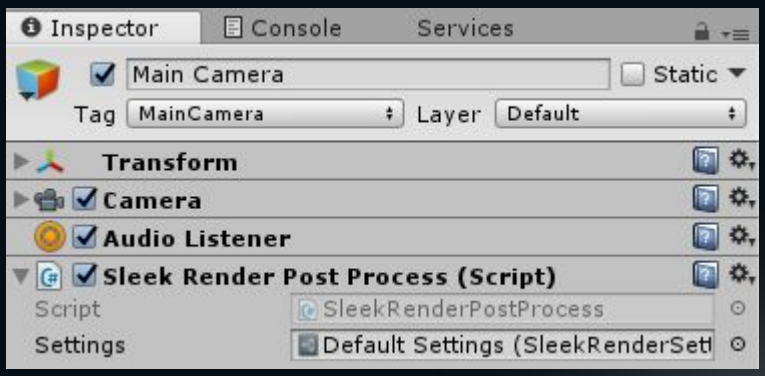
Sleek Render

v 0.8

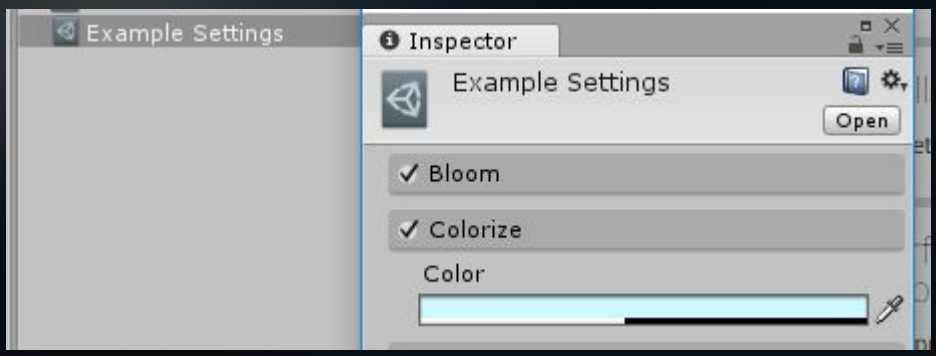
Thanks for purchasing Sleek Render!
Please read the setup and usage part carefully.

1. Setup

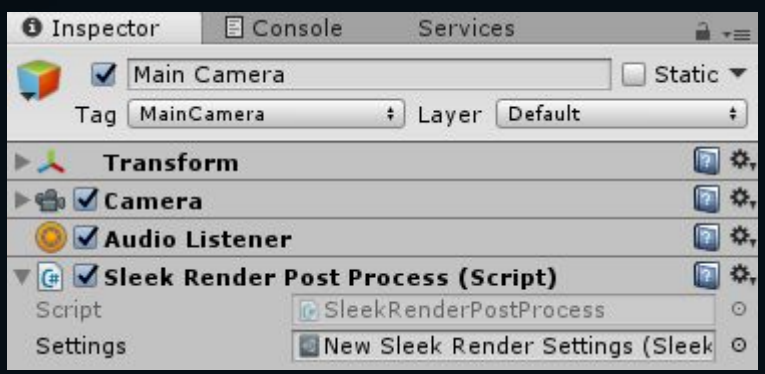
1. Add "*SleekRenderPostProcess*" component to your scene camera.



2. Create a Sleek Render Settings asset in the Project View. To do it, right click anywhere inside the Project View in Unity, and choose Create -> Sleek Render Settings. This will create a settings asset where you can tweak the post process rendering parameters.



3. Link this newly created asset to a "*SleekRenderPostProcess*" component "*Settings*" field.



4. Done. Now let's tweak some parameters.

2.1 Bloom

Bloom settings include:

1. Bloom Threshold
2. Bloom Intensity
3. Bloom tint
4. Bloom texture size
5. Bloom passes
6. Brightpass Luma calculation

Bloom Threshold sets the brightpass parameters. It shows how bright or dim should be the light to be picked up by the bloom effect. Bright pass is not binary, that is - it's not just black-or-white after the step, it gradually decreases the bloom brightness based on the pixel luminance

Bloom Intensity sets the additive strength of the bloom. The higher the value, the more bright the bloom will appear.

Bloom tint changes the initial bloom color. It's useful when you need to stylize the overall "glow" feel of the picture. Notice that this color can be further changed by the Color overlay effect.

Bloom texture size changes the size of the blurring texture. Smaller size means bigger bloom, but more artifacts on small objects (flickering). Bigger size - smaller bloom, but less artifacts or small objects. It's advised to set it to 128x128 for a general scene, and it can be reduced to 32x128 to achieve anamorphic bloom on big areas of bright light (sky). Performance differences of different texture sizes is **negligible**, that is, there would be little to no performance gains when switching to 32 from 128.
There's also an option to toggle "Preserve aspect ratio" which will ensure that bloom shape will remain constantly uniform on all screen aspect ratios.

Bloom passes count sets the number of times Sleek Render blurs the scene using differently sized textures. More bloom passes mean better quality, but worse performance. Generally using 5 passes should be fast enough for most cases. But, if needed, total number can be brought down to 3 if extra performance "punch" is required. In current version, only 5 and 3 passes are available as they cover most of the general bloom setups.

Brightpass Luma calculation parameter sets the way of calculating the pixel brightness from three color channels. It's set to Uniform by default, so all colors are picked by bloom equally. It works for most cases, especially if the scene has a lot of vivid and bright colors that need to be picked by bloom. But it's not a very physically correct way of calculating brightness. If you need to bloom physical objects like Sky, it's advised to select "SRGB" from the dropdown menu. You can also use a custom luma vector if required.

2.2 Color overlay

Sleek render has a versatile color grading tool. It's not based on LUTs, so color overlay will appear uniform over the whole image.
The Colorize parameter sets the Color Overlay (RGB) and it's Intensity (Alpha). The more alpha the color has, the more apparent the colorize effect becomes. Zero alpha means no color grading will be visible.

Common usage patterns of Color Overlay are:

1. **Desaturate**. Set color to WHITE and set Alpha to 1 (fully desaturated grayscale image) or somewhere in between (less alpha - more saturation).
2. **Colorize**. Set color to desired value and tweak Alpha to make it more or less apparent.
3. **Fade In / Fade Out.** Color Overlay effect can be used to greatly reduce fillrate wasted on Fade Out - Fade In effects. Rather than making a fullscreen black UI Texture and tweening it's alpha, you can set color to BLACK and tween Alpha from 0 to 1 to Fade Out or from 1 to 0 to Fade In. It will create the same fade effect without wasting any additional fillrate.

2.3 Vignette

Vignette is an effect in photography or filmmaking where corners and edges of the image are slightly darkened or painted with some color. It can help to focus the viewer on important things in the scene and adds a bit of realism to the picture.

Vignette parameters are:

1. Vignette Begin Radius - distance on which the vignette begins (larger radius - vignette effect starts closer to edges/corners)
2. Vignette Expand Radius - vignette gradient shape and size. (larger radius - softer vignette gradient)
3. Vignette Color - pretty self-explanatory. Black color with full alpha will darken the vignetted parts of the image completely.
4.

2.4 Brightness / Contrast

Brightness / Contrast effect allows tweaking overall brightness (uniform color addition to make everything look brighter) and contrast using a simple common contrast formula.

3. Relative cost calculator

Sleek Render has a relative cost calculator that prints out approximate time values for the current setup. It can be used as an initial frame budget planning tool and depends on one's min-tier hardware, as well as hardware-dependent visual settings used in a game. Time values were acquired using native graphics profilers on NVIDIA Tegra 3 and ARM Mali400. It's important to note that exact timings greatly depend on GPU hardware architecture and it's always advised to double check the timings on a device using native graphics profilers.

4. Script control

To edit post process parameters via scripts, add "*using SleekRender*" line at the top of your script, create a "*SleekRenderSettings*" public field. Then, link the same settings asset that is used to control the effects on your main camera. After that, change parameters by accessing and changing public fields of the settings asset.

5. Support

For any questions related to this package, feel free to write to:
- Unity forum thread https://goo.gl/pzgt3p
- support email support@nadezhdin.org