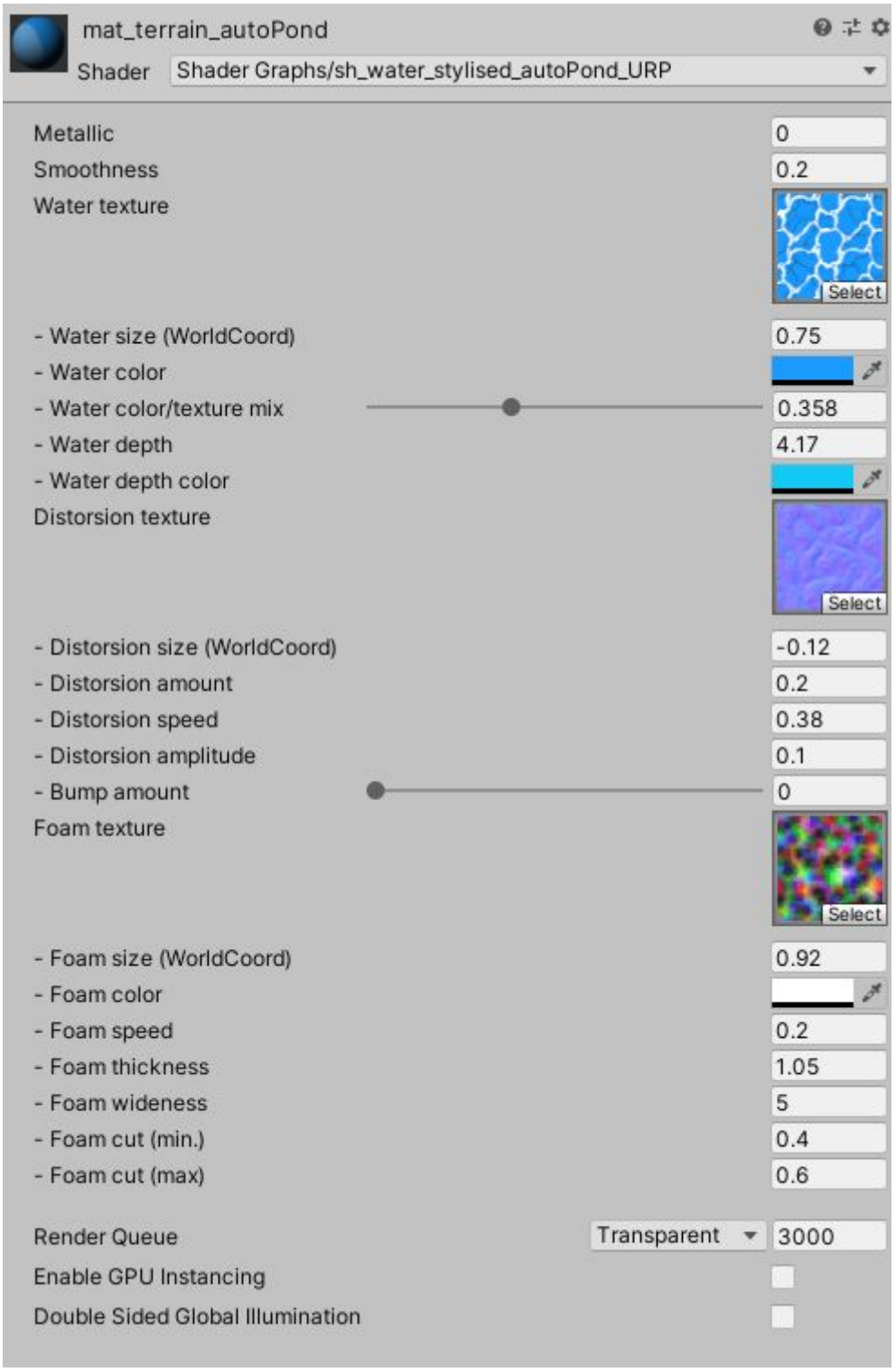
You could do this quite easily directly on the unity inspector scrolling down through the properties of your material once the shader is set:




Setting up the render queue to 2900 should make the deal

# SHADERS & PROPERTIES

## Shader : sh_water_stylised_autoPond_URP



## Use

This shader is meant to render without effort cartoony lakes, ponds and other surfaces of “still” water.
No UV’s are required due it uses triplanar projection (world mapping projection) for the texture coordinates.

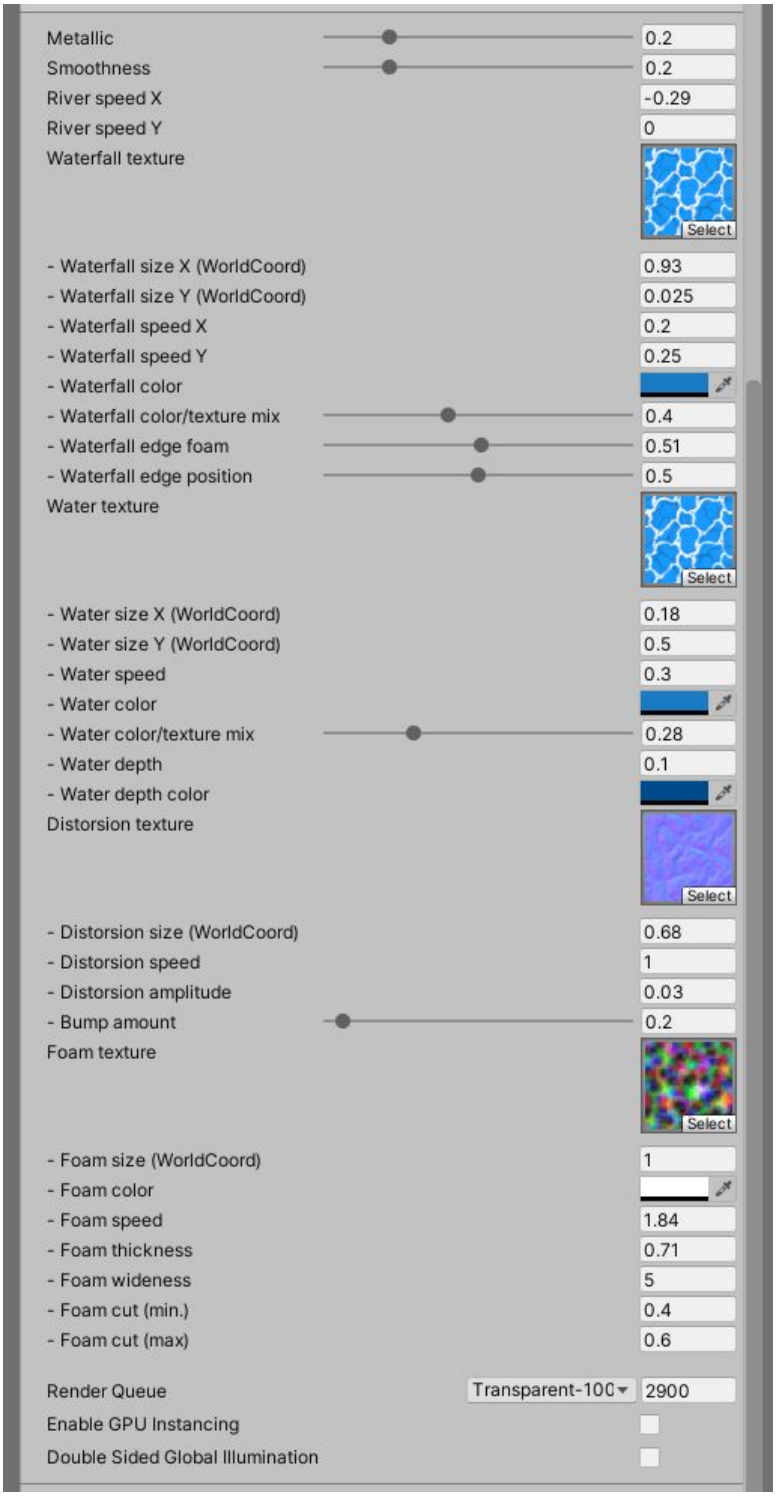
## Properties

- **Metallic + Smoothness**
    - The shader is fully PBR compatible so it could be light regularly using the “smoothness” and “metallic” factors the same way as the regular standar PBR shaders
- **Water texture**
    - Albedo for the water texture color
- **Water size (world coord)**
    - Scale factor for the albedo triplanar mapping
- **Water color**
    - color overlay that will mixed with the albedo texture color
- **Water color/texture mix**
    - Mix factor that determines the amount of “texture” or “color” the albedo composite will result
- **Water depth**
    - Factor that emulates how “visible” is the ground through “shallow waters”. Using this factor you can emulate how the water changes its colour on the shallow/deeper areas
- **Water depth color**
    - Alternative color for “shallow water” areas
- **Distortion texture**
    - Normalmap used to distort the albedo texture and emulate water surface waves/distortion (it is using on the PBR lighting computation too)
- **Distortion size (world coord)**
    - Scale factor for the distortion normal-map mapping
- **Distortion amount**
    - Amount of “distortion” caused by the previous normal-map
- **Distortion speed**
    - Determines the speed (waving effect) of the distortion component
- **Distortion amplitude**
    - Determines the amplitude/distance of the distortion waving effect.
- **Bump amount**
    - Amount of “bumpiness” caused by distortion texture normal-map used on the PBR calculation
- **Foam texture**
    - Special texture used as a “transition-mask” on the edges (where the water meets the ground) of the water surface
- **Foam color**

    - Determines the color of the “foam edge” of the water surface
- **Foam speed**
    - Determines the speed (waving effect) of the foam mask
- **Foam thickness/wideness + Foam cut (min/max)**
    - This components control how big and sharp is the “foam edge” on the water surfaces

# SHADERS & PROPERTIES

## Shader : sh_water_stylised_autoRiver_URP




### Use

This shader is meant to map easyly rivers and waterfalls from the same water stream. Again, no UV's are required due it uses triplanar projection (world mapping projection) for the texture coordinates.

## Properties

**IMPORTANT:** Some of the properties may be skipped due they're used exactly the same way as on the "sh_water_stylised_autoPond_URP" shader

- **Metallic + Smoothness**
  - Same as previous shader
- River speed X/Y
  - Controls the speed and direction of the river stream
- **Waterfall texture**
  - Sets the waterfall albedo texture
- **Waterfall size X/Y (World Coord)**
  - Scale factor for the albedo triplanar mapping
- **Waterfall speed X/Y**
  - Controls the speed and direction of the waterfall
- **Waterfall color**
  - Color overlay that will mixed with the albedo texture color
- **Waterfall color/texture mix**
  - Mix factor that determines the amount of "texture" or "color" the albedo composite will result
- **Waterfall edge foam**
  - Controls the thickness of the transition (aka foam) between the "waterfall" and the "river" shader part
- **Waterfall position**
  - Controls position/angle of the transition between the "waterfall" and the "river" shader part
- **Water texture**
  - Same as previous shader
- **Water size X/Y (World Coord)**
  - Same as previous shader (but with a non regular scaling factor)
- **Water speed**
- **Water color**
- **Water color/texture mix**
- **Water depth**
- **Water depth color**
- **Distortion texture**
- **Distortion size (world coord)**
- **Distortion amount**
- **Distortion speed**
- **Distortion amplitude**
- **Bump amount**
- **Foam texture**
- **Foam color**
- **Foam speed**
- **Foam thickness/wideness + Foam cut (min/max)**
  - All same as previous shader

# CONTACT

- Mail
  - zerinlabs@gmail.com
- Twitter
  - @zerinlabs
- Site
  - https://zerinlabs.blogspot.com/
- Youtube channel
  - https://www.youtube.com/channel/UC-u0QyXSJUS60hAfc-UnF-A