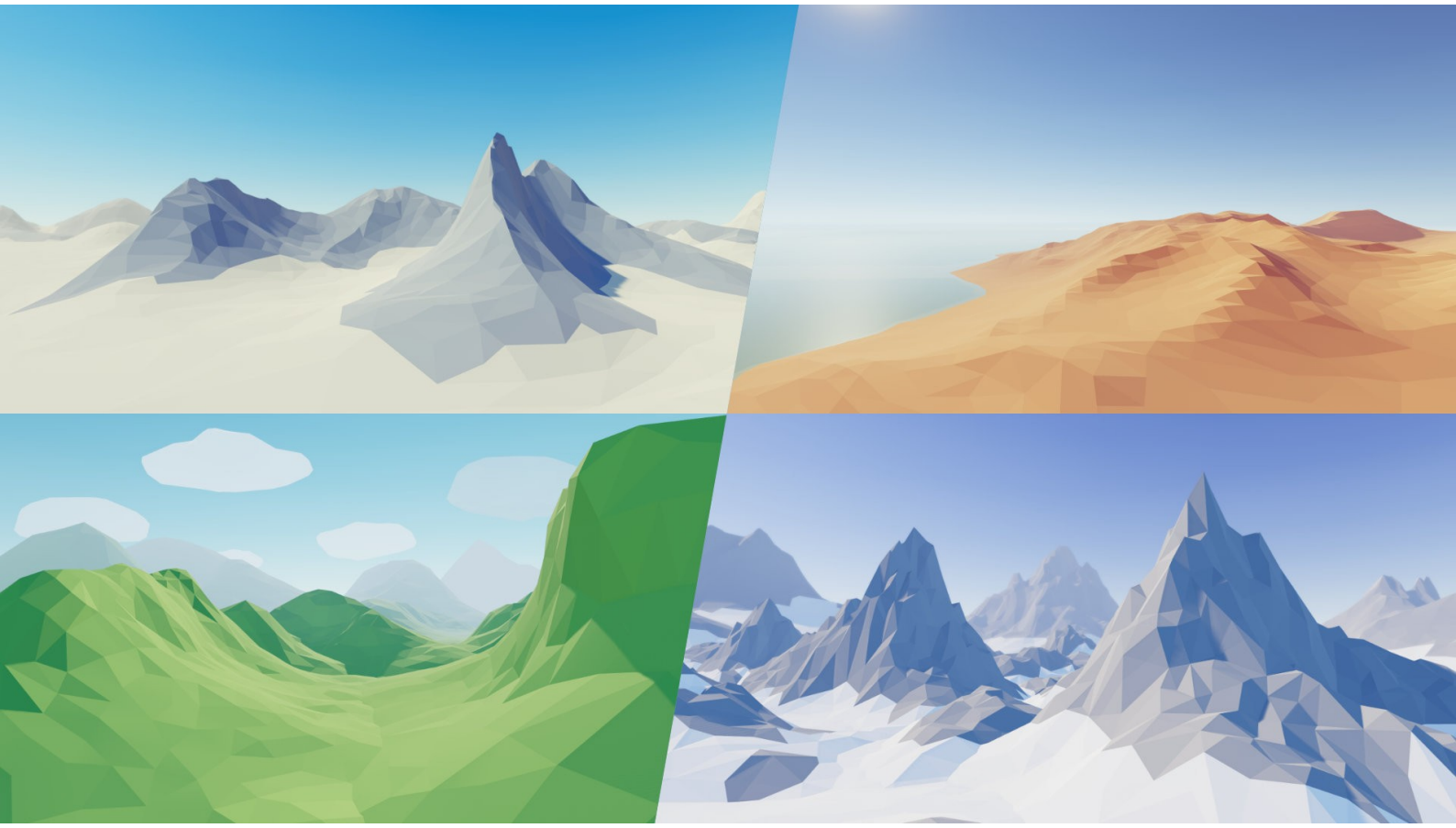
LOW POLY MODULAR TERRAIN PACK



Contacts

E-mail: justinas@lmhpoly.com

Website: https://lmhpoly.com/contact/

Follow me on **Twitter** to see what I'm working on right now:

https://twitter.com/lmhpoly

LMHPOLY

Don’t miss out, and be the first!

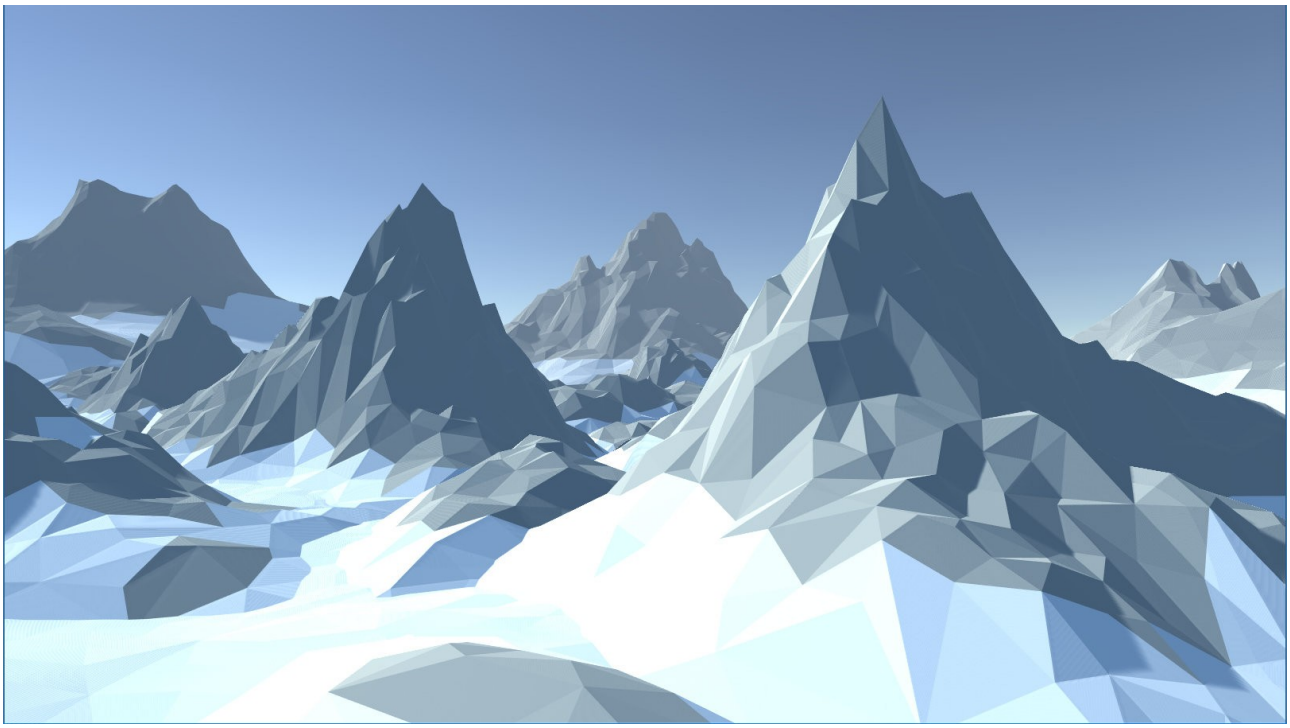
Get notified about the new “Low Poly Modular Terrain Pack” and other asset updates + my new game asset releases straight to your inbox.

Subscribe to LMHPOLY Game Asset Newsletter.

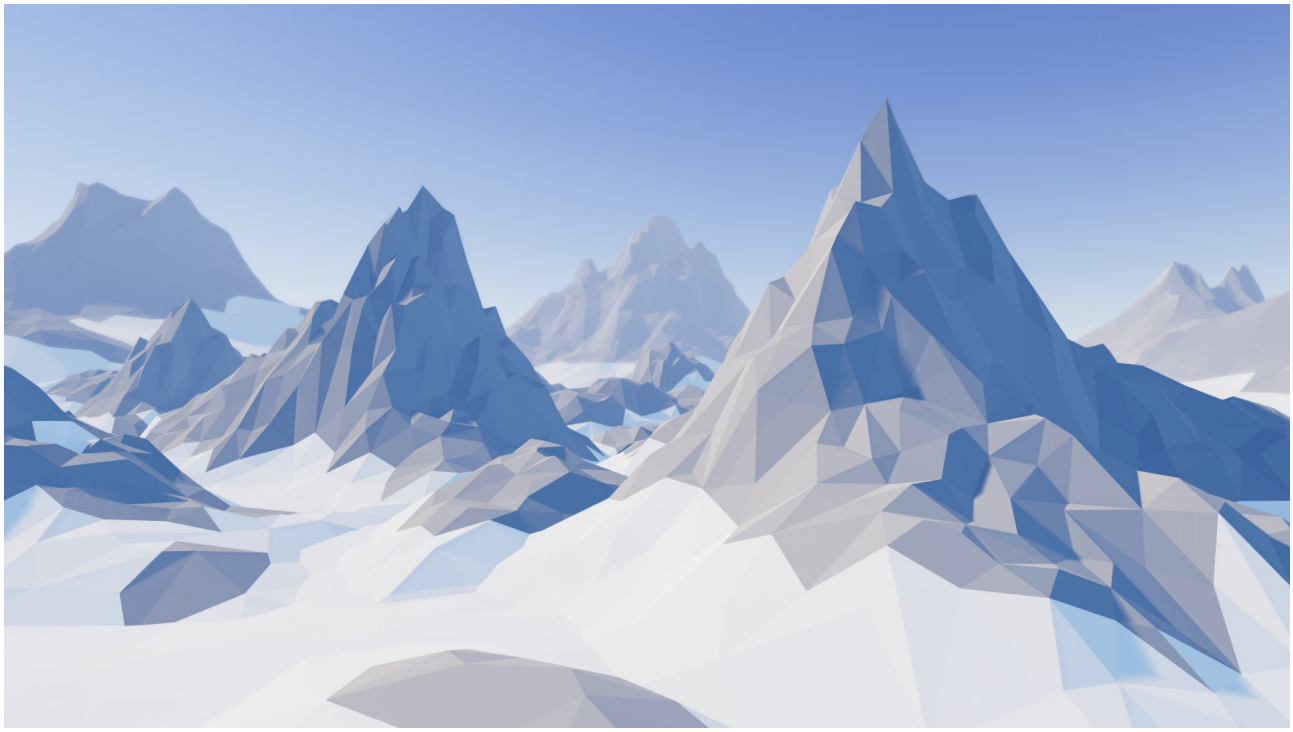
Content

Demo Scenes

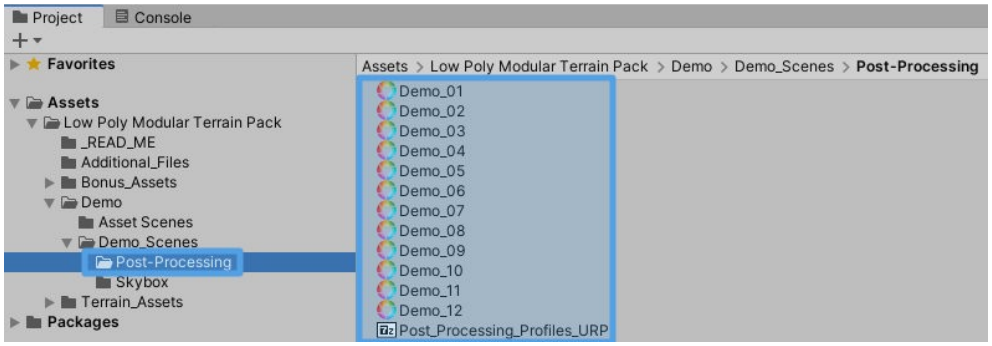
Now, as you have imported the whole "**Low Poly Modular Terrain Pack**" to your Unity project, go to *Low Poly Modular Terrain Pack/Demo/Demo_Scenes* - and open any Demo Scene (here is a **Demo_04** example). By default, the scene should look something like this inside the **Game** view using **Gamma** Color Space and without any image effects applied.



To make it look like this:



you need to use **Post-Processing Profile** on each demo scene.



Follow the steps below to setup **Post-Processing** image effects for Demo Scenes!

Post-Processing in Unity 2019.4 LTS and up – (Built-In Render Pipeline)

Post-Processing in Unity 2019.4 LTS and up – URP (Universal Render Pipeline / Universal 3D)

*You need at least Unity 2019.4 LTS to setup Post-Processing by following my tutorial!

BONUS

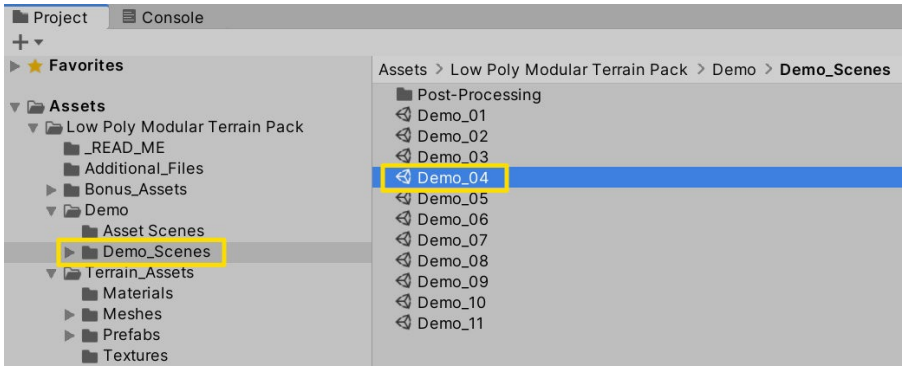
UPDATE! You can watch my video tutorial on the lighting and post-processing workflow I use for my low poly scenes if you want to light your own newly created scene in Unity:

Unity URP Tutorial - Lighting And Post-Processing

Unity 2020 Tutorial - Lighting And Post-Processing Low Poly Scene

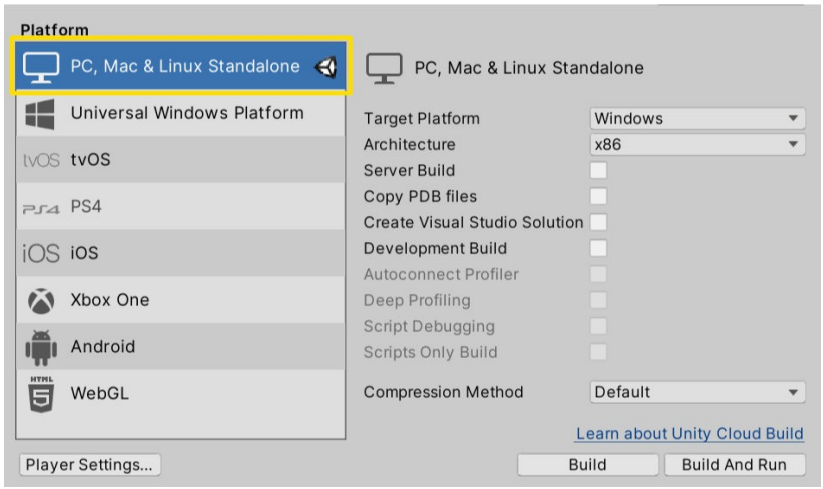
How to Setup Demo Scenes (Post-Processing) in 3D (Built-In Render Pipeline) (For PC)

Before we start, let's open the **Demo_04** scene located at: *Low Poly Modular Terrain Pack/Demo/Demo_Scenes*

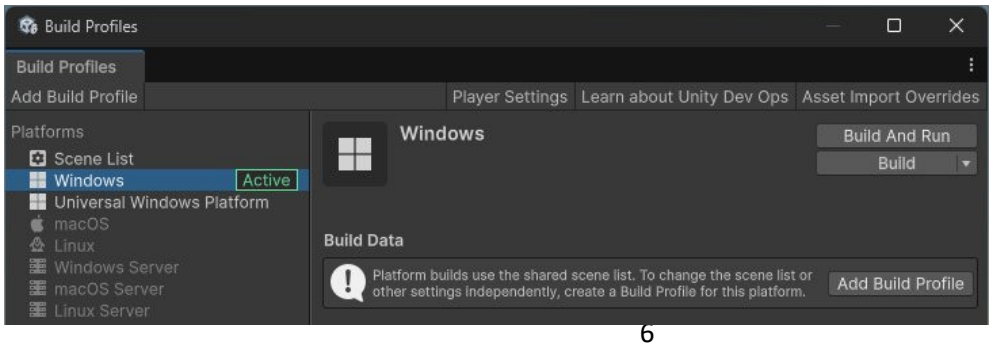


Then go to *File > Build Settings*

Make sure you are using a **PC, Mac & Linux Standalone** build.



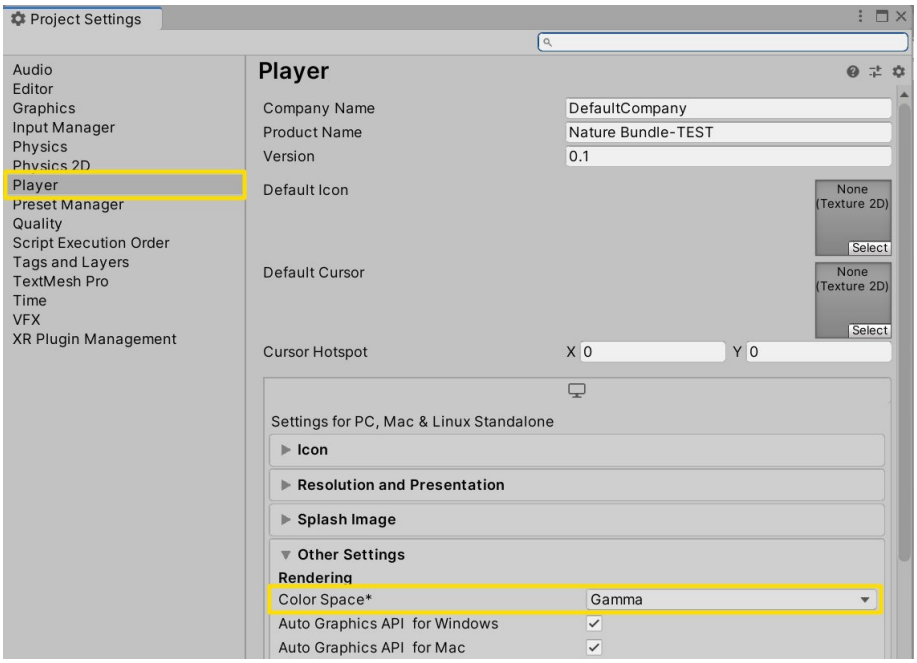
For Unity 6, go to the *File > Build Profiles*



1. Change to the Linear Color Space

Go to the *Edit > Project Settings*

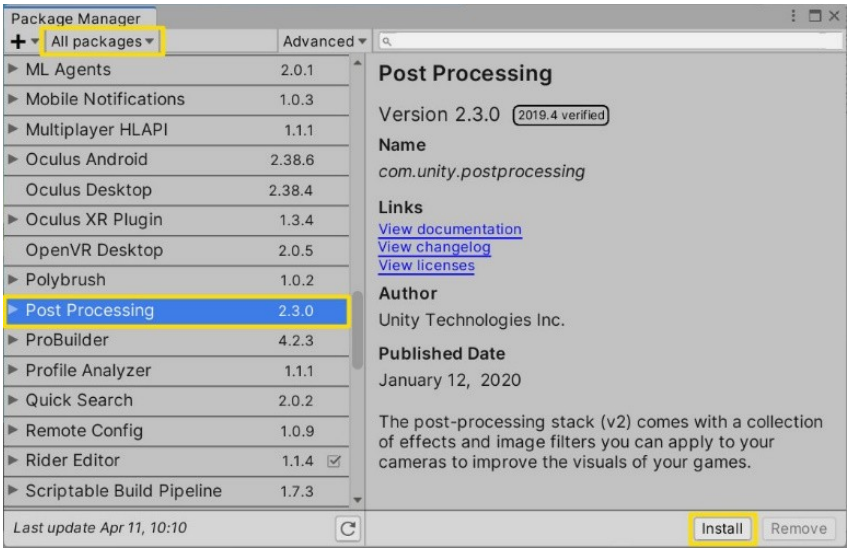
Open the **Player** tab, **Other Settings** section, and set the **Color Space*** to **Linear**.



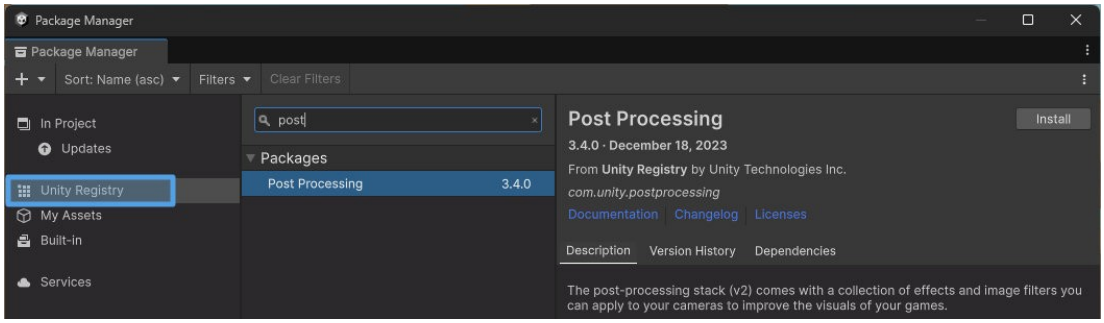
2. Install the Post-Processing

Go to the *Window > Package Manager*

Set view to **All packages**, search for the **Post Processing**, select it, and click **Install**.



For Unity 6 it's located in the Package Manager, Unity Registry tab:

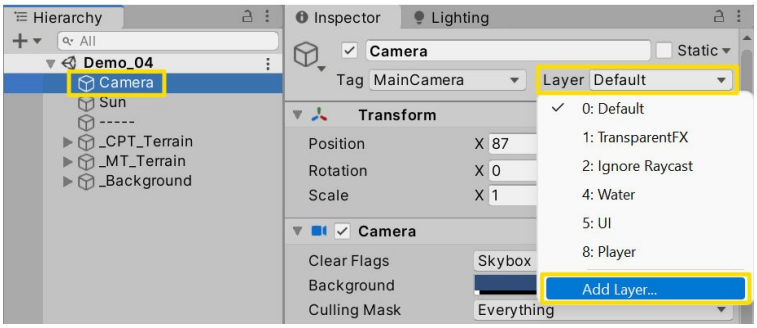


***NOTE:** If you have problems in the later steps setting up the Post-Processing:

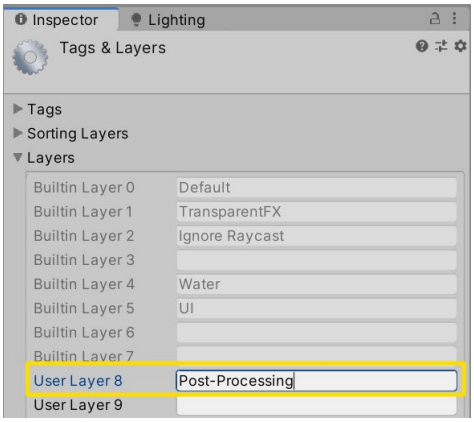
- Restart Unity.
    - If it still doesn't work, go to *Window > Package Manager*, and remove the **Post Processing** package.
    - Restart Unity
    - Install the **Post Processing** package again. Now it should work.

3. Set up the Post-Processing

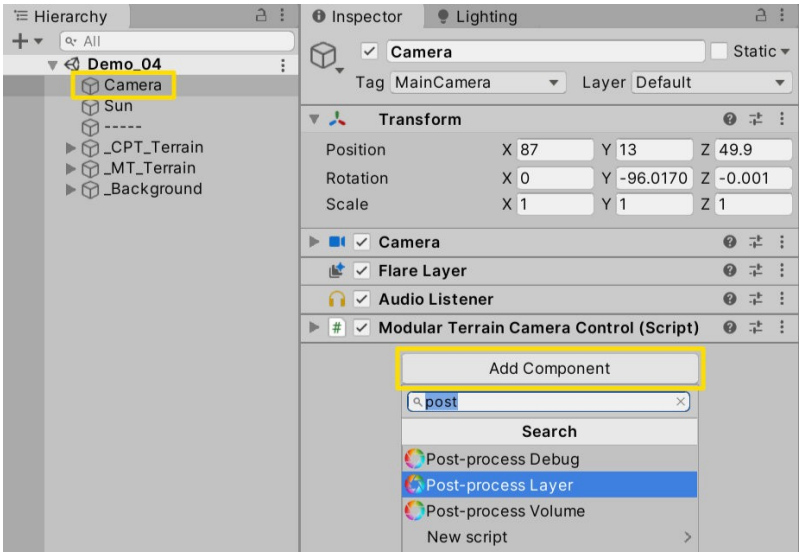
Select the **Camera** in the **Hierarchy**, click on **Layer > Add Layer**



Let's add a new layer to any blank space and call it ***Post-Processing*** *(you can call it however you want)*.

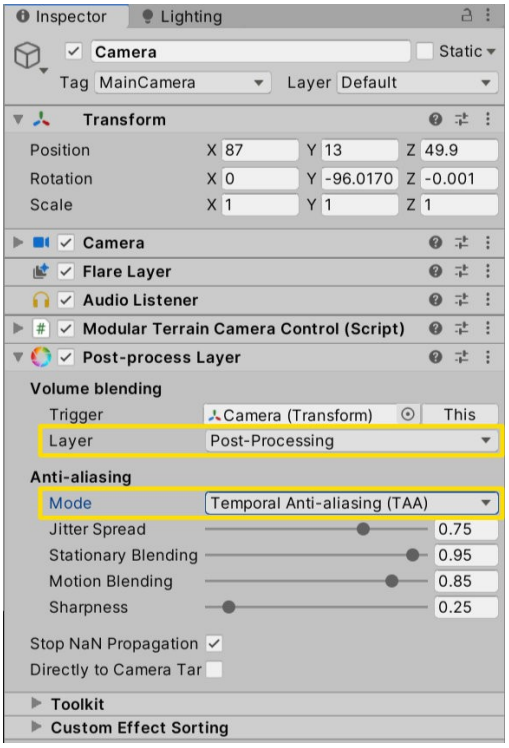


Select the **Camera** again, click on **Add Component**, and type **post** in the search bar. You should see 3 Post-process components. Click on **Post-process Layer** to add it to the **Camera**.

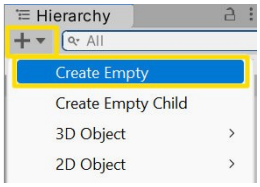


And set the **Layer** to **Post-Processing** *(the Layer we just created).*

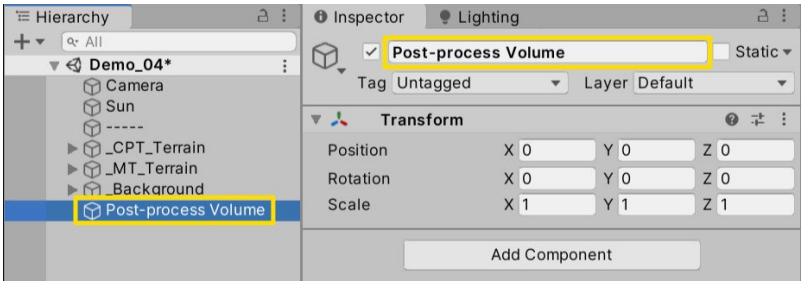
Also, I like to set **Anti-aliasing** to **Temporal Anti-aliasing (TAA)** - to get rid of those jagged edges and some screen tearing when moving the Camera in the game.



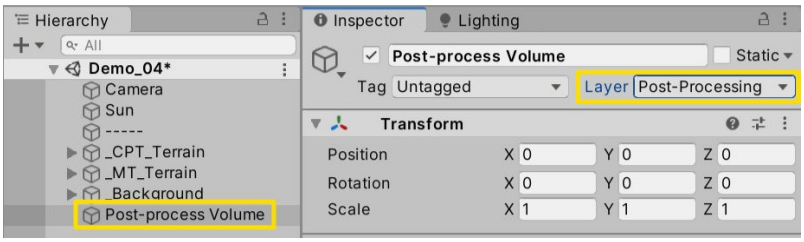
Now, inside the **Hierarchy**, we need to **Create Empty** gameObject



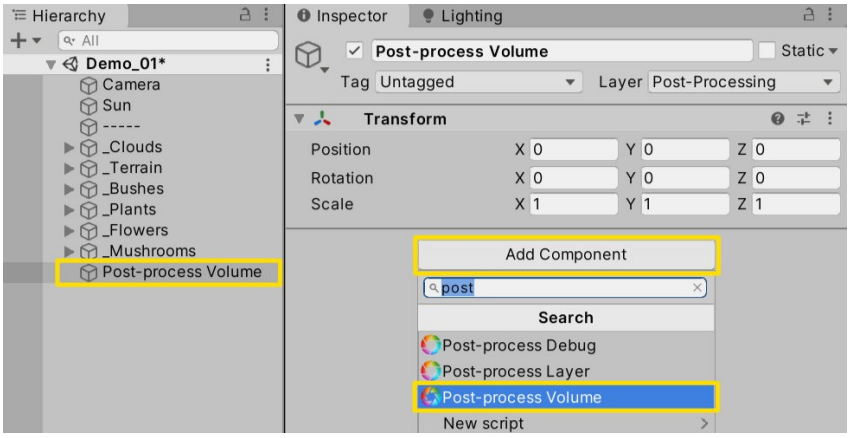
Let's call it **Post-process Volume**



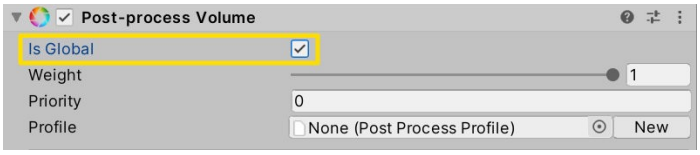
Set the **Layer** to **Post-Processing** *(the Layer we created before).*



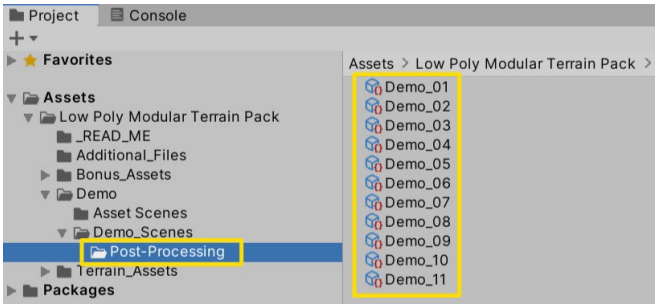
Add Component > Post-process Volume



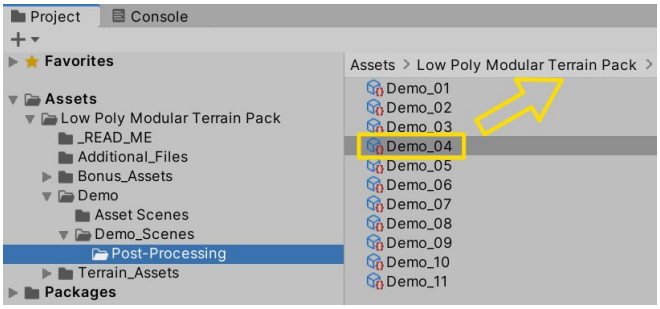
Enable **Is Global**



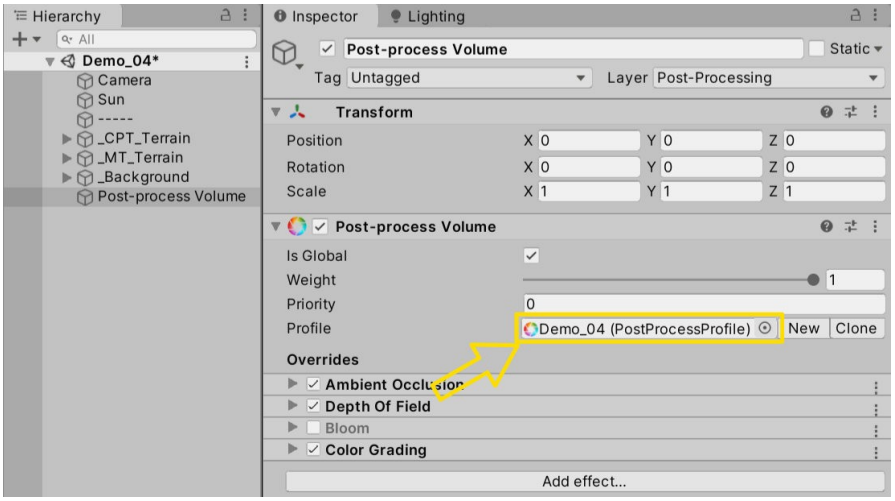
Then go to *Low Poly Modular Terrain Pack/Demo/Demo_Scenes/Post-Processing.* Here you can find my pre-made custom **Post-Processing Profiles**, which we can use for every Demo scene to quickly apply effects like Color Grading, Ambient Occlusion, etc.



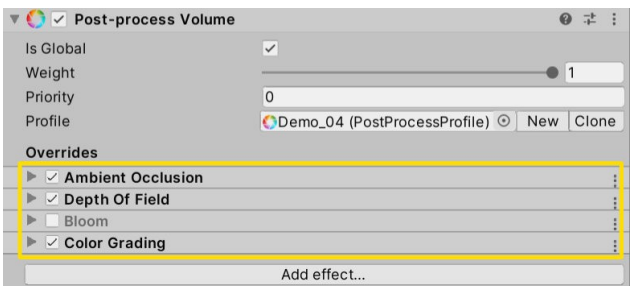
Drag and drop **Demo_04** *(Post-Process Profile)*



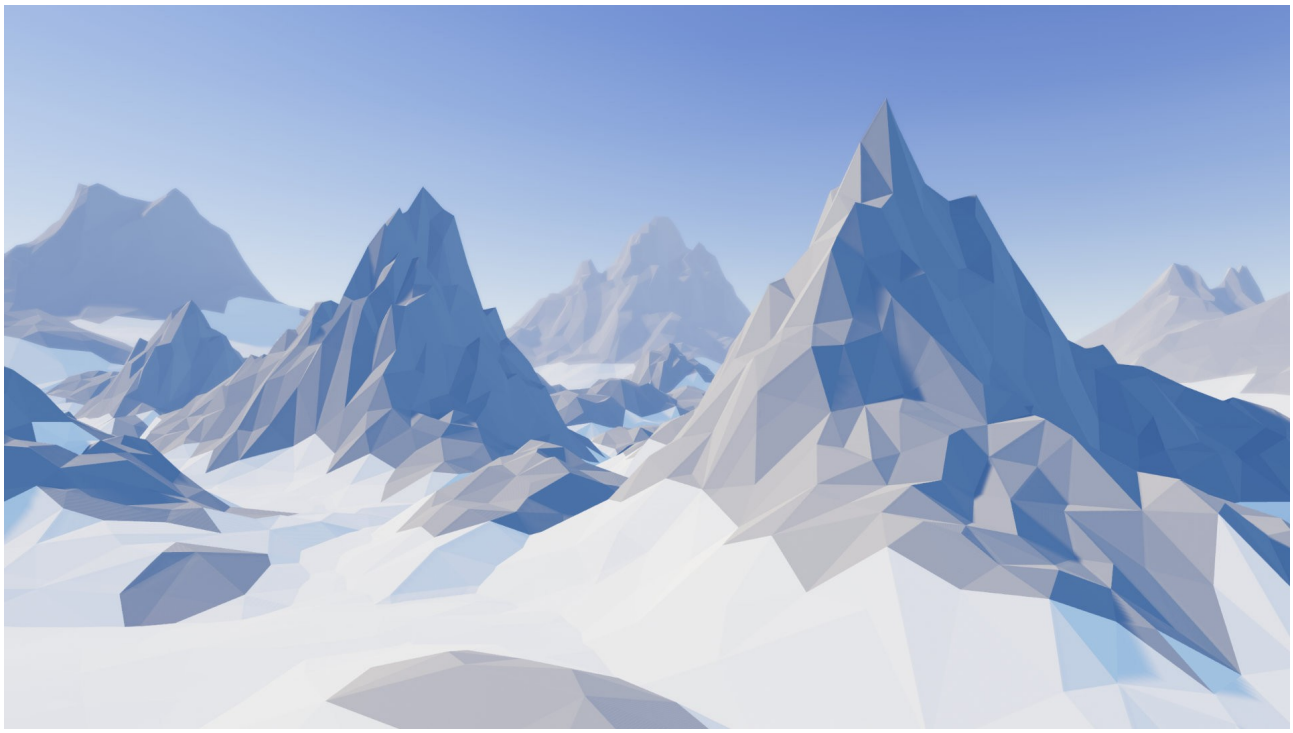
To the **Profile** area in the **Post-process Volume** section



Here you can see what effects this scene is using, which you can easily edit:



After completing these steps, your scene should look like this:

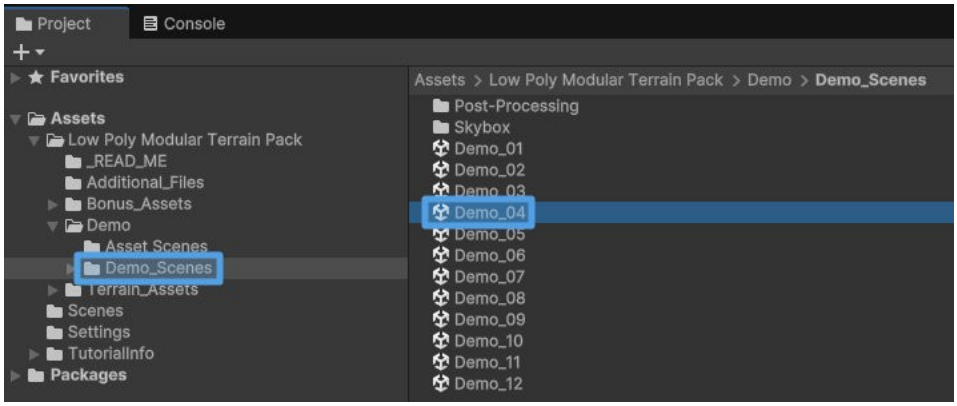


*__For Low-End PC's__ - if you hit play and it lags, try disabling Post-Processing effects one by one on the Post-Processing Profile settings!

To quickly add the Post-Processing effects to any other Demo scene by applying my custom Post-Processing profiles, you need to repeat all the steps from: adding **Post-process Layer** to the **Camera**.

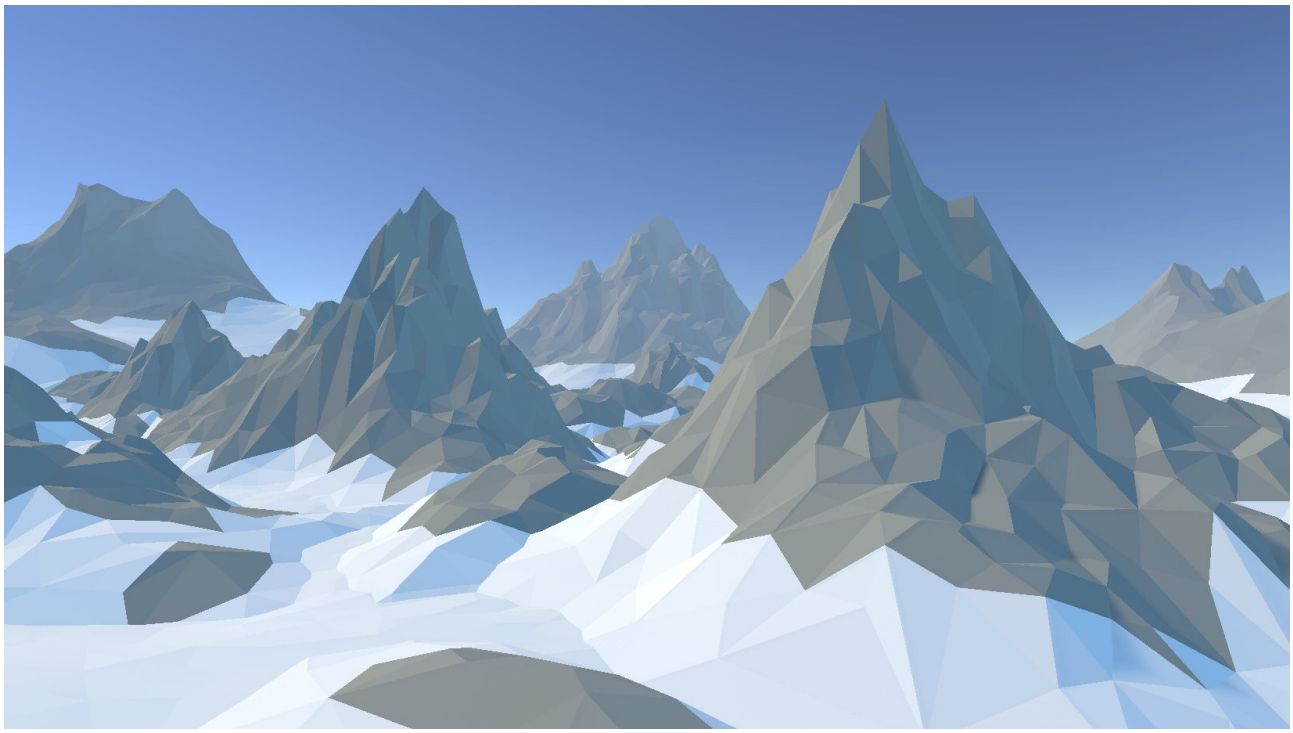
How to Setup Demo Scenes (Post-Processing) in URP / Universal 3D (For PC)

Before we start, let's open the **Demo_04** scene located at: *Low Poly Modular Terrain Pack\Demo\Demo_Scenes.*



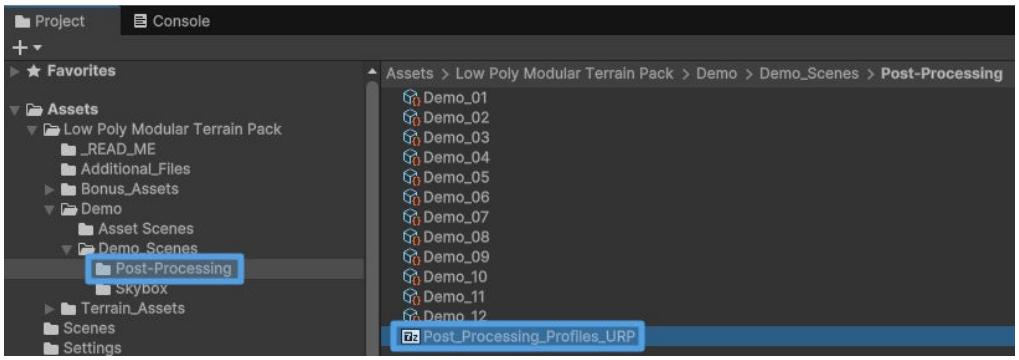
*If your scene has **Pink Materials**, read on how to fix it here.

Your scene "**Demo_04**" should look like this:



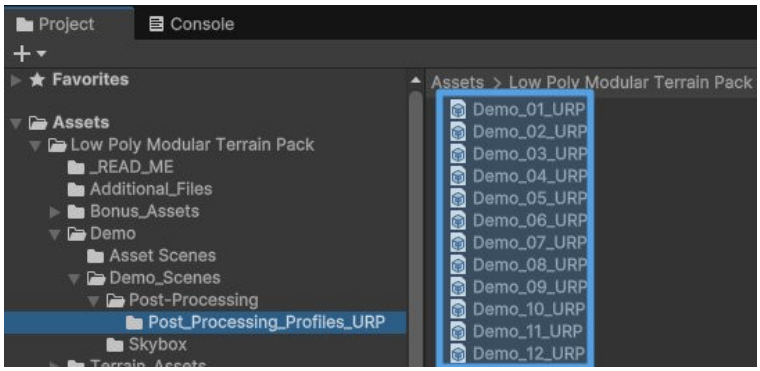
Unity 6 example

Open the **Post-Processing** folder located at: *Low Poly Modular Terrain Pack\Demo\Demo_Scenes\Post-Processing*

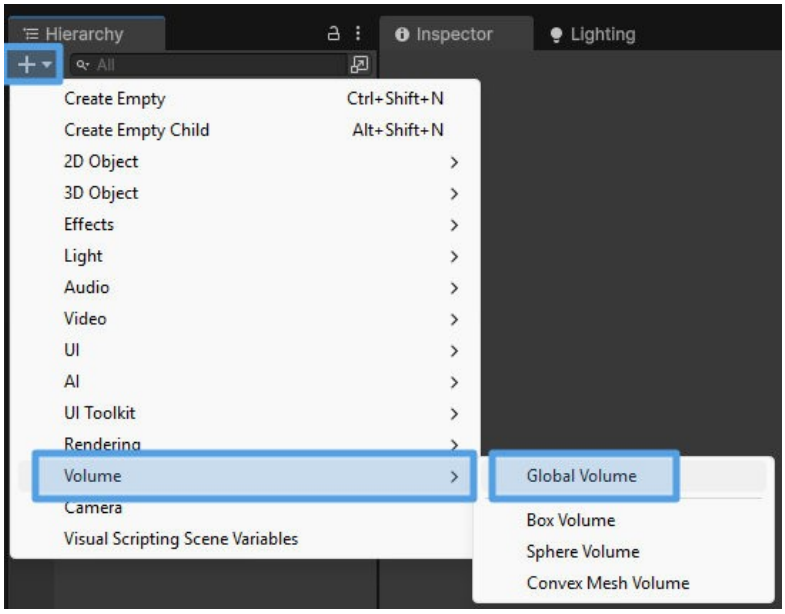


You will find the zip file "**Post_Processing_Profiles_URP**." Open it in the file explorer and extract the files inside to a new folder called *"Post_Processing_Profiles_URP,"* or whatever you call it.

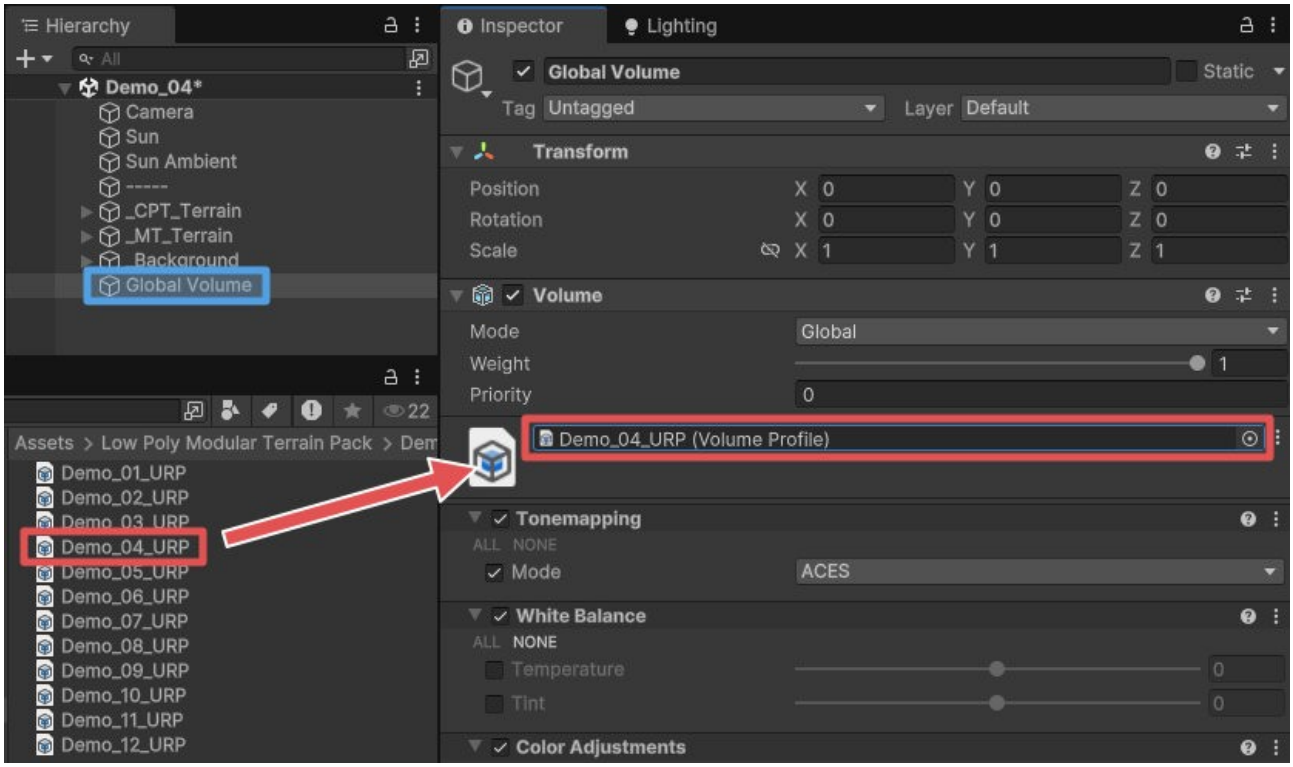
You will see 12 Post-Processing Profile files for 12 demo scenes: *Demo_01_URP, Demo_02_URP, etc.*



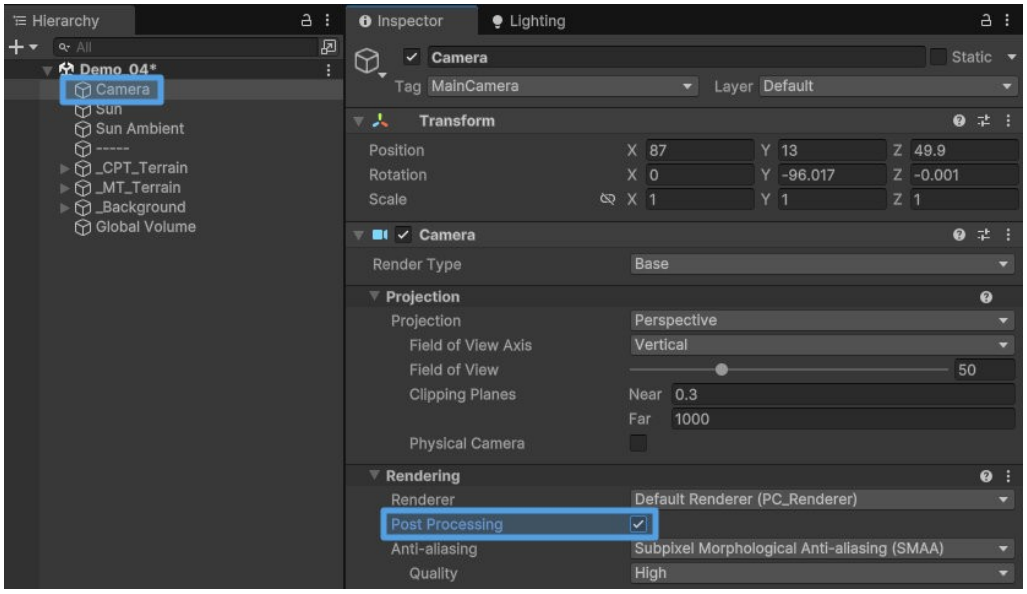
Now, inside the **Hierarchy**, press the '**+**' *> Volume > Global Volume*



*Select the newly created **Global Volume** inside the **Hierarchy**. Drag and drop **Demo_04_URP** Post-Processing Profile to the **(Volume Profile)** location (highlighted in the picture below.)*



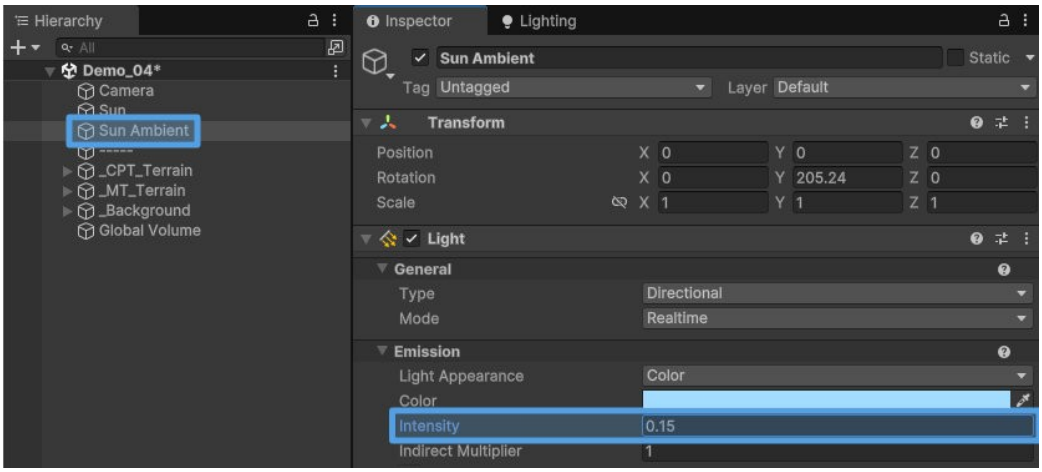
*Now, Post-Processing will work in the **Scene** View only. To make it appear in the **Game** View, select the **Camera** and enable **Post-Processing**.*



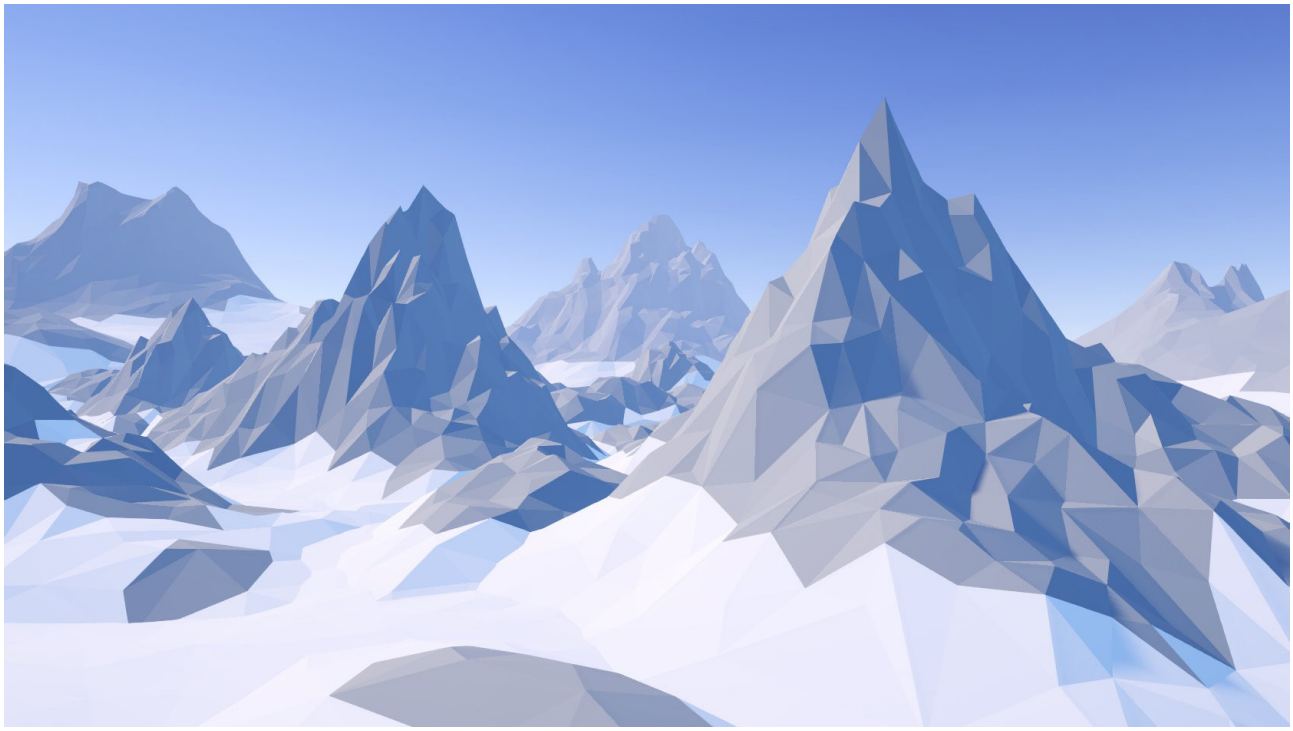
*You can also enable **Anti-aliasing**, which is below the Post Processing option, to make the image look smooth. Can be a big performance hit, depending on the hardware and the mode you choose.

BONUS TIP

In Unity URP / Universal 3D, **'Sun Ambient'** can be too strong compared to when it's used in the Unity Built-In render pipeline. I recommend reducing the **'Sun Ambient' Intensity** to a lower value. For **Demo_04**, I set the **'Sun Ambient' Intensity** to **0.15**, and it looks better, not too bright.



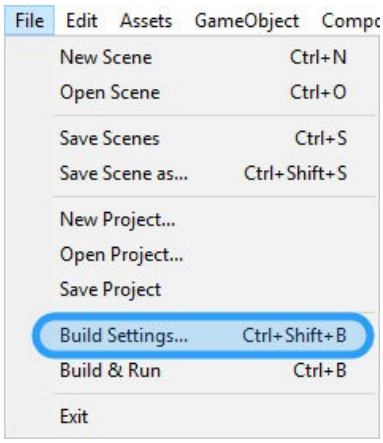
And that's it. The **Demo_04** scene should look like this:



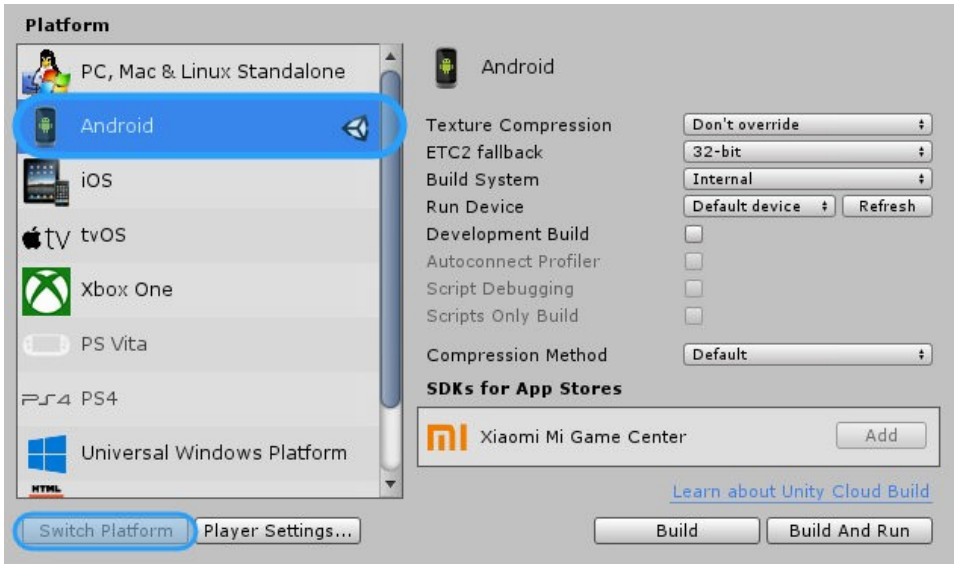
How to Setup Demo Scenes in Unity 2019.4 LTS and up (For Android)

1. Make sure you are using **Android** build!

Go to *File > Build Settings*



Select **Android** and hit the **Switch Platform** button.

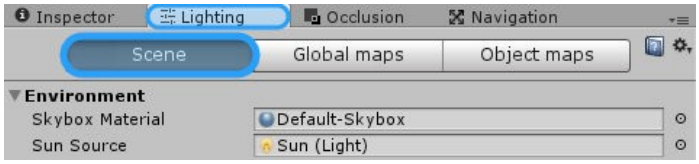


For Unity 6, go to to *File > Build Profiles*

2. **Clean GI Cache** (Optional – **Skip** this if you don't have any light baking errors!)

Before you go to the next step, you need to disable the **Auto** build/bake feature.

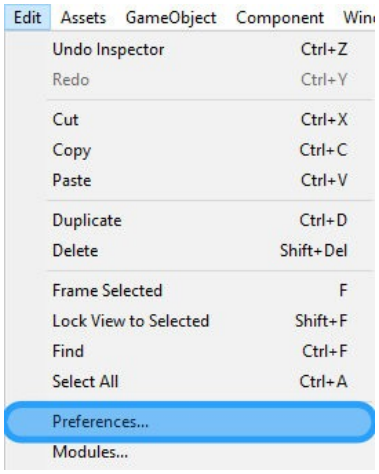
You can find it in **Lighting** and select the **Scene** tab (If you don't see Lighting tab go to *Window > Lighting > Settings).*



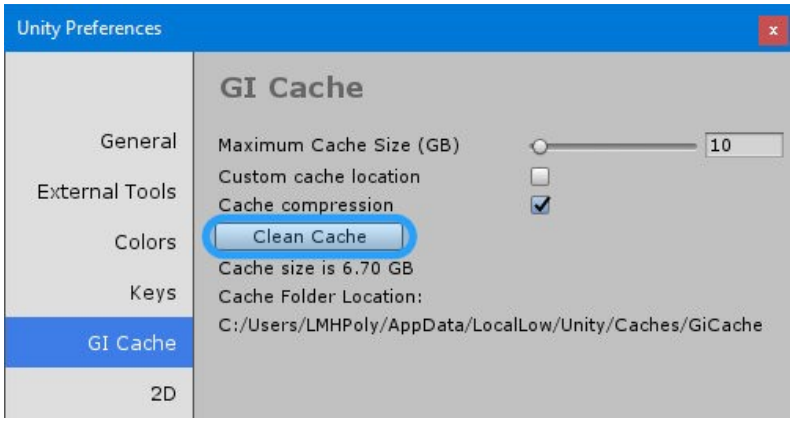
At the bottom you will see this:

uncheck **Auto Generate**.

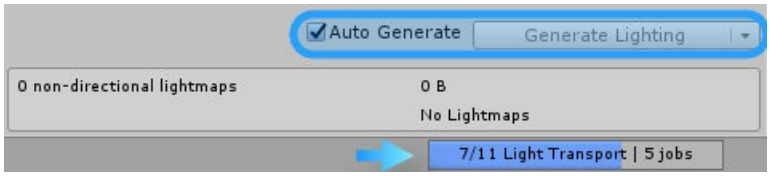
Go to *Edit > Preferences*



Select **Gi Cache** tab and press on the **Clean Cache** button!



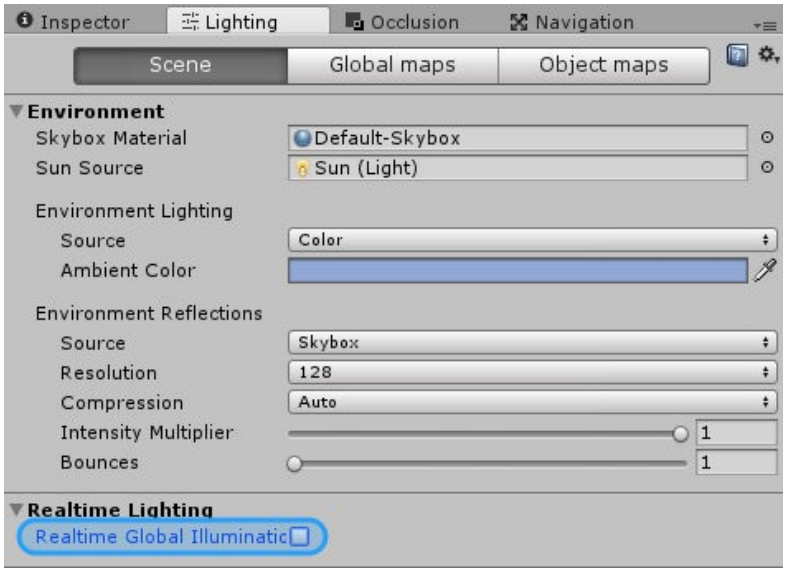
Enable **Auto Generate**/bake feature



and wait until the generation is done (blue loading bar at the right bottom corner).

3. Disable **Realtime Global Illumination** (Optional – for slightly better performance)

You can find it in **Lighting** and select the **Scene** tab (If you don't see Lighting tab go to *Window > Lighting > Settings).*

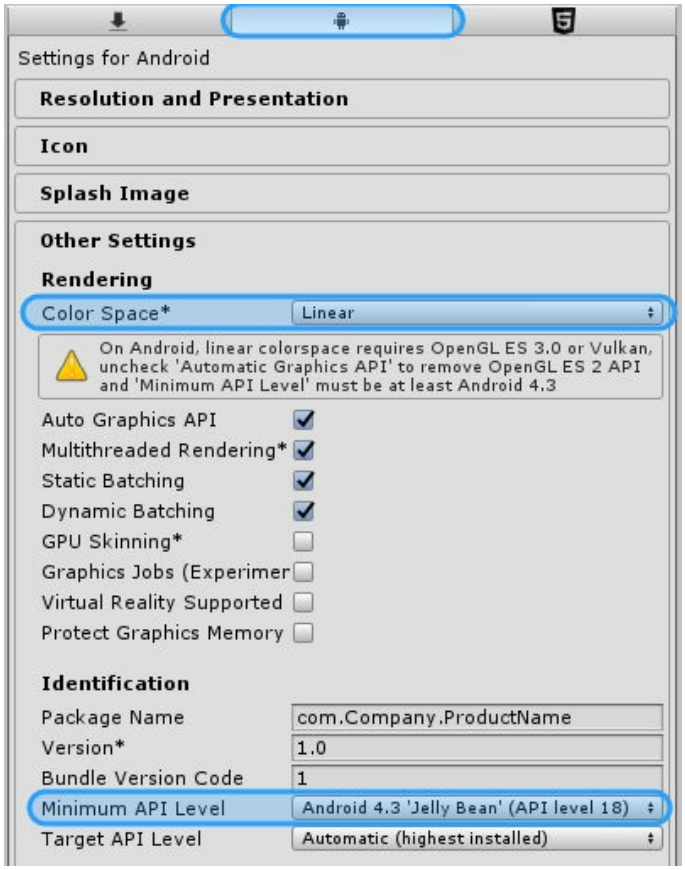


4. Make sure that **Color Space** is set to **Linear** (not all devices support it).

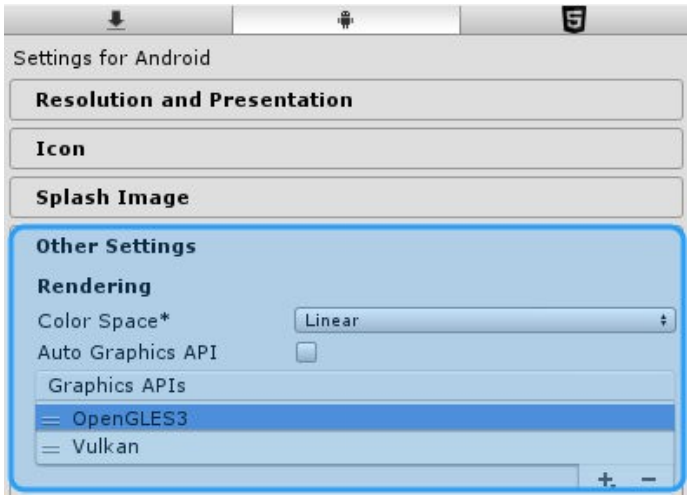
Go to *Edit > Project Settings > Player*

In the **Other Setting** tab, you will find **Color Space***, set it to **Linear**.

To use **Linear** Color Space on Android, you need to set the **Minimum API level** to at least **Android 4.3 (API level 18)** or higher!

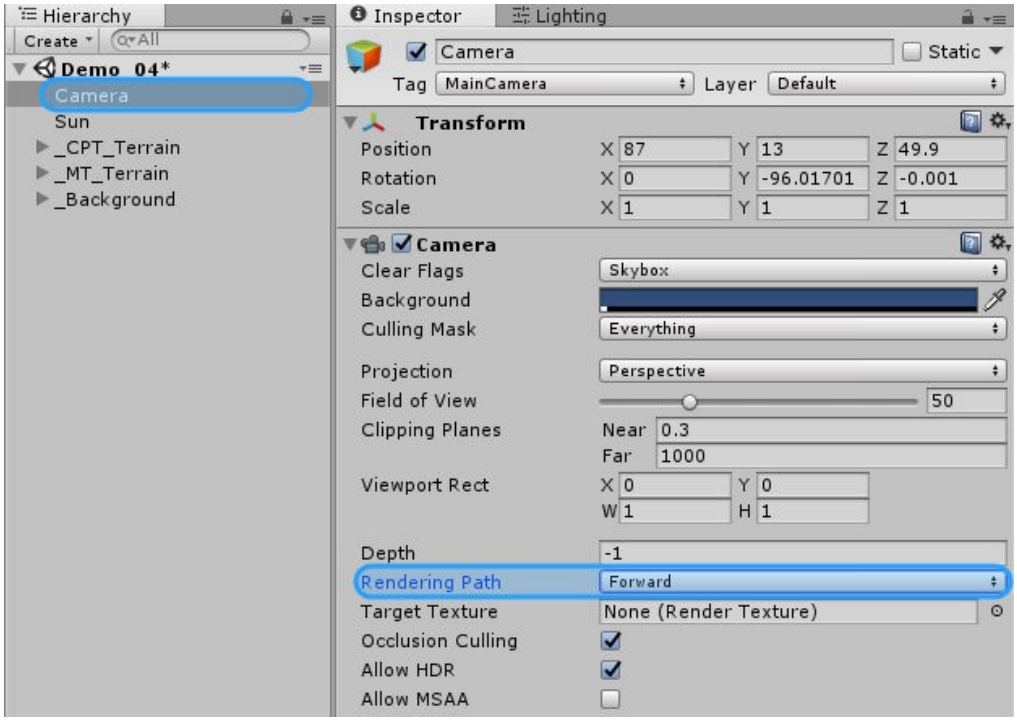


Also, uncheck **Auto Graphics API** and remove all Graphic APIs from the list, leave only **OpenGLES3** and **Vulkan** (if you can't see it, press on **+** to add it). Make sure your Android device supports one of those graphic APIs!



5. Make sure that you are using **Forward Rendering**. (Use Forward Rendering instead of Deferred for better mobile performance).

Select the **Camera** and make sure that **Rendering Path** is set to **Forward.**

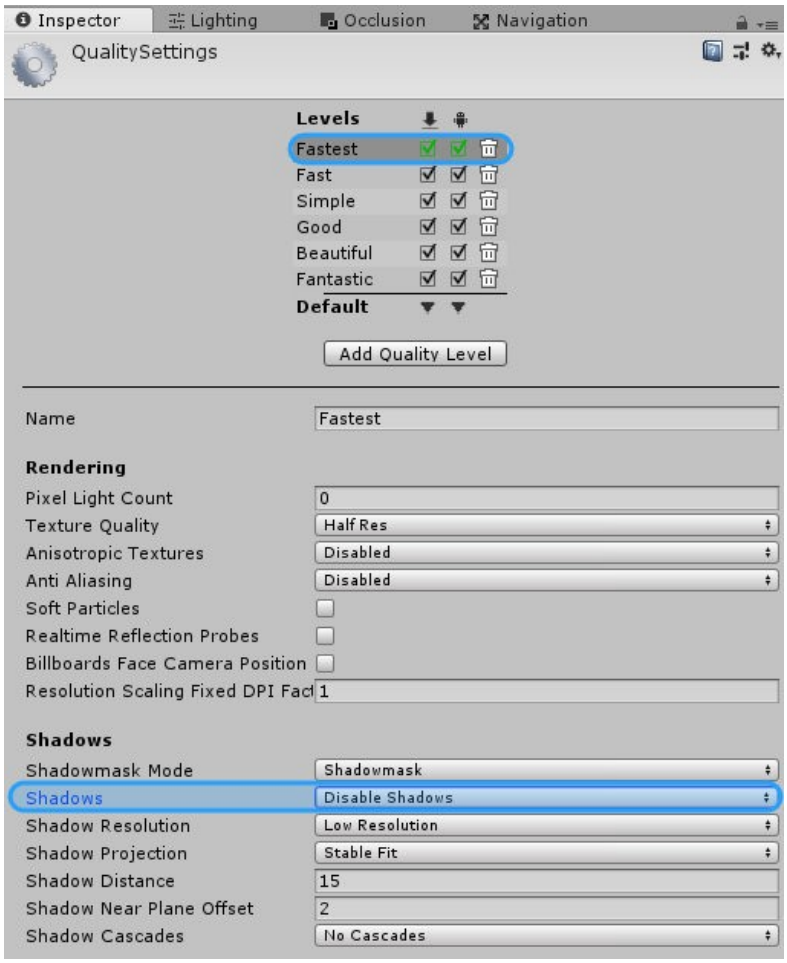


*If you set **Rendering Path** to **Deferred**, the game might slow down a lot on mobile!

6. Disable **Real-time Shadows** (Optional – for much better performance).

Go to *Edit > Project Settings > Quality*

Select Android quality level, which is in **Green Color**, for me, it's **Fastest**. And set **Shadows** to **Disable Shadows**.



*Realtime shadows are not recommended to use on mobile devices because they decrease the performance significantly. You should bake them instead. Or use them only on high-end devices.

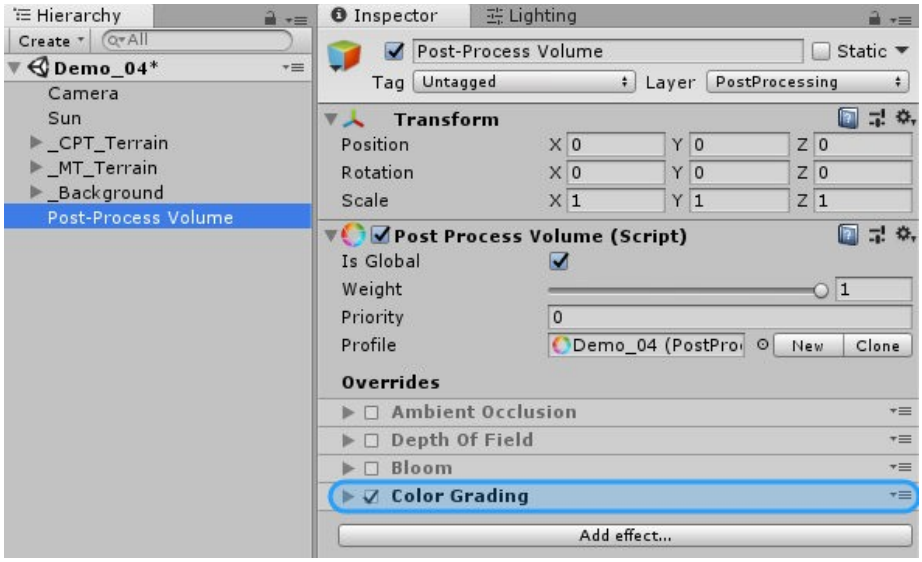
7. Import and enable **Post Processing** image effects (Optional – **Big** performance hit for mobile devices!).

Go to the part of the documentation: [Post-Processing in Unity 2019.4 LTS and up]

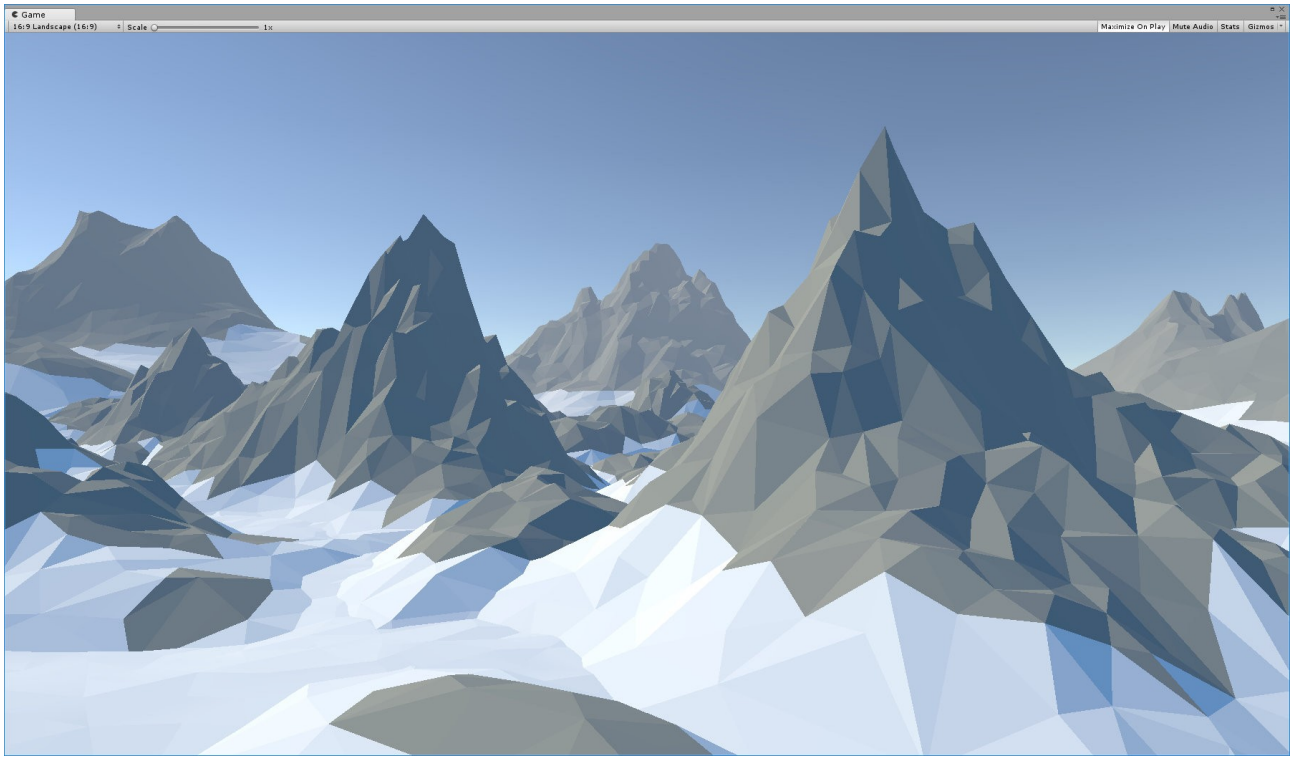
And follow those steps.

*I highly recommend not to use Post-Processing effects on mobile devices because it's a huge hit to performance!

If you will use **Post-Processing** effects, use **Color Grading** only, and leave everything else disabled. It will look nice, and it will work great on high-end devices (Tested on Google Pixel 2 XL).



Now your **Demo_04** scene should look like this (if you skipped all **Optional** steps, and with Realtime Shadows **Disabled**):

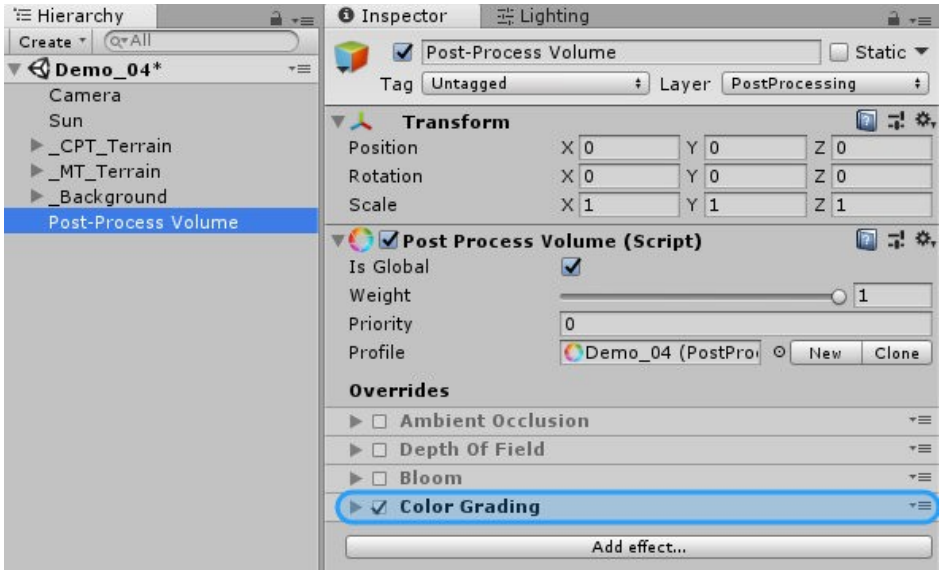


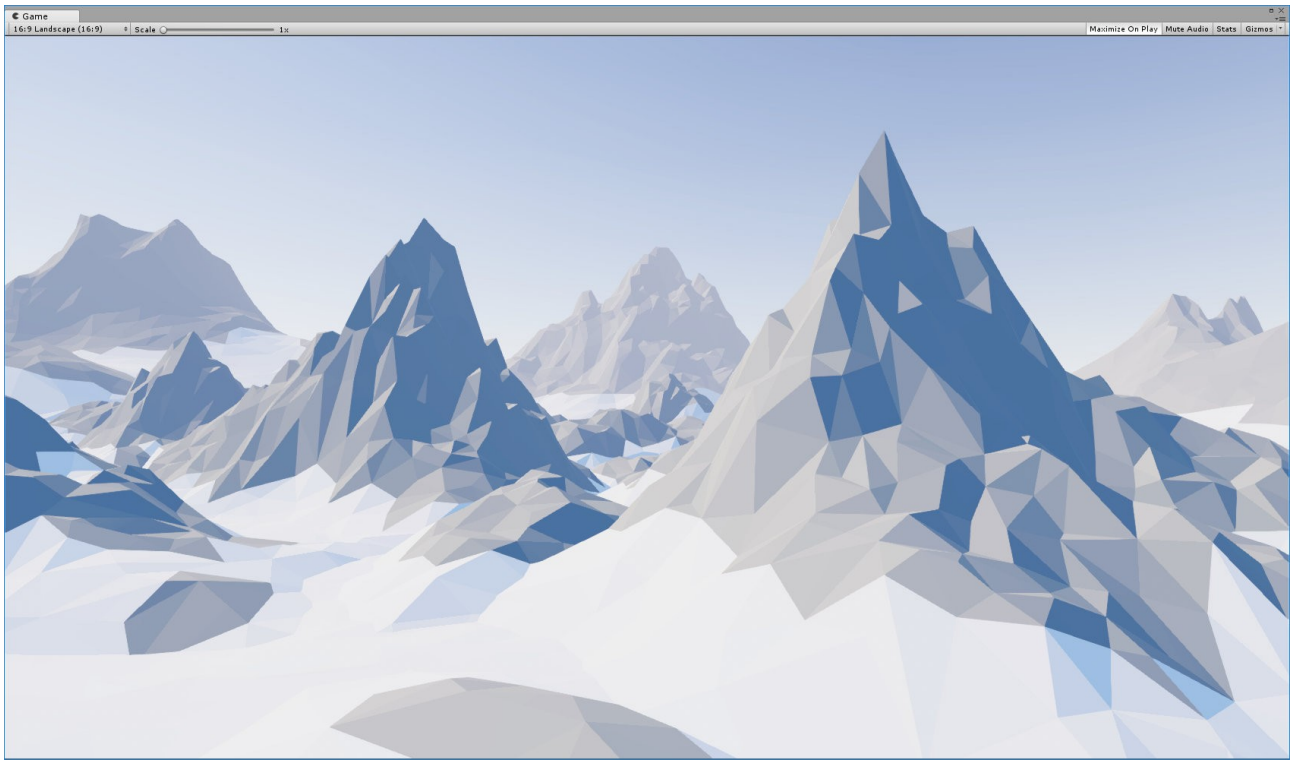
By using **Linear** lighting feature for **Android** and **iOS**, you can achieve much better results than using **Gamma** lighting!

All demo scenes, including **Demo_04,** has been tested on old Xperia Z Ultra (runs at solid 60FPS): without Post Processing effects, using Realtime GI, Linear Color Space, Forward Rendering Path and Real-time Shadows disabled.

*I don't have an **iOS** device, so I can't test it on that!

Demo_04 scene with the same settings + Post Processing (**Color Grading** enabled only)



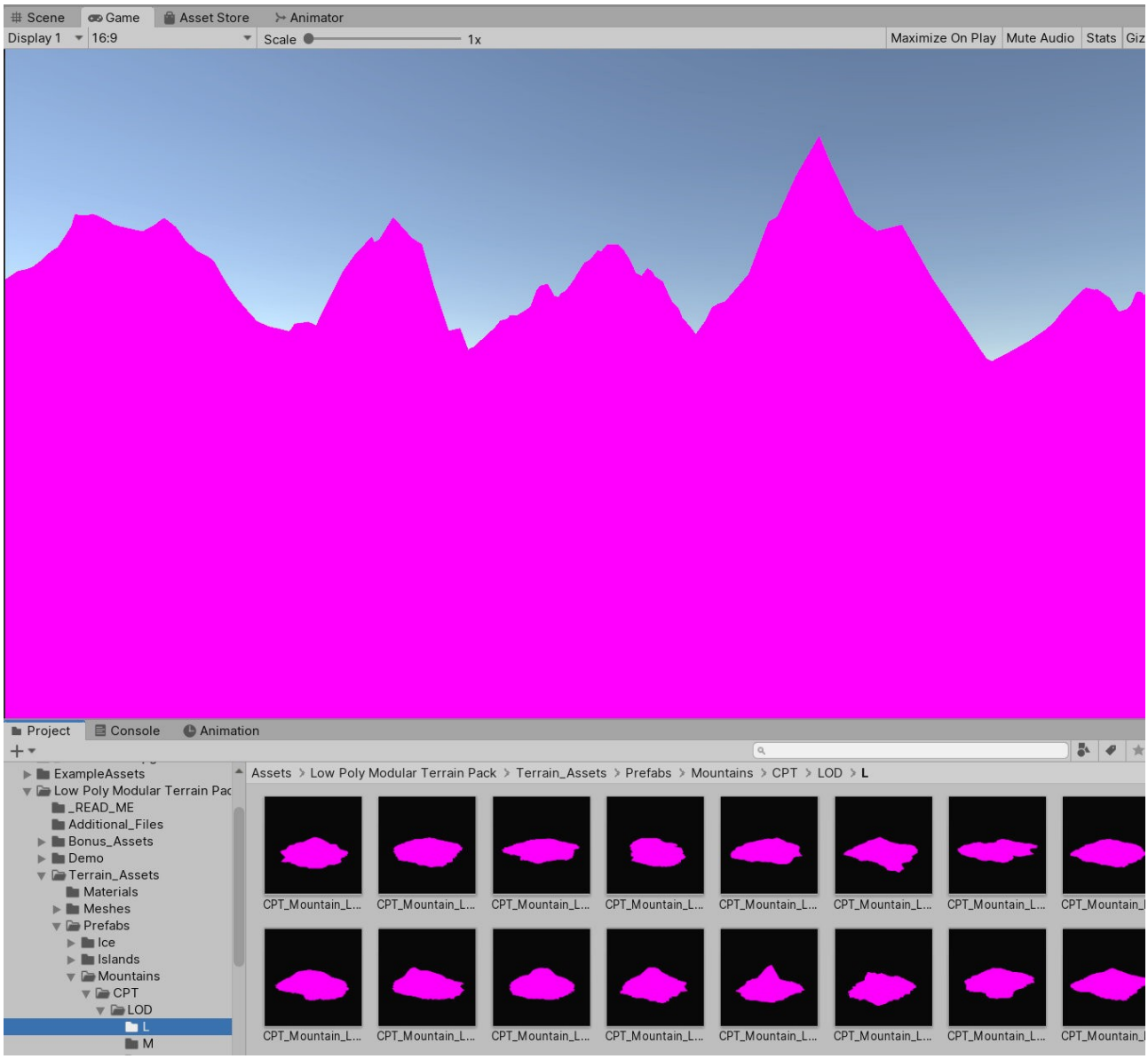


Tested on Google Pixel 2 XL – runs at solid 60fps. Xperia Z Ultra drops to 30fps for using Color Grading.

Unity (URP / Universal 3D)

Fix Pink Materials

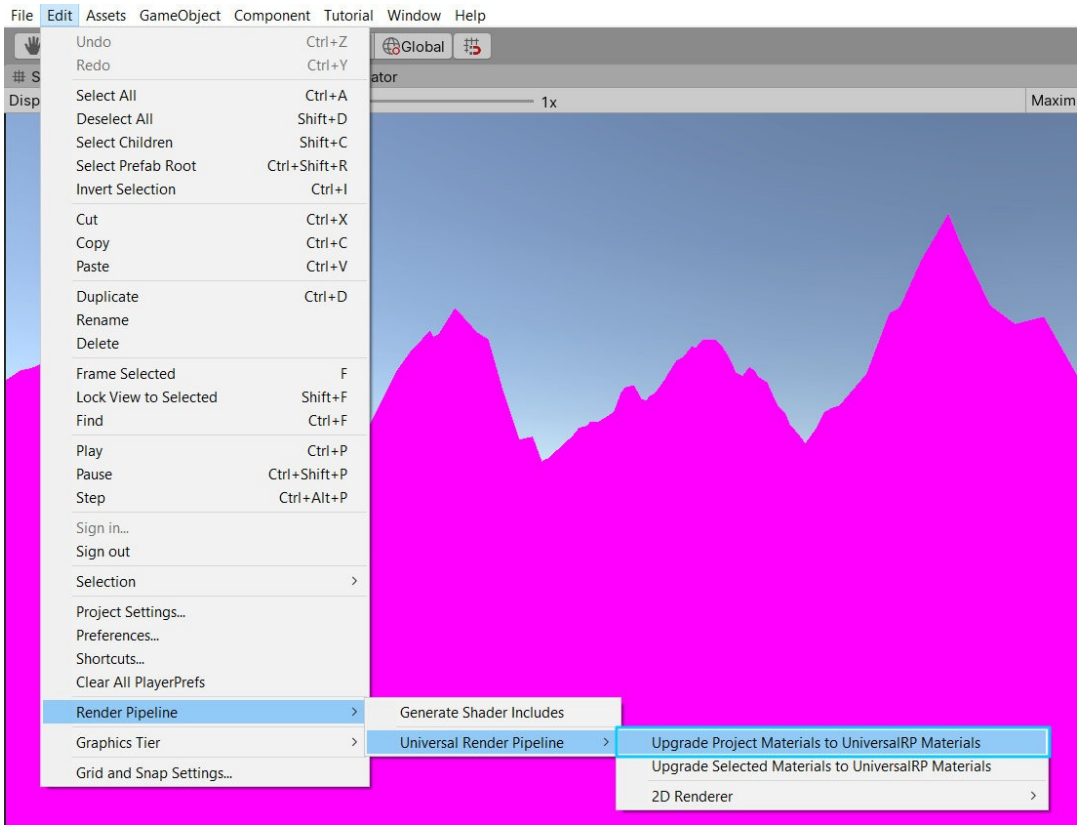
You might encounter pink textures/materials after importing **Low Poly Modular Terrain Pack** to your Unity project, which is using **Universal Render Pipeline (URP) / Universal 3D**.



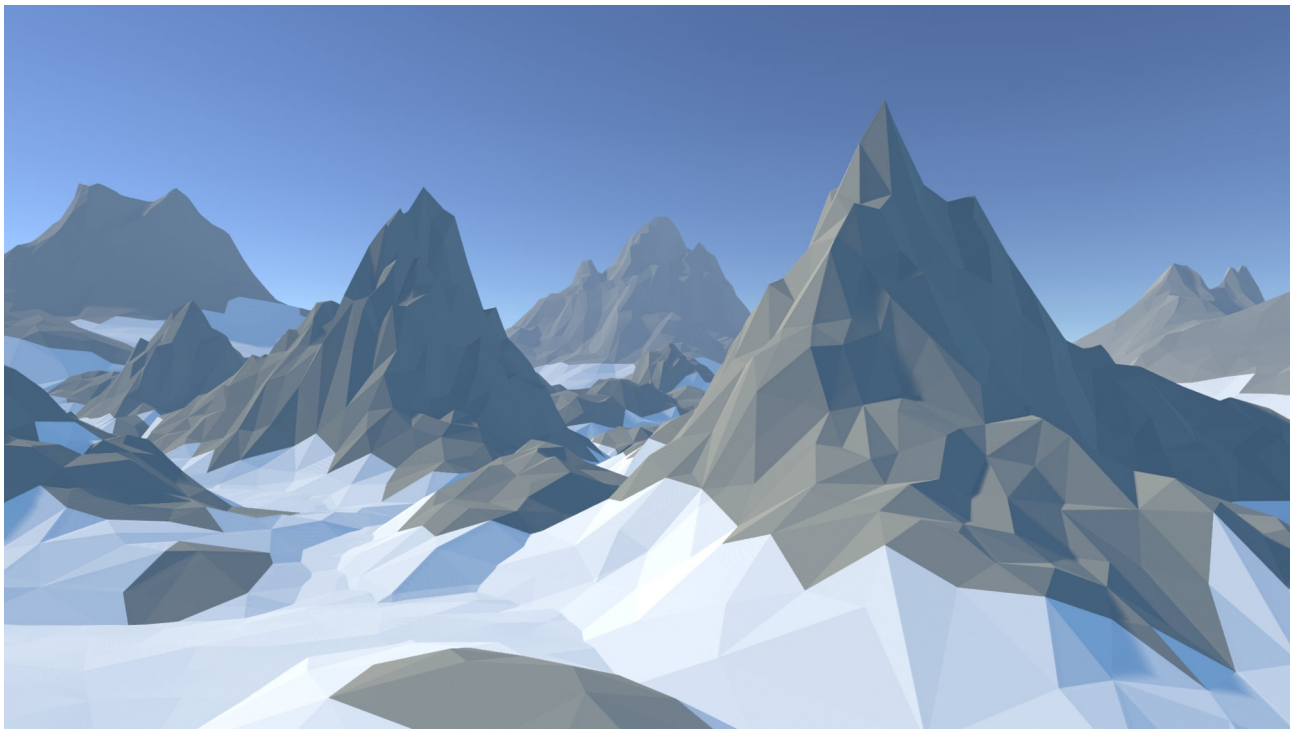
It's because all of **Low Poly Modular Terrain Pack** assets use material with a default **Standard Unity shader**. **URP / Universal 3D** use different materials and shaders. So we need to change all materials from **Standard shader** to **Universal Render Pipeline/Lit** shader.

For older versions of Unity, for example 2019.4

Go to *Edit > Render Pipeline > Universal Render Pipeline >* **Upgrade Project Materials to UniversalRP Materials**

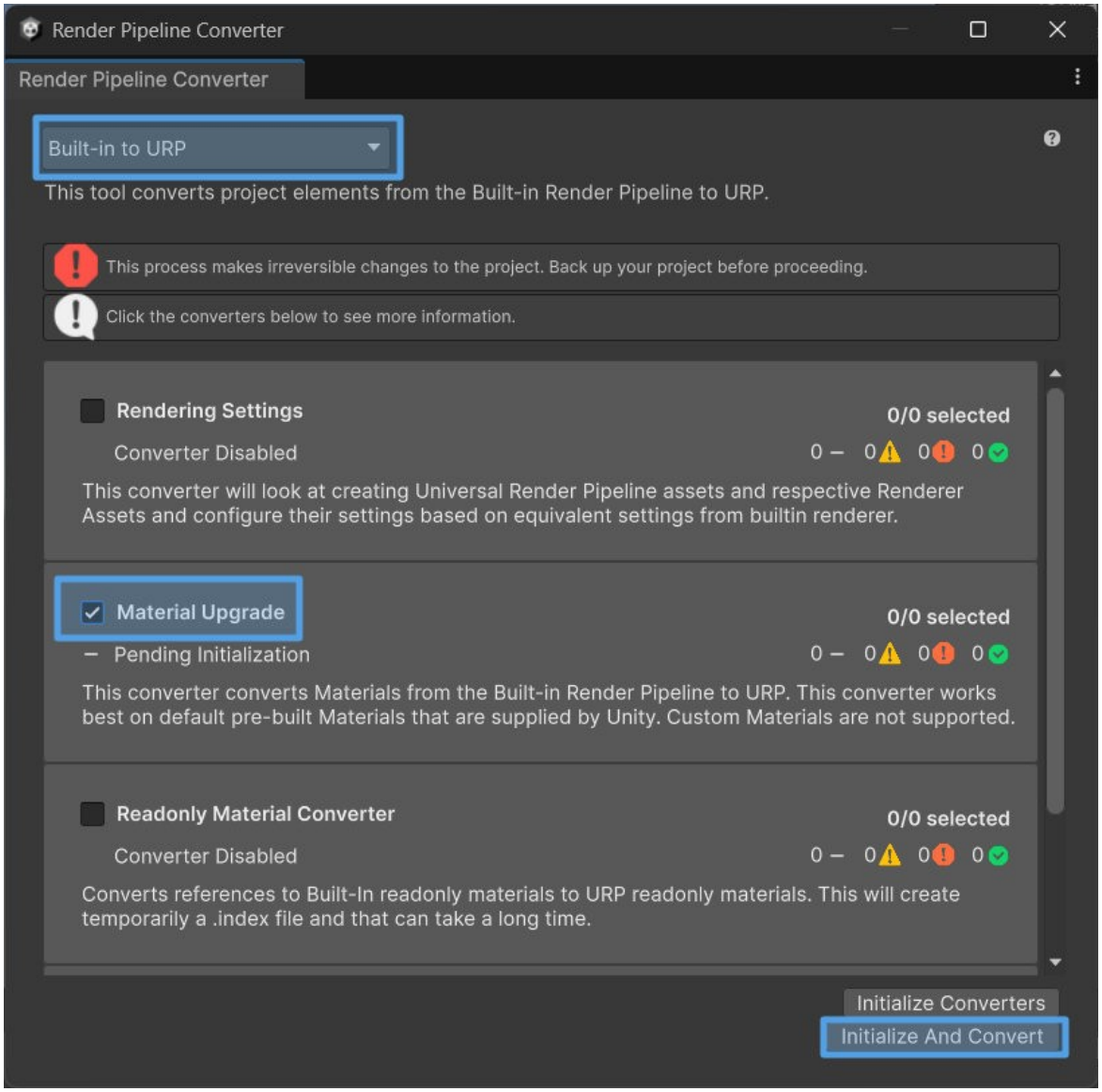


And it's done!



For newer versions of Unity, for example Unity 2021 - Unity 6

Go to *Window > Rendering >* ***Render Pipeline Converter.***

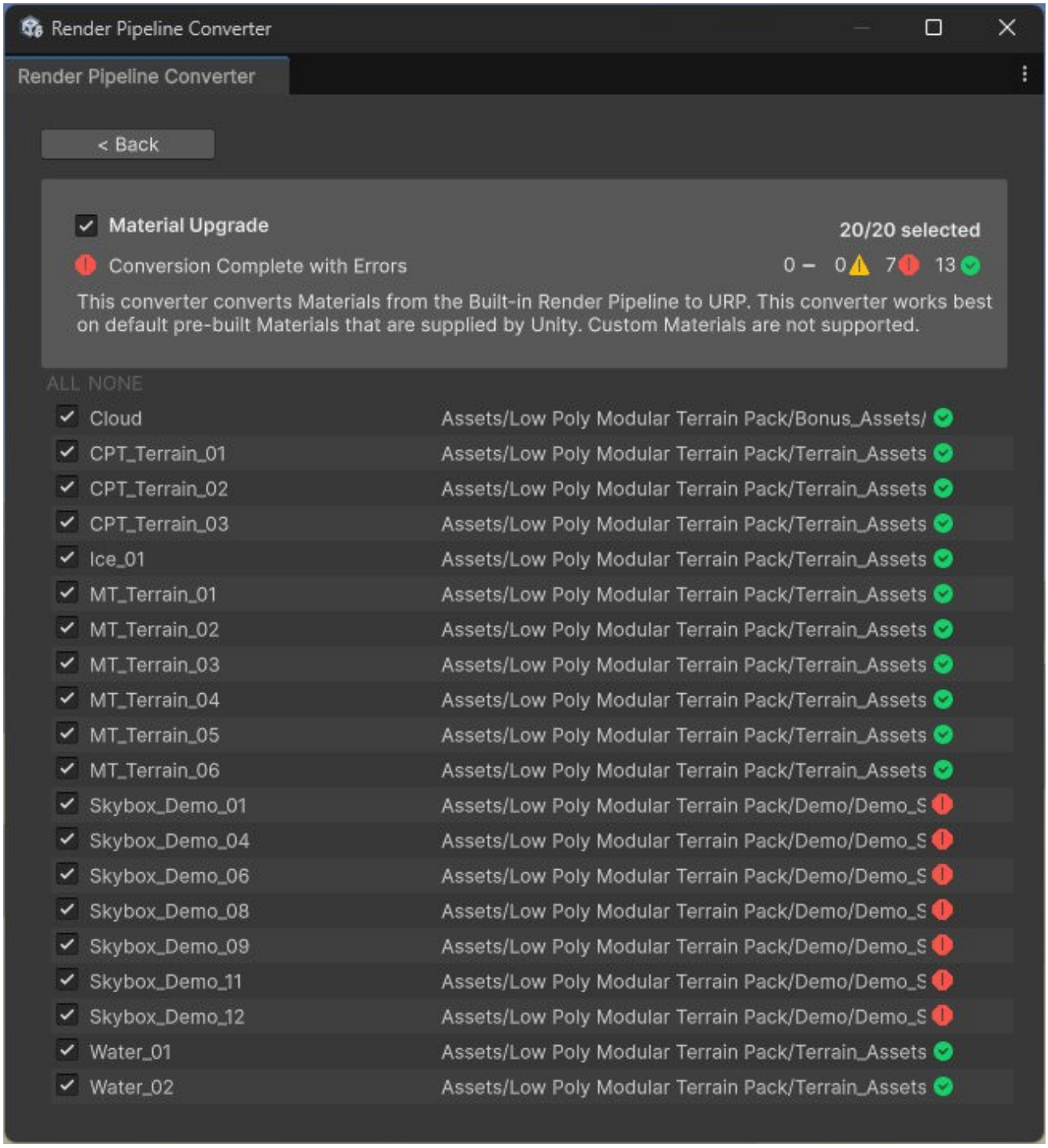


Make sure that the **Built-in to URP** is set. Enable **Material Upgrade** and press **Initialize And Convert**.

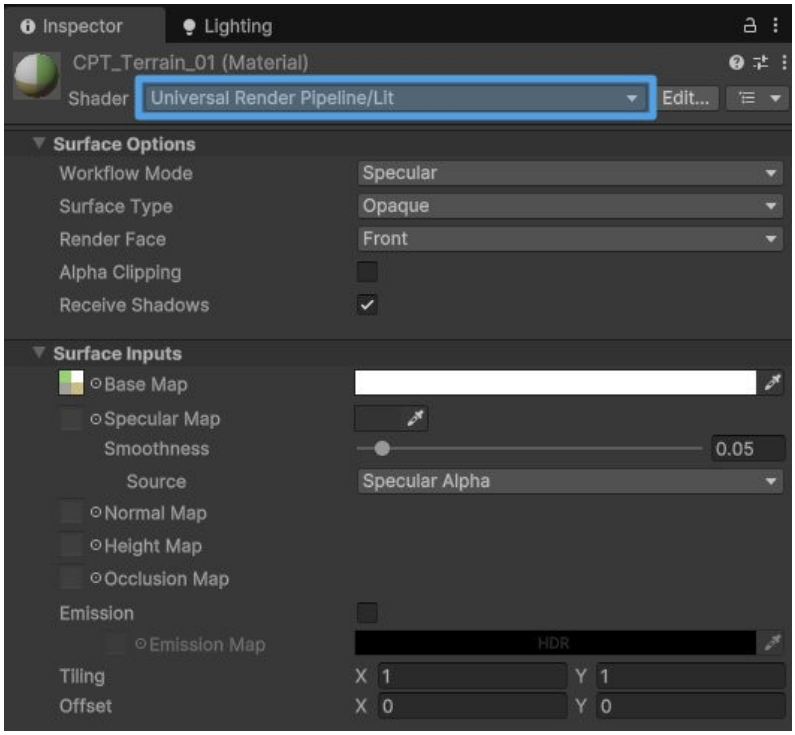
After it's completed you will see the message ***'Conversion Completed with Errors.'***

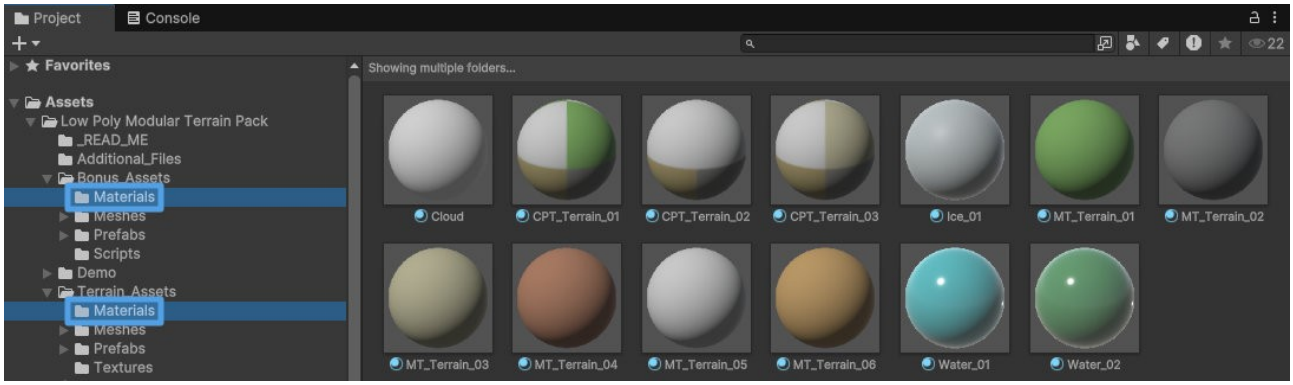
This is what we want!

You can click on the **Material Upgrade** section to see which of the Materials were completed conversions and which ones not.



All of the Materials that used **Standard Unity shader** (Pink Materials) were converted to the **Universal Render Pipeline/Lit** shader.

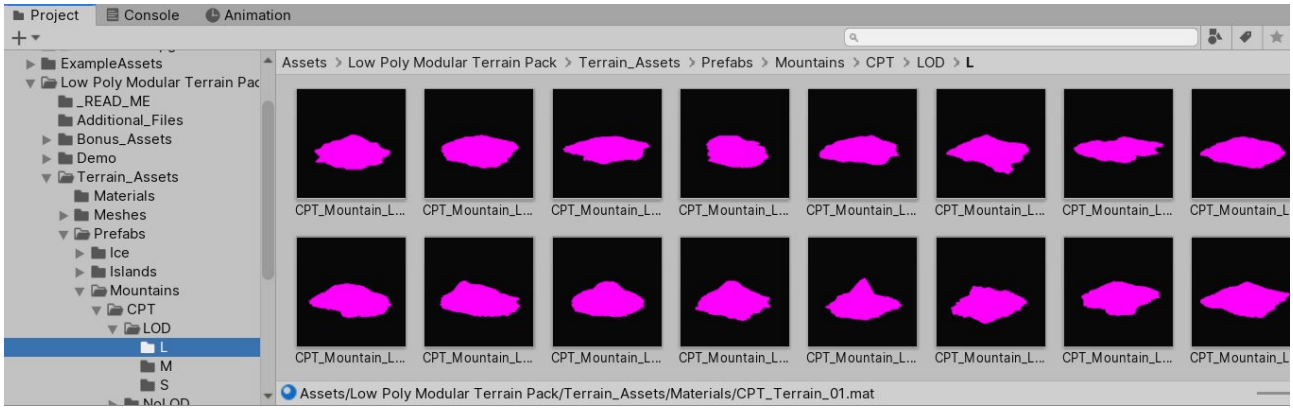




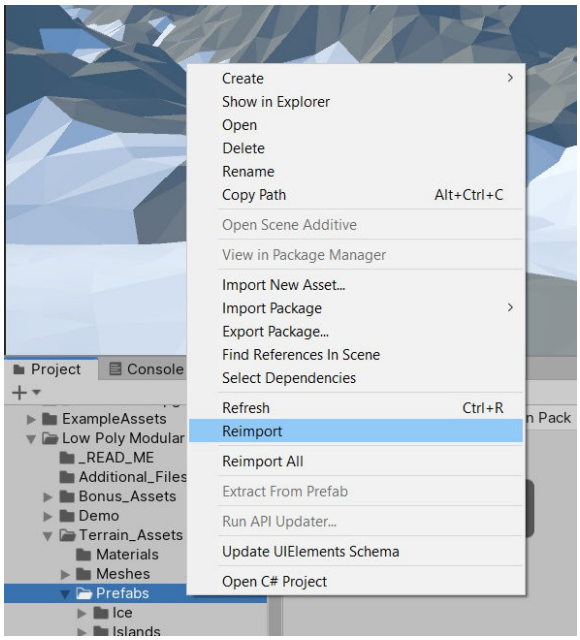
*Other Materials that were showing conversion errors use different shaders, which don't need to be converted for URP.

*You can also do conversion manually by selecting **Material** and changing the **Shader** but it's much slower.

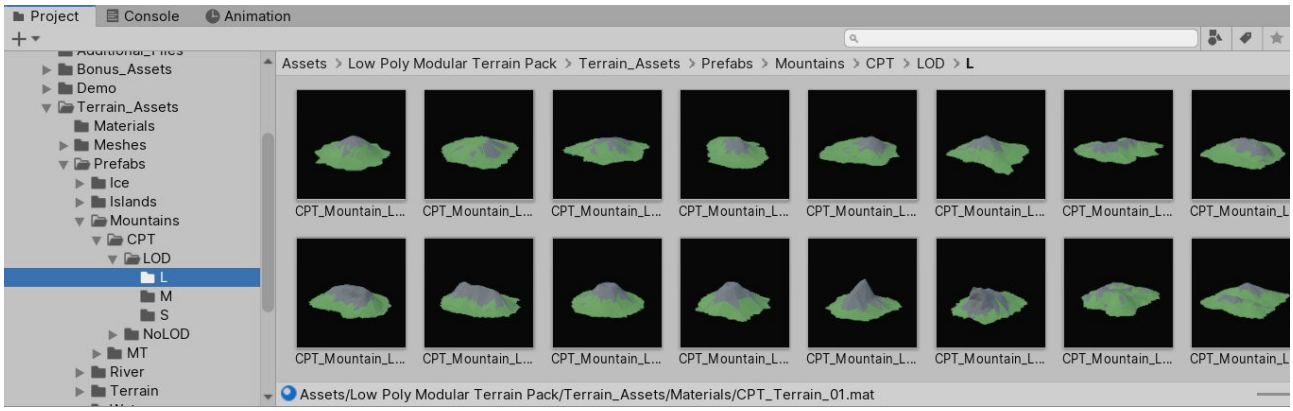
Now if you go to *Low Poly Modular Terrain Pack/Terrain_Assets/Prefabs/Mountains/CPT/LOD/L* - or inside any other prefab folder. You might see all of the prefabs in **Pink** color even after converting materials.



To fix that - press **RMouse** on the **„Prefabs“** folder and select **Reimport**.

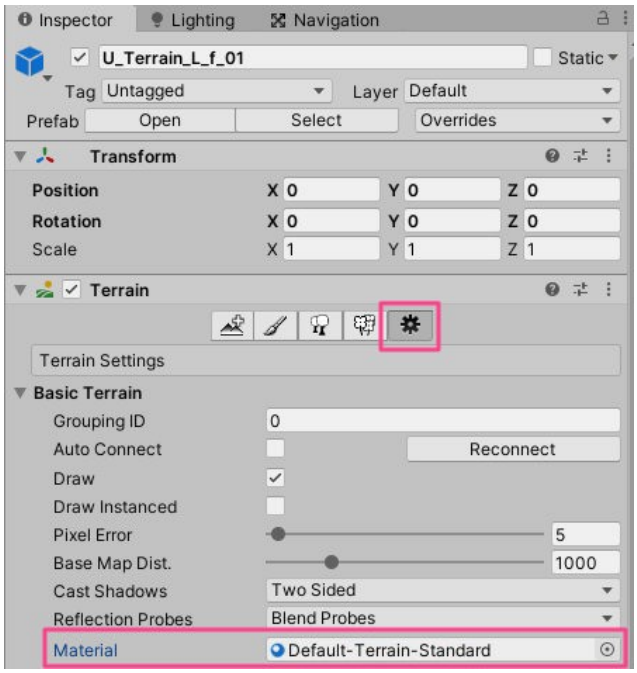


And it‘s fixed!



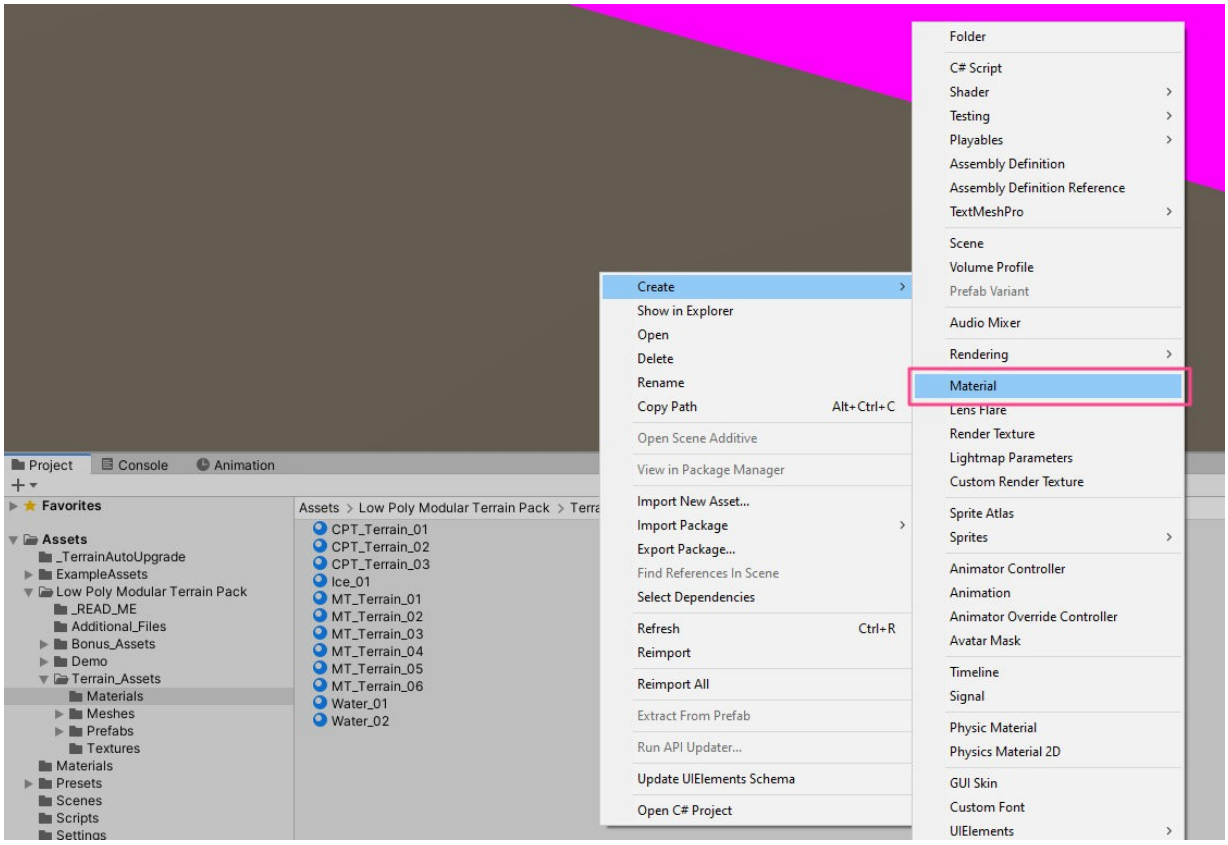
How to fix pink textures on U_Terrain in URP

U_Terrain uses **Default-Terrain-Standard** material from a built-in render pipeline.

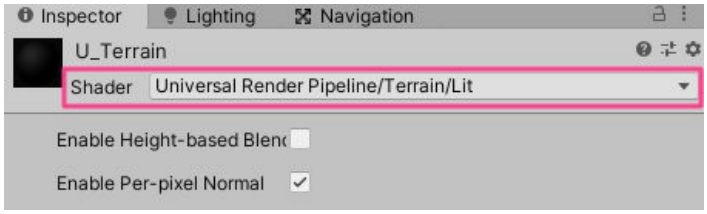


URP / Universal 3D uses completely different terrain material, which you need to create and apply manually!

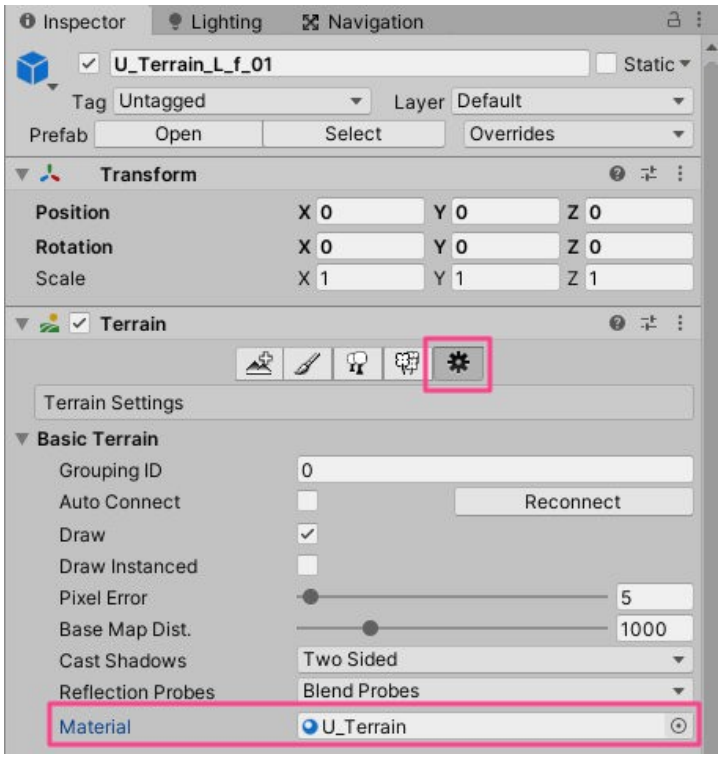
Create a new Material. I will call it ***U_Terrain***:



Select newly created Material ***U_Terrain*** and change **Shader** to **Universal Render Pipeline/Terrain/Lit**



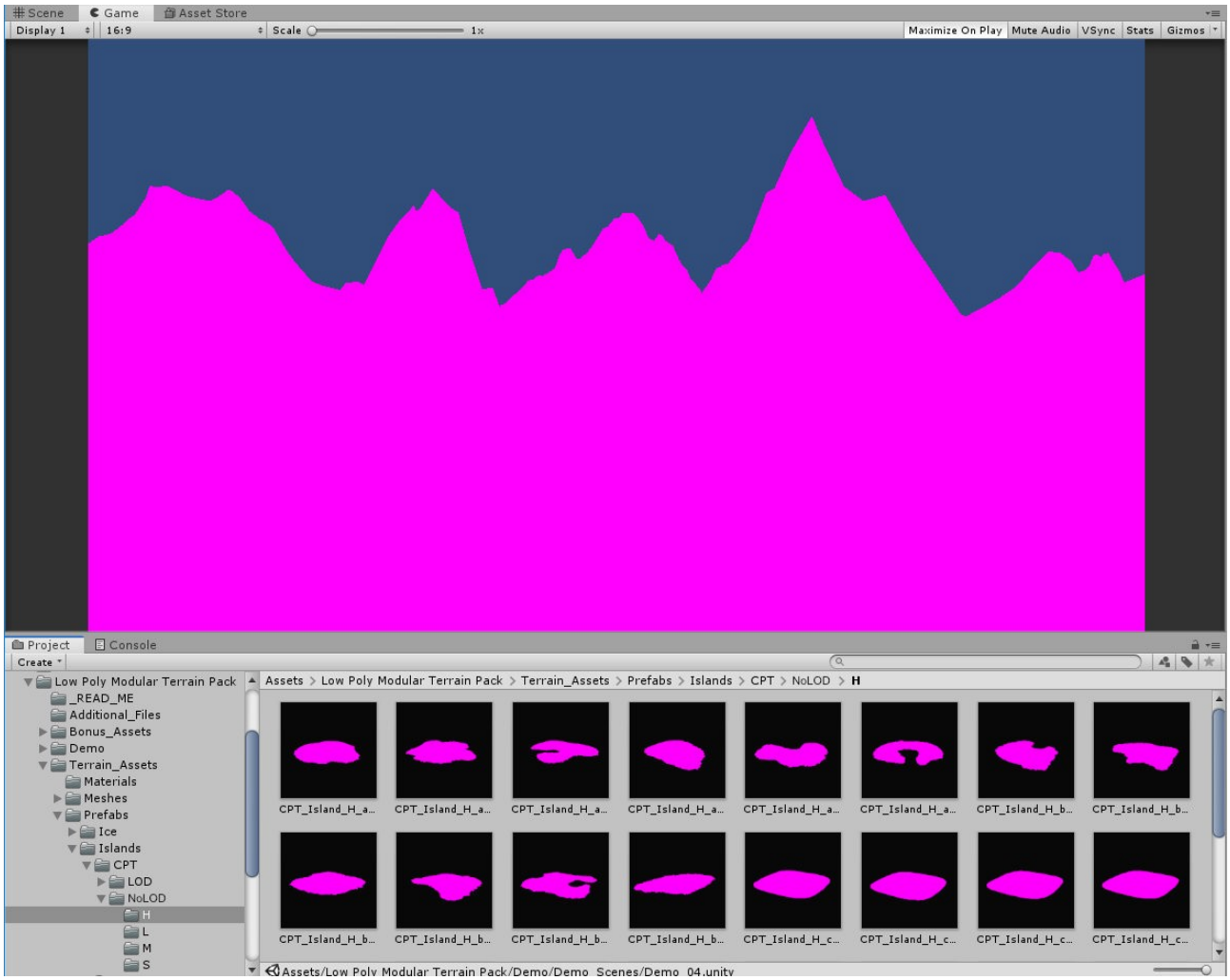
And apply it to the **U_Terrain** prefab



Unity (HDRP / High Definition 3D)

Fix Pink Materials

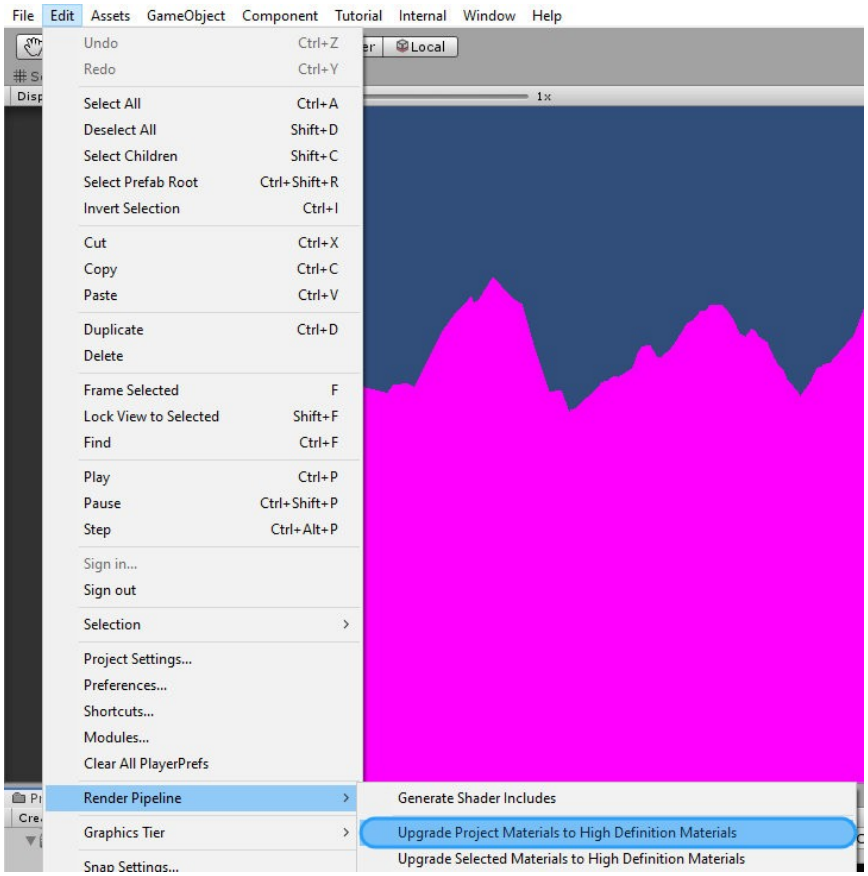
You might encounter pink textures/materials after importing **Low Poly Modular Terrain Pack** to your Unity **High Definition Render Pipeline (HDRP / High Definition 3D)** project.



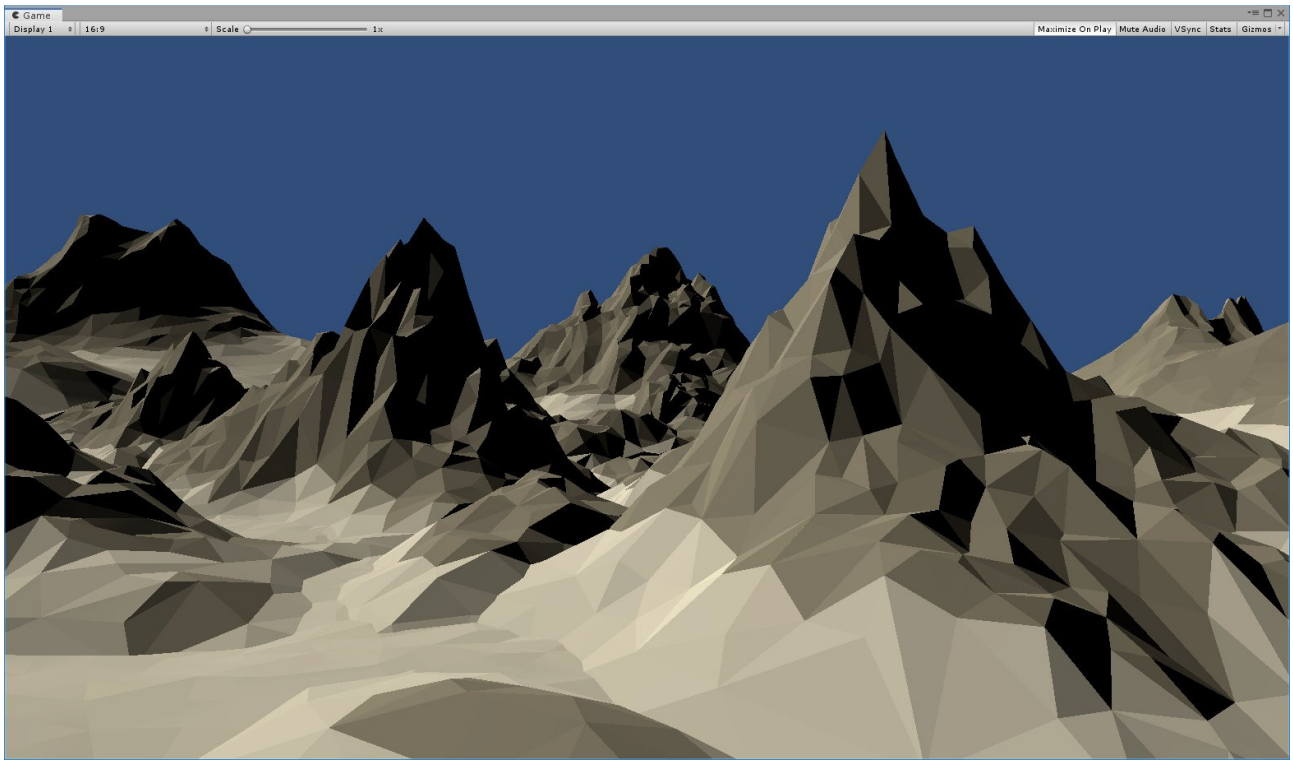
It's because all of **Low Poly Modular Terrain Pack** assets use materials with a default **Standard Unity shader**. But **HDRP / High Definition 3D** uses different materials and shaders, so we need to change all materials from **Standard shader** to **HDRP/Lit** shader.

For older versions of Unity, for example 2019.4

Go to *Edit* > *Render Pipeline* > **Upgrade Project Materials to High Definition Materials**



And it's done! Almost.



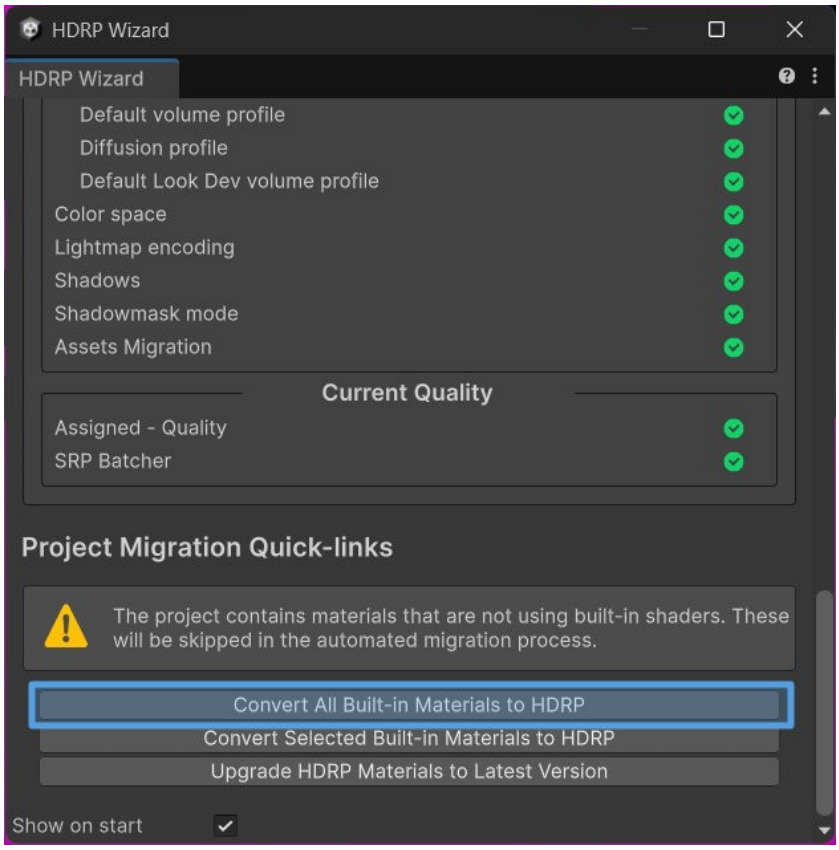
For newer versions of Unity, for example Unity 2021 - Unity 6

Go to Edit *> Rendering > Materials >* ***Convert All Built-In Materials to HDRP***.

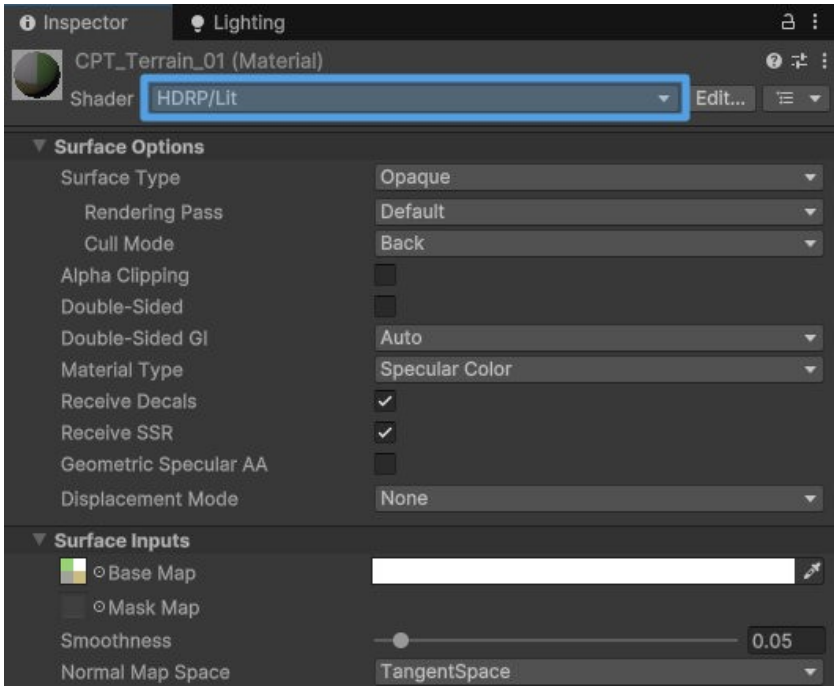
Or

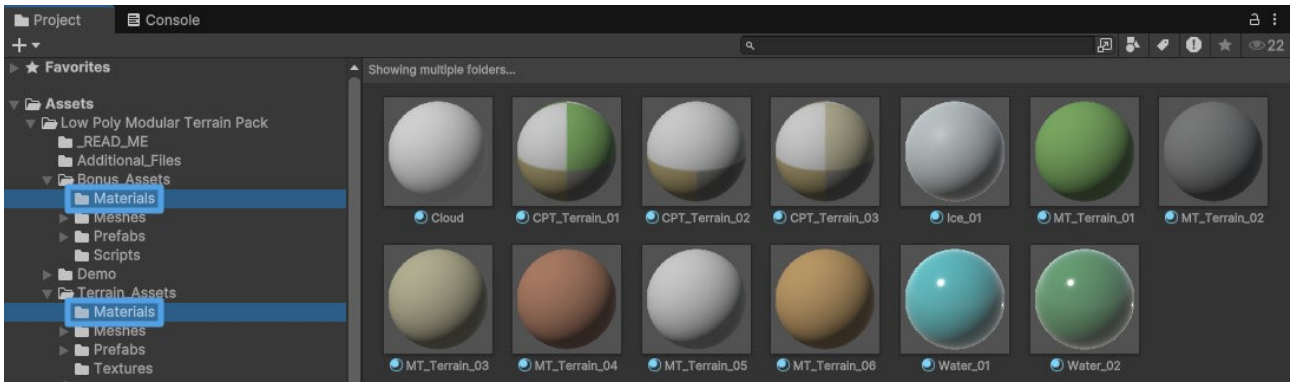
Go to *Window > Rendering >* ***HDRP Wizard.***

Scroll to the bottom and press **Convert All Built-In Materials to HDRP**.



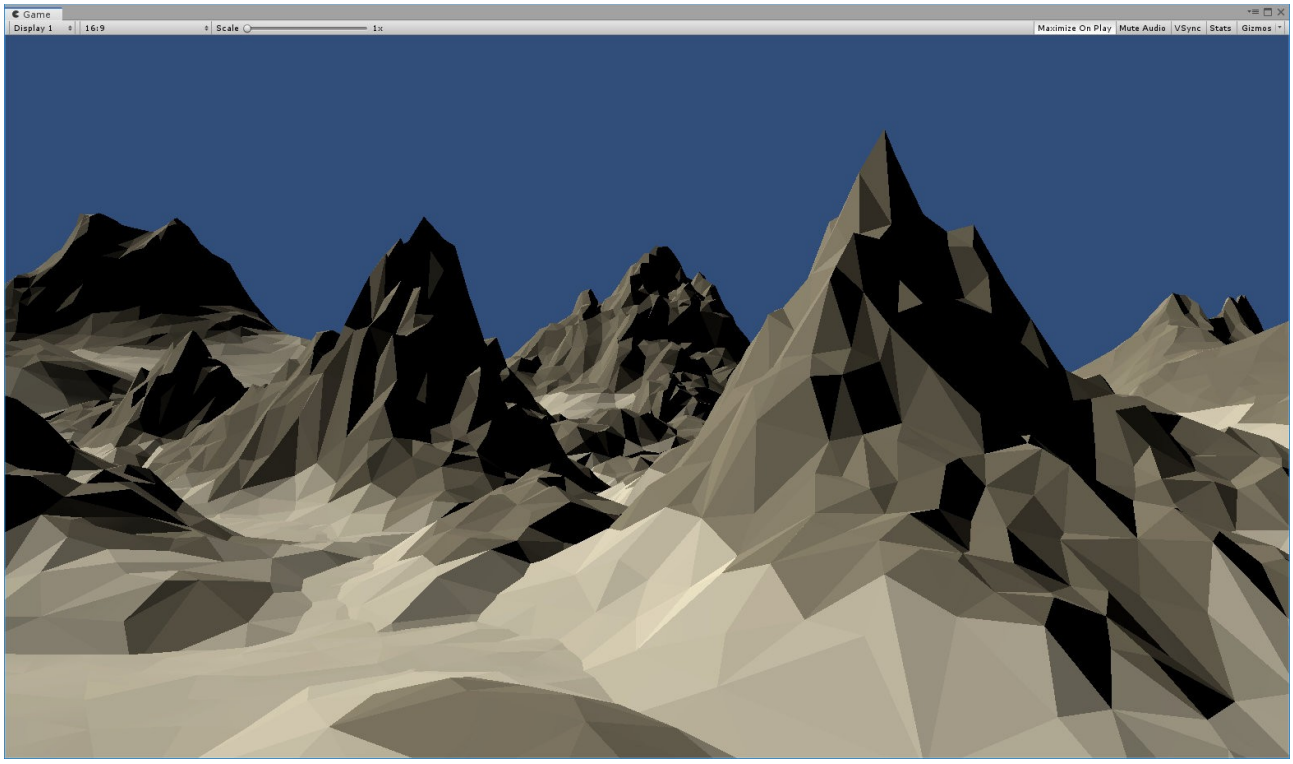
All of the Materials that used **Standard Unity shader** (Pink Materials) were converted to the **HDRP/Lit**.





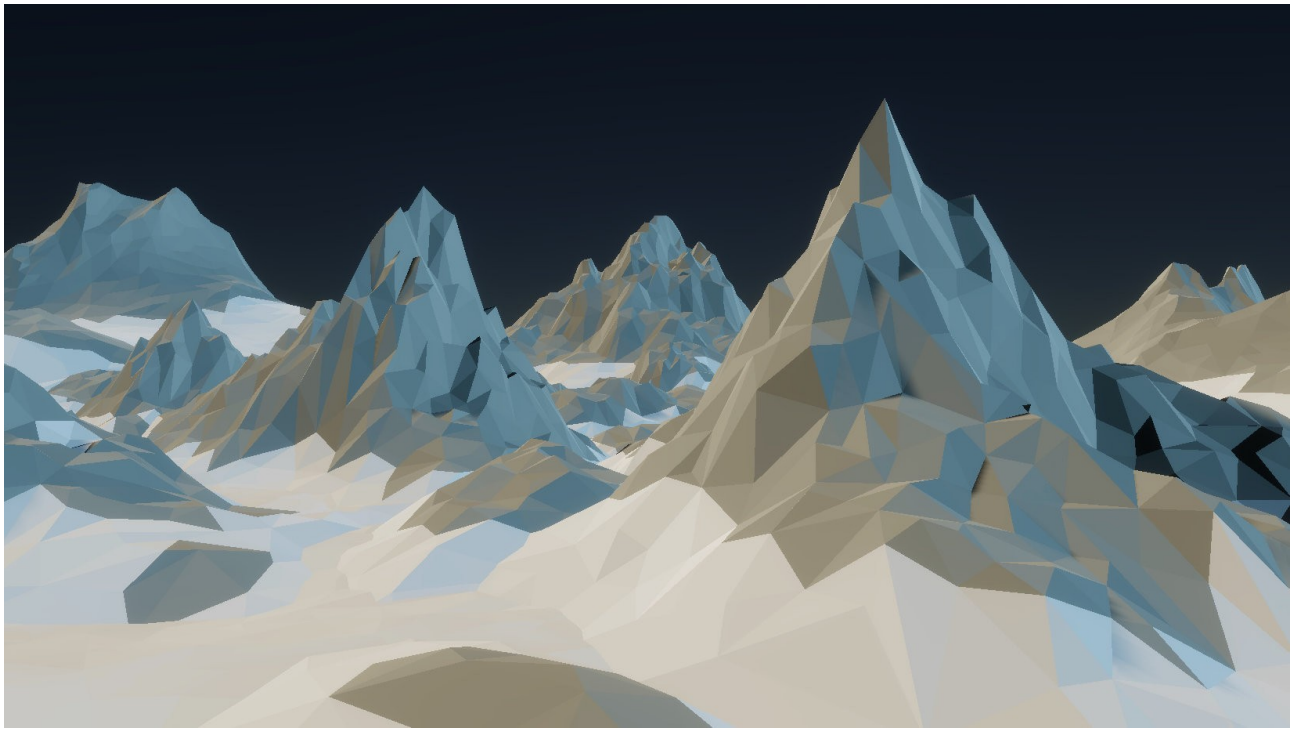
*You can also do a conversion manually by selecting the **Material** and changing the **Shader**, but it's much slower.

Your scene should look something like this:



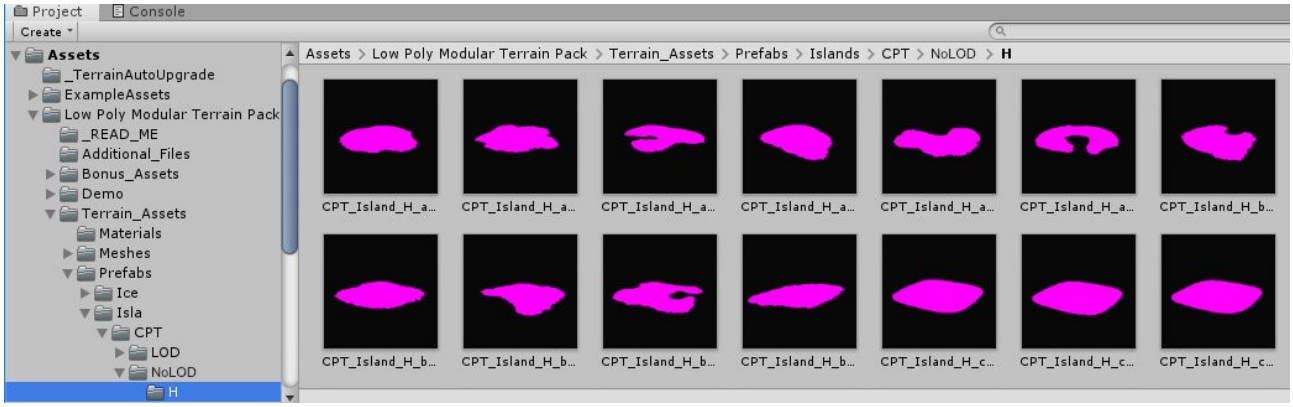
Unity 2019.4 example

Or like this in a newer versions of Unity:

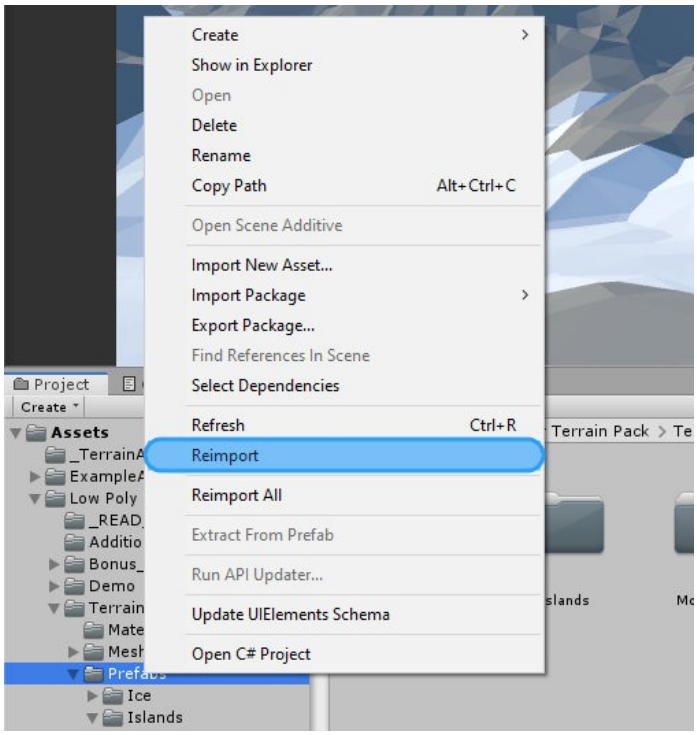


Unity 6 example

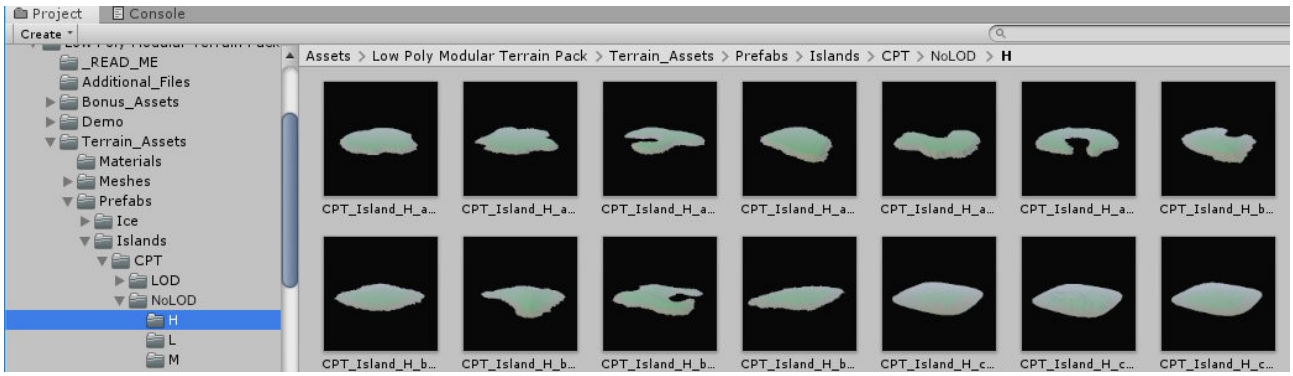
Now if you go to *Low Poly Modular Terrain Pack/Terrain_Assets/Prefabs/Islands/CPT/NoLOD/H* - or inside any other Terrain folder. You might see all of the prefabs in **Pink** color.



If that is the case, to fix that - press **Right Mouse Button** on **Prefabs** folder and select **Reimport**.

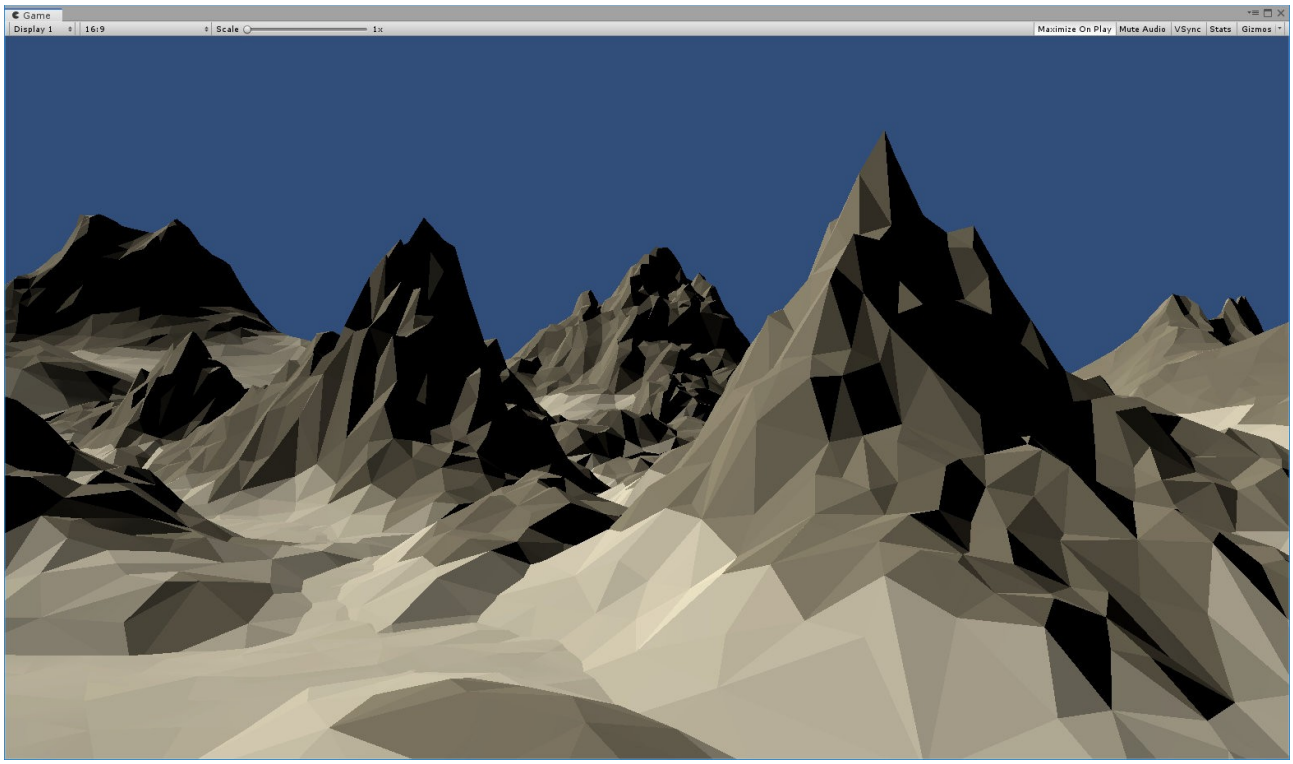


And it's fixed!



Fix Dark Lighting in HDRP / High Definition 3D

If the demo scenes look dark, without any lighting like this:

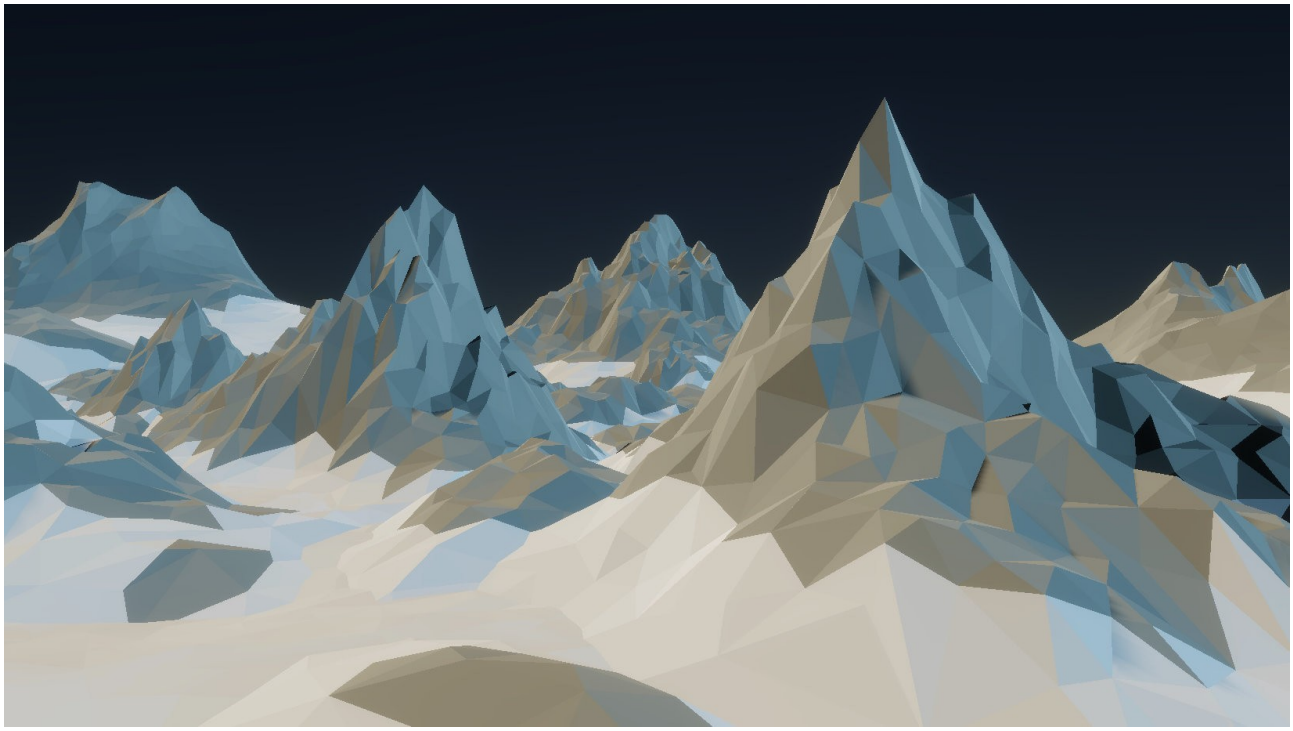


Unity 2019.4 example in the image above.

Demo_04 example, located at *Low Poly Modular Terrain Pack/Demo/Demo_Scenes*

Try selecting the **Directional Light (Sun)** in the **Hierarchy** (disable it and enable again) to update the lighting in the scene.

If your scene is lit but it has black shadows and the dark skybox like this:



Unity 6 example in the image above.

You need to add **Scene Settings / *Sky and Fog Global Volume*** to the scene.

In older versions of Unity, for example Unity 2019.4

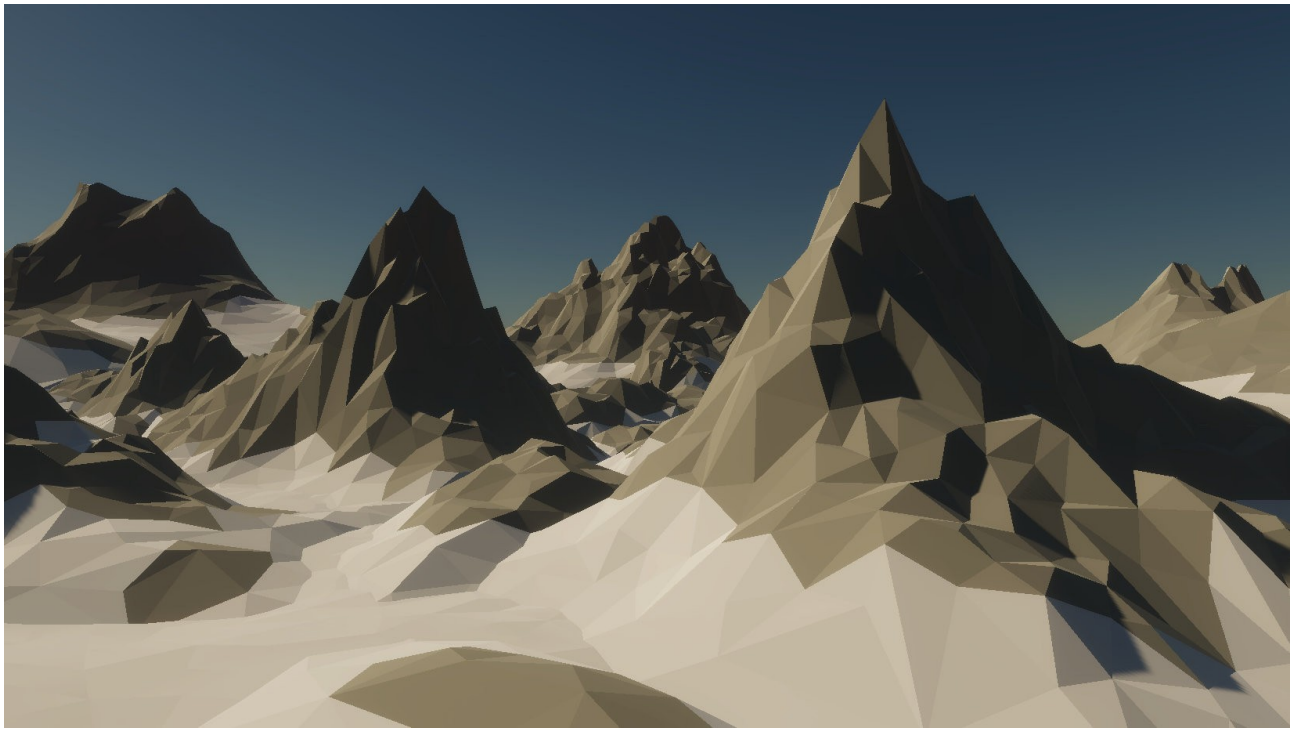
Go to *GameObject > Rendering >* ***Scene Settings***.

In the newer versions of Unity, for example Unity 6

Go to *GameObject > Volume >* ***Sky and Fog Global Volume***.

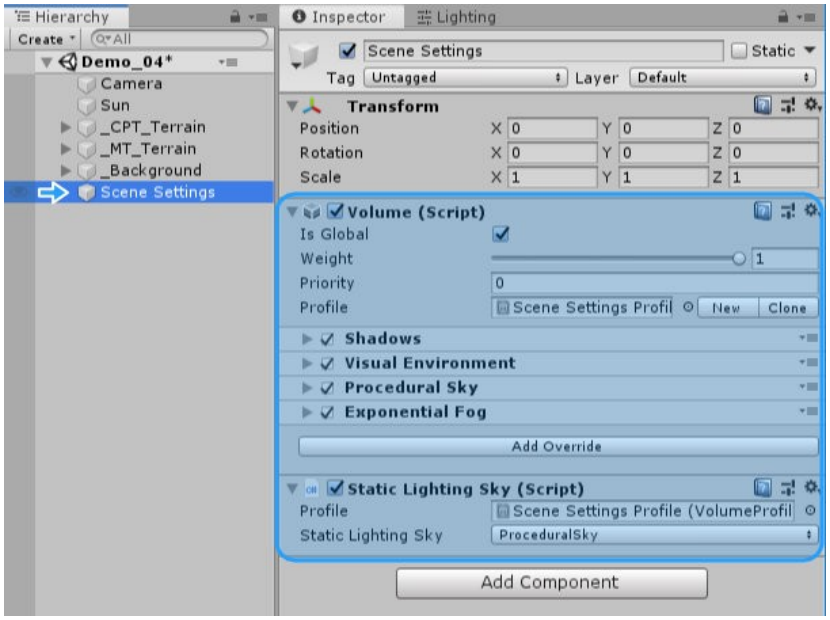
Also, select the **Directional Light (Sun)** in the **Hierarchy**, and inside the **Shadows** section, change **Resolution** to **High** or **Ultra** to make it look much better.

And you will see that the **Skybox** is applied to the scene right away + High resolution shadows.

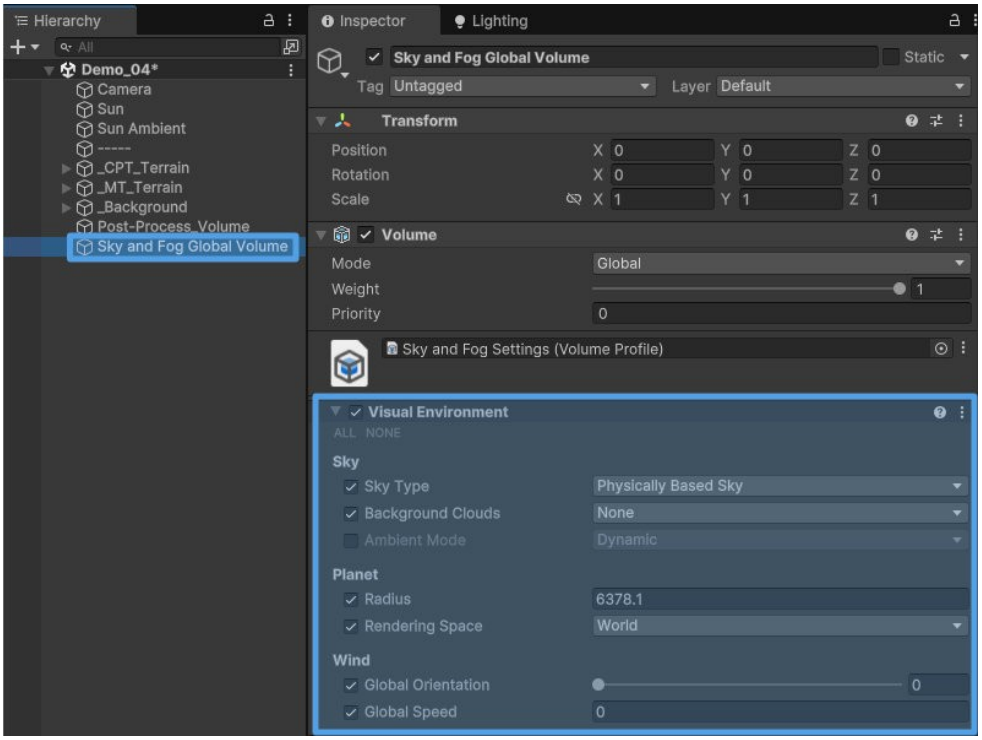


It could still look a bit dark and different compared it to how it would look in Built-In Render Pipeline or URP.

With the **Scene Settings / Sky and Fog Global Volume** selected, you can change a bunch of scene settings like (Shadows, Skybox, Fog, and much more).



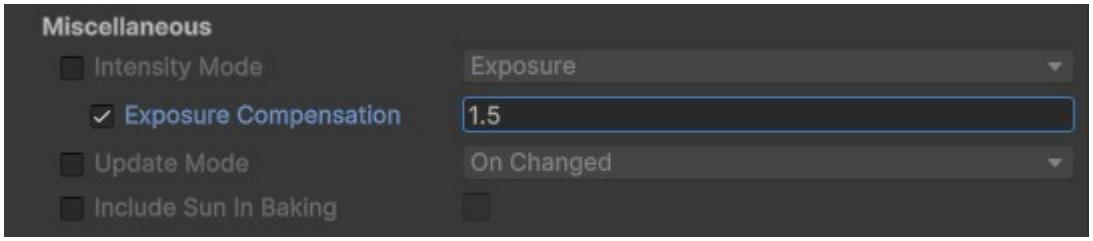
Unity 2019.4 example



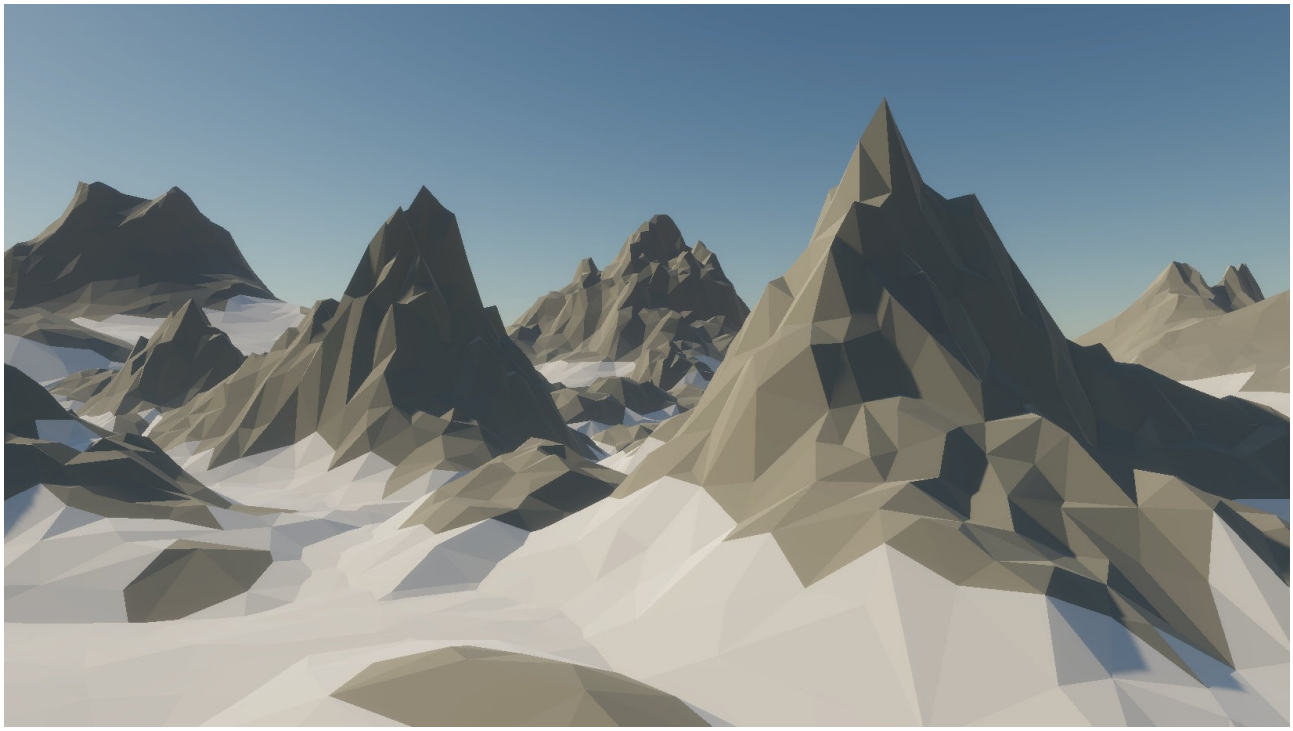
Unity 6 example

You need to play a bit with all of those settings to achieve similar results which you can get by default using Unity without HDRP.

For example, to make the scene lighter, you can change the **Exposure Compensation** to something like **1.5**.



Now, the scene looks brighter:

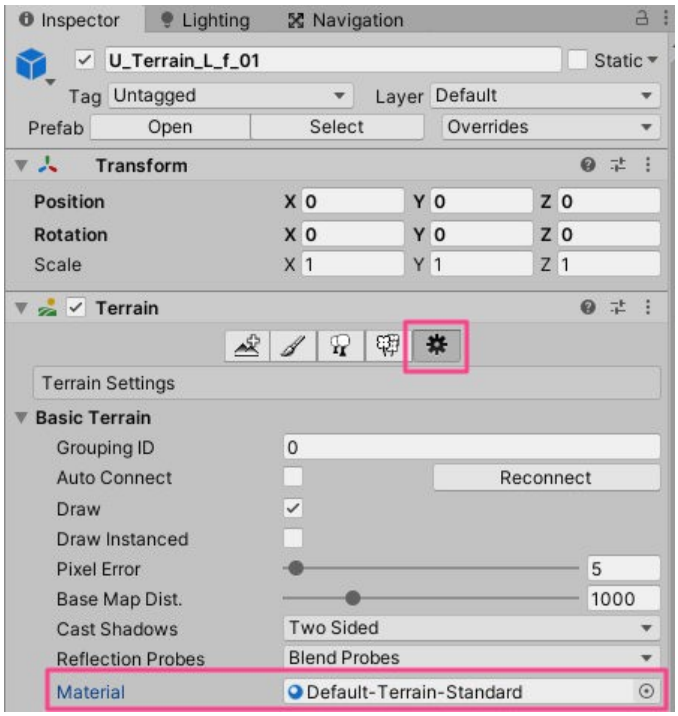


You can also select the '**Sun Ambient**' in the scene **Hierarchy**, and disable it, because Unity HDRP generates the second Sun in the scene by using 2 Directional Lights.

Select the **'Sun'** and play with **Intensity** too.

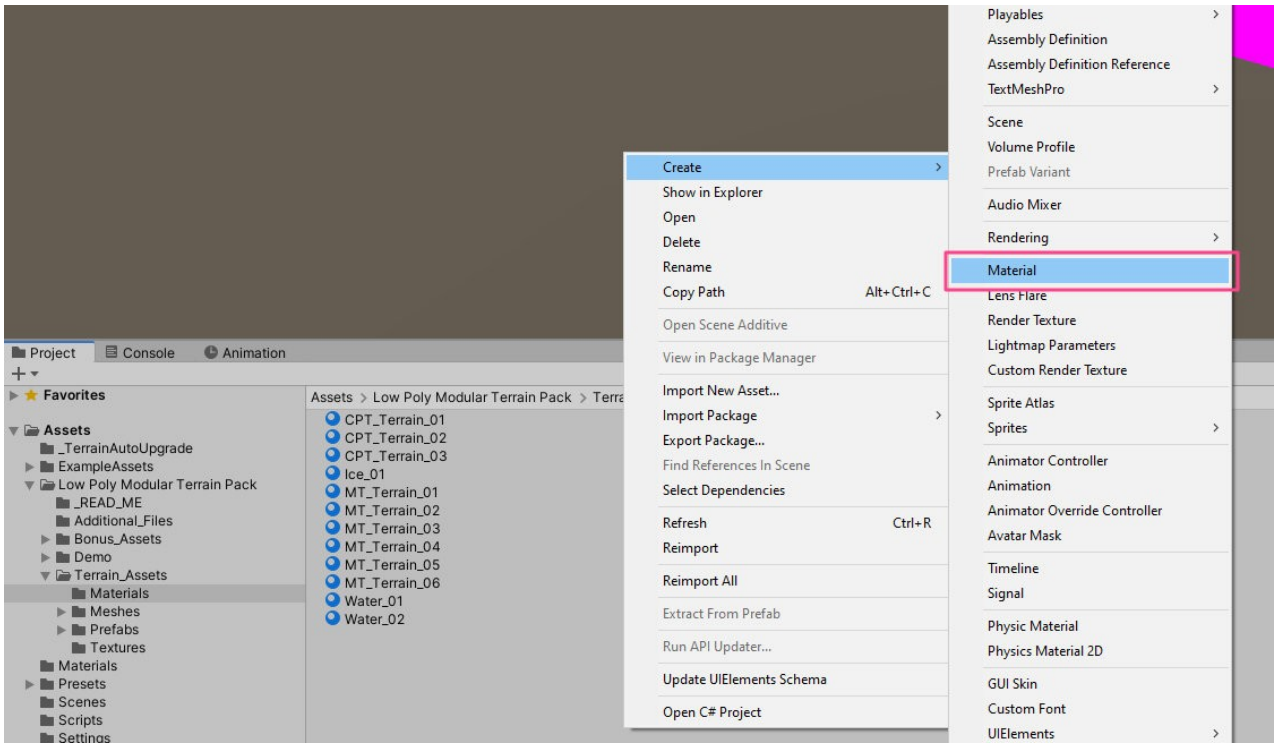
How To Fix Pink Materials On U_Terrain In HDRP / High Definition 3D

U_Terrain uses **Default-Terrain-Standard** material from a built-in render pipeline.

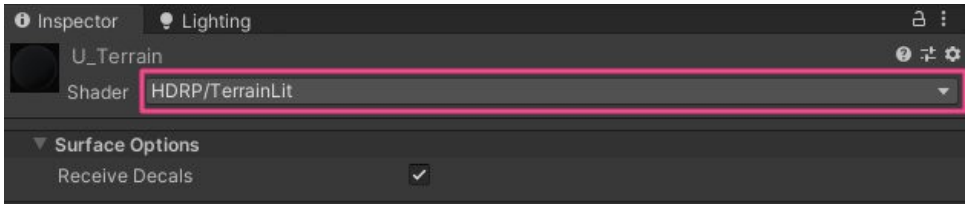


HDRP / High Definiition 3D uses completely different terrain material, which you need to create and apply manually!

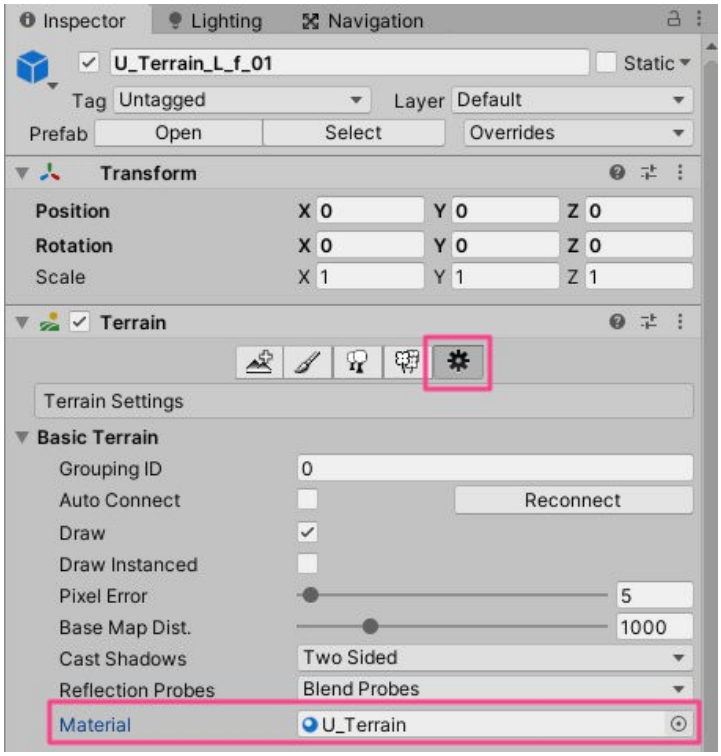
Create a new Material. I will call it ***U_Terrain***:



Select newly created Material ***U_Terrain*** and change **Shader** to **HDRP/TerrainLit**



And apply it to the **U_Terrain** prefab

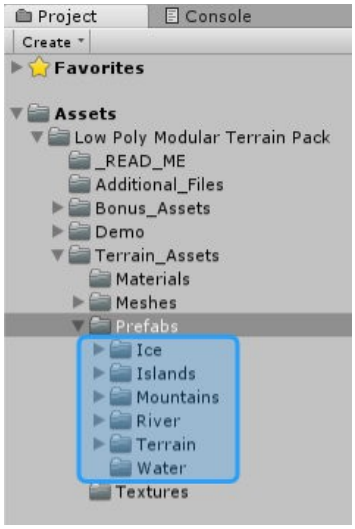


How to use “Low Poly Modular Terrain Pack”

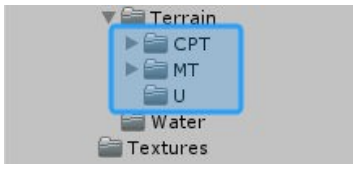
Watch the VIDEO TUTORIAL! Or follow the steps below.

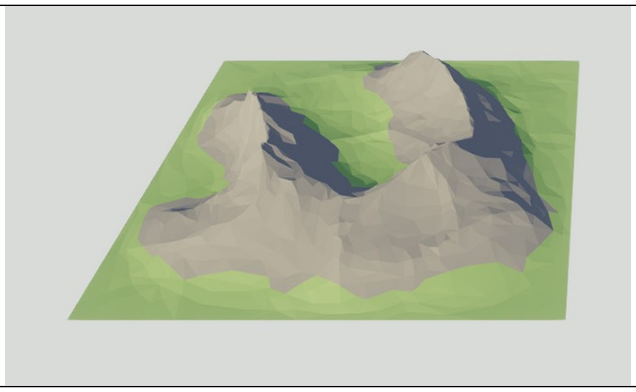
Go to *Low Poly Modular Terrain Pack/Terrain_Assets/Prefabs*

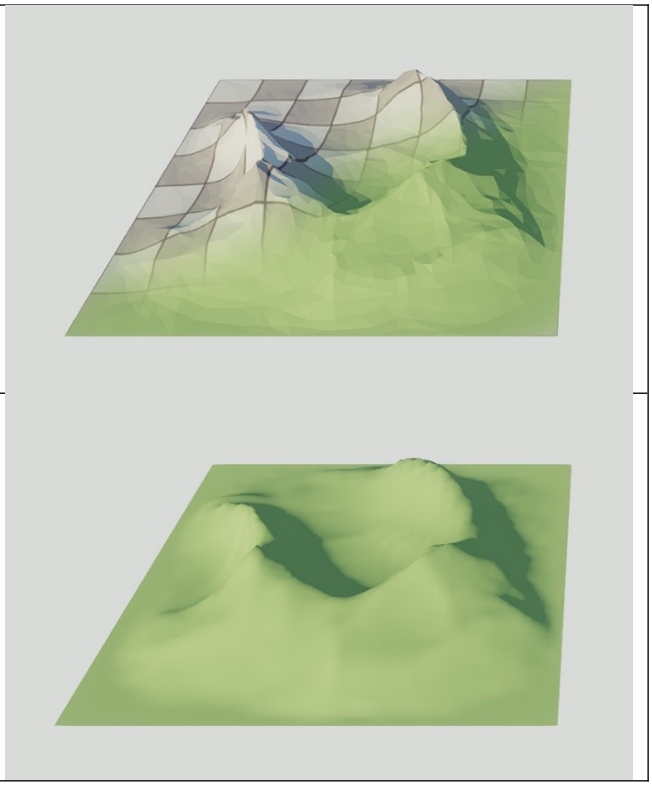
Choose which **Prefab** type you want to import to your scene:



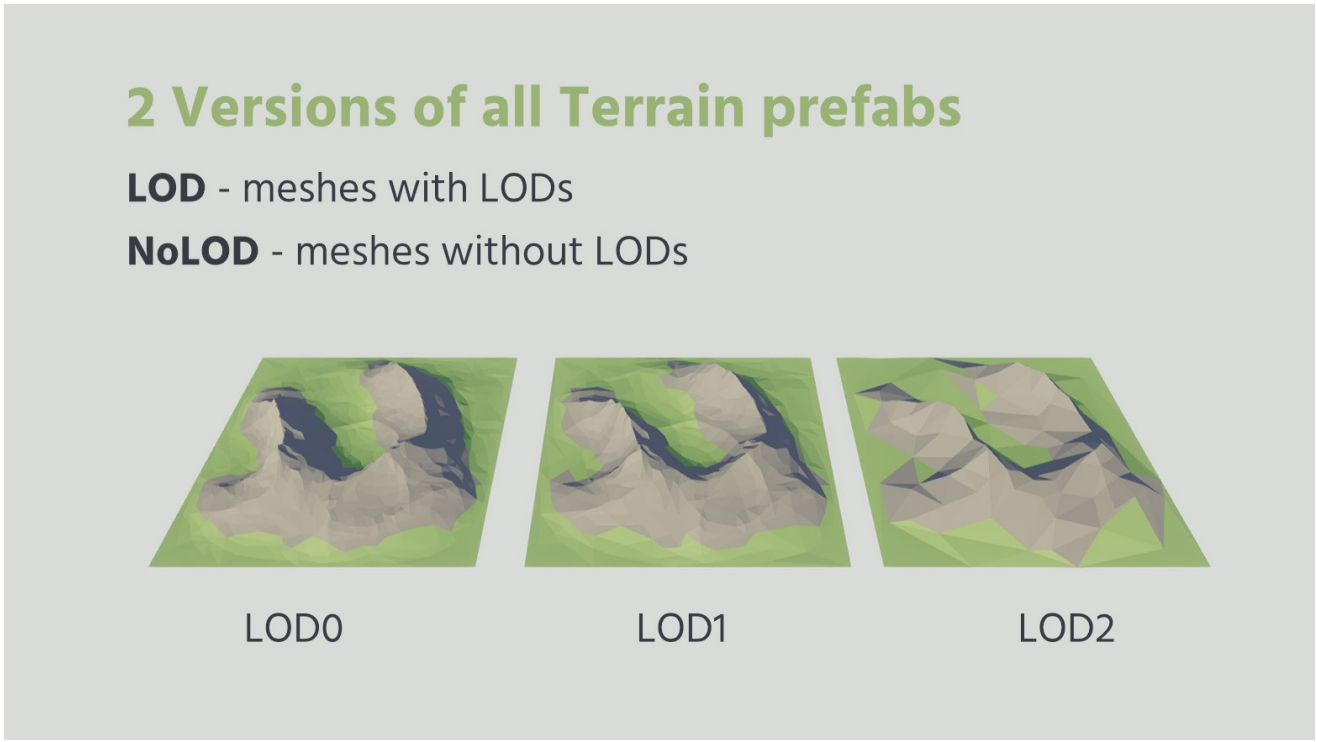
For example, open folder **Terrain**. You will see that you can choose between **3** types of Terrain:



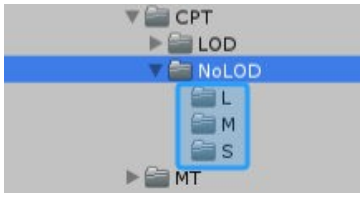
CPT – Color Palete Texture. All CPT prefabs use 1 material + 1 color palete texture atlas 64x64.	

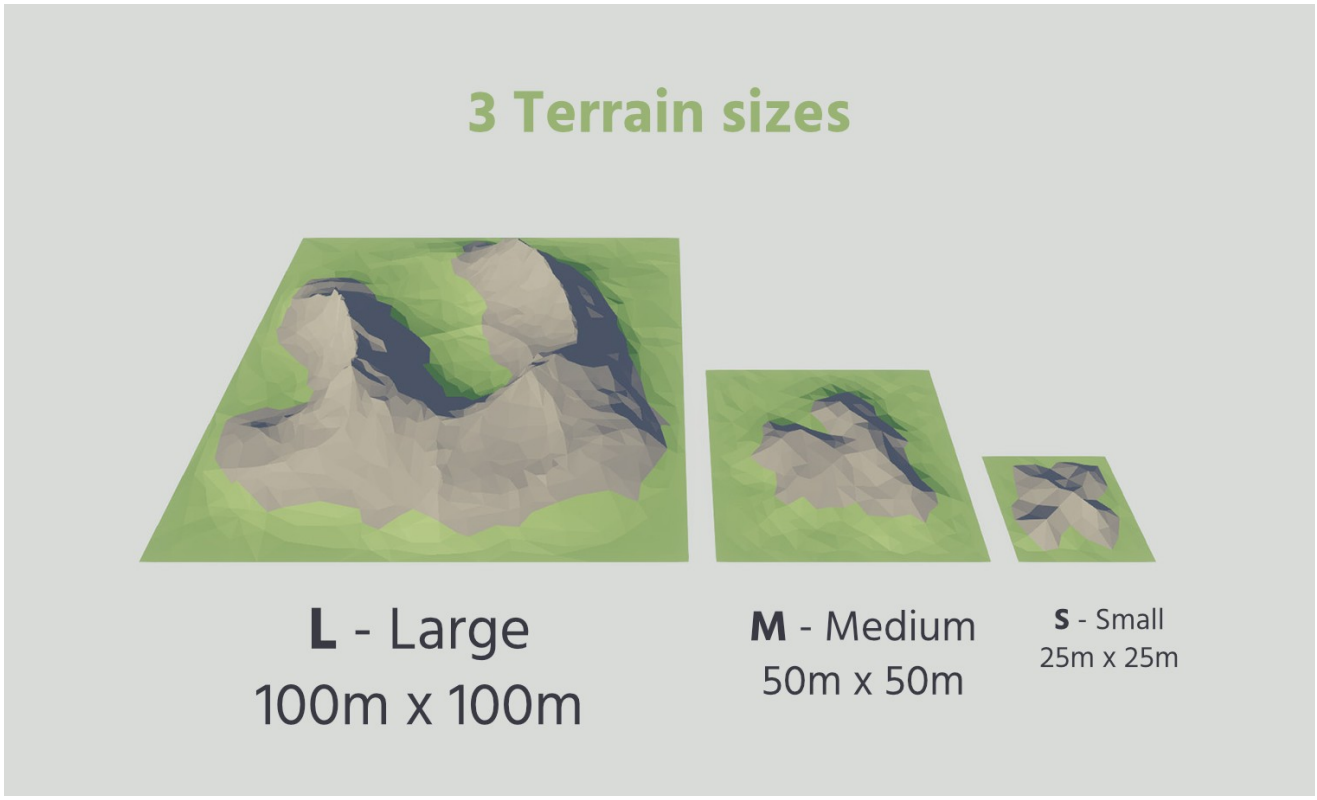
MT – Material & Texture. All MT prefabs use 1 material. You can also apply a seamless texture.	
U – Unity Terrain. You can edit the terrain shape, paint textures, draw grass, trees, etc.	

Open folder **CPT**. You can choose between **LOD** and **NoLOD**:



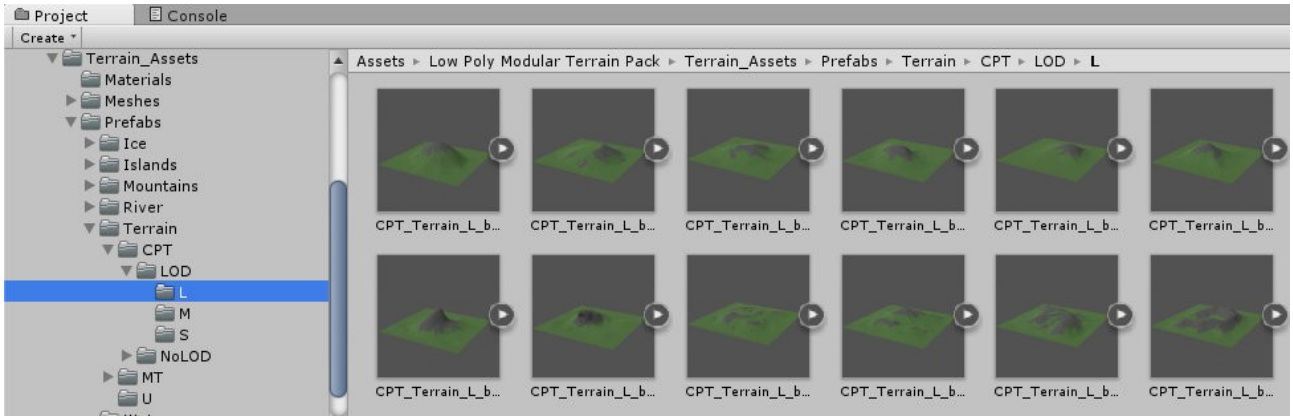
Open folder **NoLOD**. You will see that you can choose between **L**, **M**, and **S**:



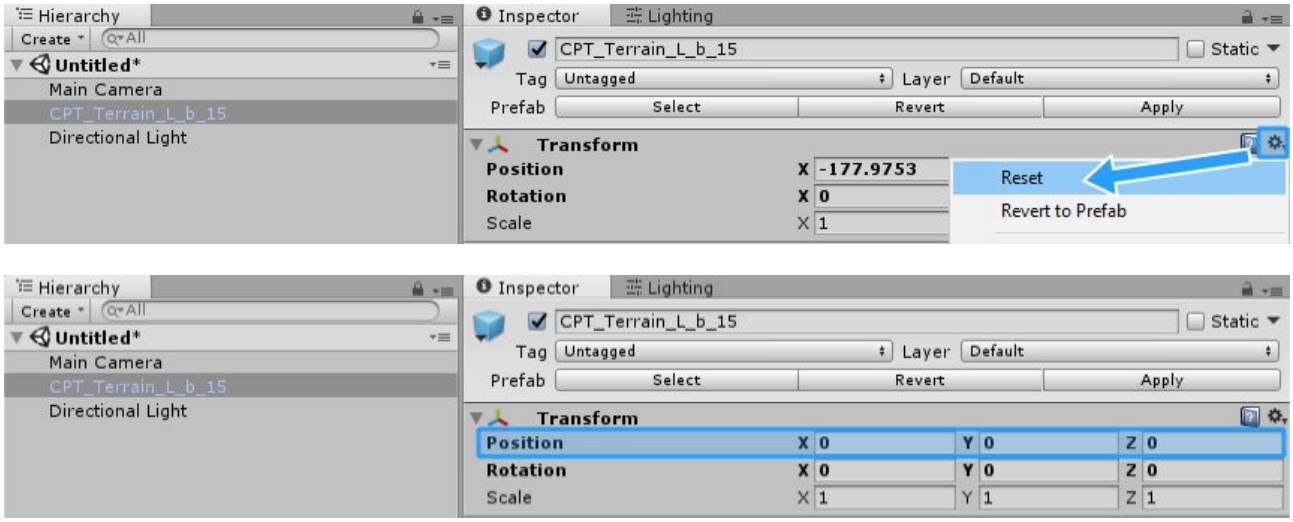


***L** – Large terrain is 100x100 meters (100x100 Unity units).

Let's open folder **L**, select and drag **Prefab** to your scene.



With a prefab selected in the Scene, **Reset Transform** (position to 0,0,0) so the Terrain will sit on the grid perfectly.



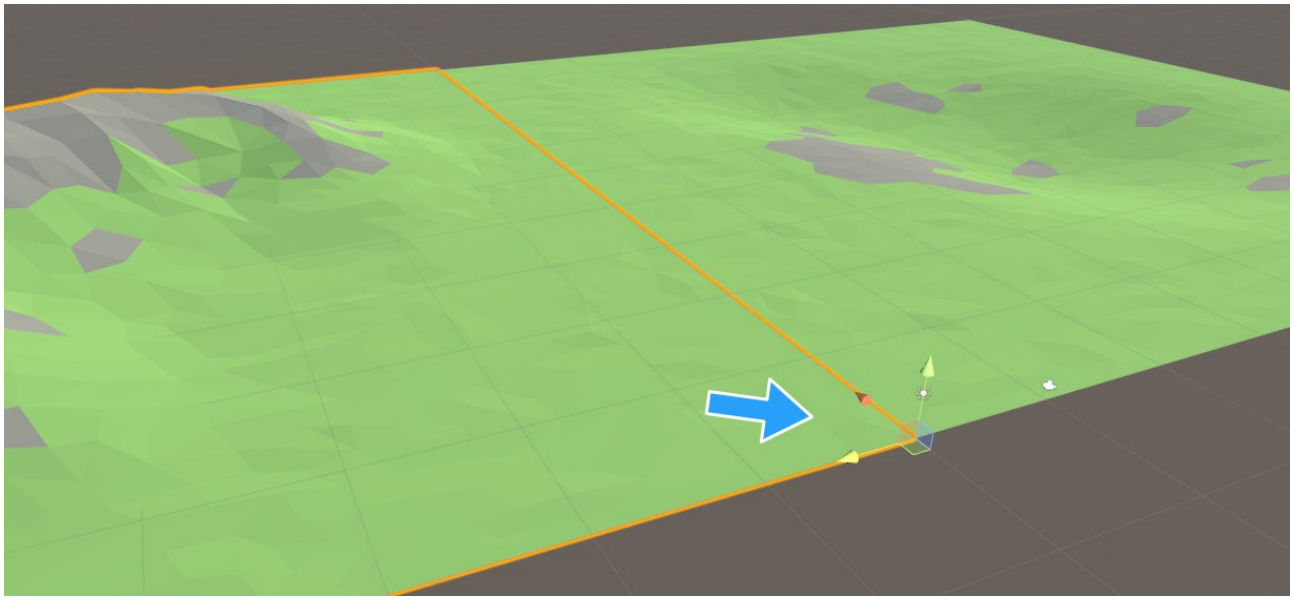
*I recommend drag and drop Prefabs straight to the **Hierarchy** tab. That way, you don't need to **Reset Transform** because it should be at **Position 0,0,0**.

Every model pivot is at the bottom corner of the model, so you can quickly drop Terrain to the Scene, duplicate or add another one, move and snap to the grid.

*Use **Pivot** and **Global** settings for the best experience!

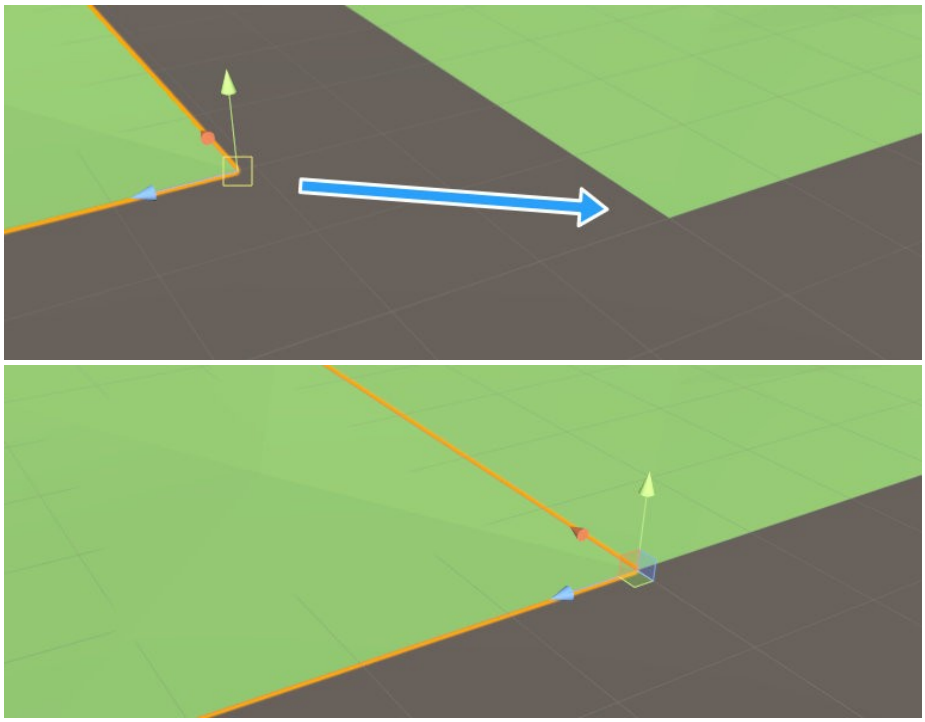
You can change it by tapping on the **buttons**, which are near Move, Scale tools.

Hold CTRL + **Grab and Drag** the Transform arrow to snap Terrain to the Grid while moving it.



*L – Large terrain is **100x100** units size, so you can easily change Position by **100** units to tile Terrain planes perfectly together.

You can also snap Terrain planes by using **V**. **Hold V** and hover the mouse cursor on the Terrain corner, and you will see a little **Yellow Square** (it shows which vertex of the mesh is selected for snapping). By **Holding V**, press and **Hold Left Mouse Button** and move it to the other Terrain corner to snap.



*I recommend using Polybrush for texture painting on **MT Terrain**. You can also draw prefabs (rocks, trees, grass, etc.) on any mesh terrain using Polybrush!

To use Bonus Assets

Go to *Low Poly Modular Terrain Pack/Bonus_Assets/Prefabs*

Select prefab you want and drag it to the scene.

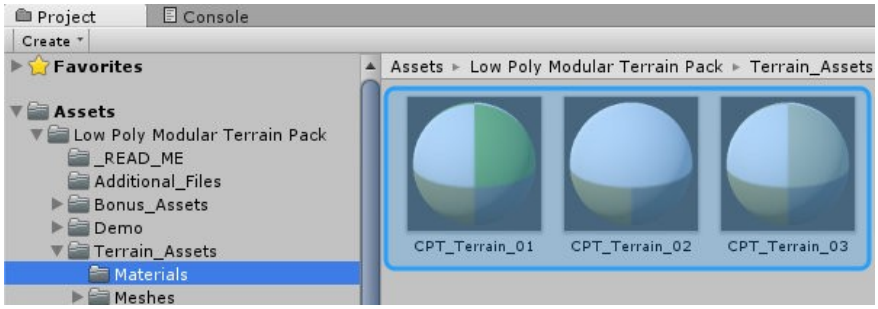
How to Change Prefab Color / Texture

CPT Terrain, Mountains, Islands, River

CPT – Color Palette Texture. All CPT prefabs use 1 material + 1 color palette texture atlas 64x64.

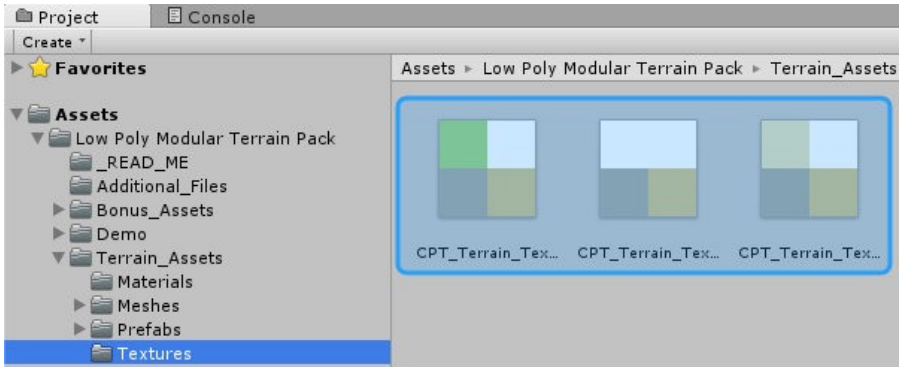
Watch the VIDEO TUTORIAL at **05:17** or follow the steps below.

Go to *Low Poly Modular Terrain Pack/Terrain_Assets/Materials* - here, you will find **3 Materials** that are used for all **CPT** prefabs (CPT Terrain, Mountains, Islands, and River).



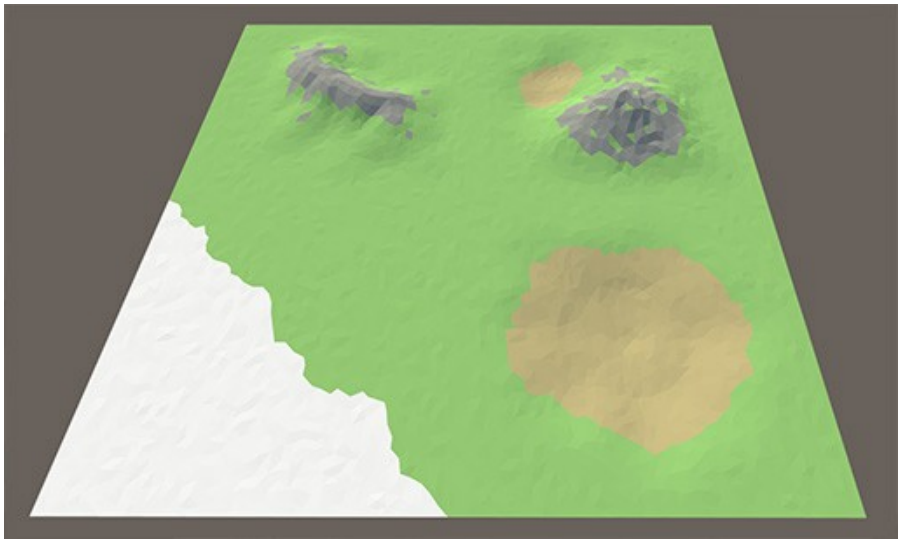
Material **CPT_Terrain_01** is applied to all **CPT** prefabs. To change the colors of CPT Prefabs, you can apply one of 3 CPT materials, or you can edit the Texture colors.

Go to *Low Poly Modular Terrain Pack/Terrain_Assets/Textures* - here, you will find **3 Textures**.



CPT_Terrain_Texture_Atlas_01.png is applied to **CPT_Terrain_01** Material. You can open this texture inside any image editing software and change the colors. There are only 4 colors on the texture:

These Colors are used like this:



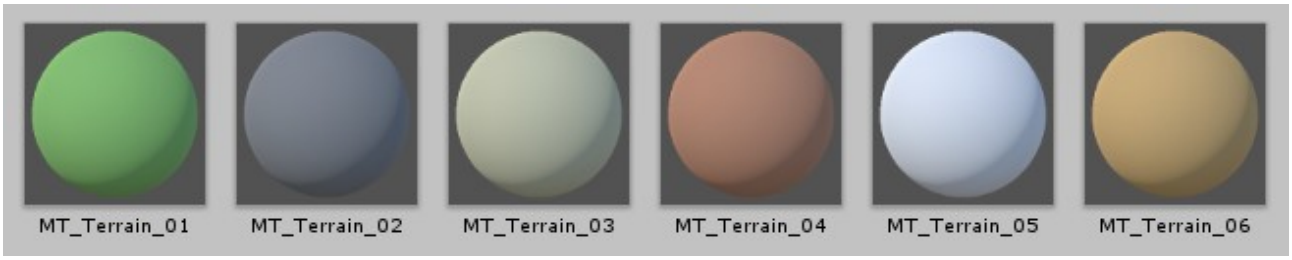
There are 3 high-resolution **.PSD** textures included to edit it more easily. Go to *Low Poly Modular Terrain Pack/Additional_Files* - here, you will find the **CPT_ Textures.rar** file. Extract it, open any of 3 textures inside Photoshop, Affinity, or any other image editing software. Then save at a small resolution like 64x64 and import to your Unity project.

MT Terrain, Mountains, Islands, River

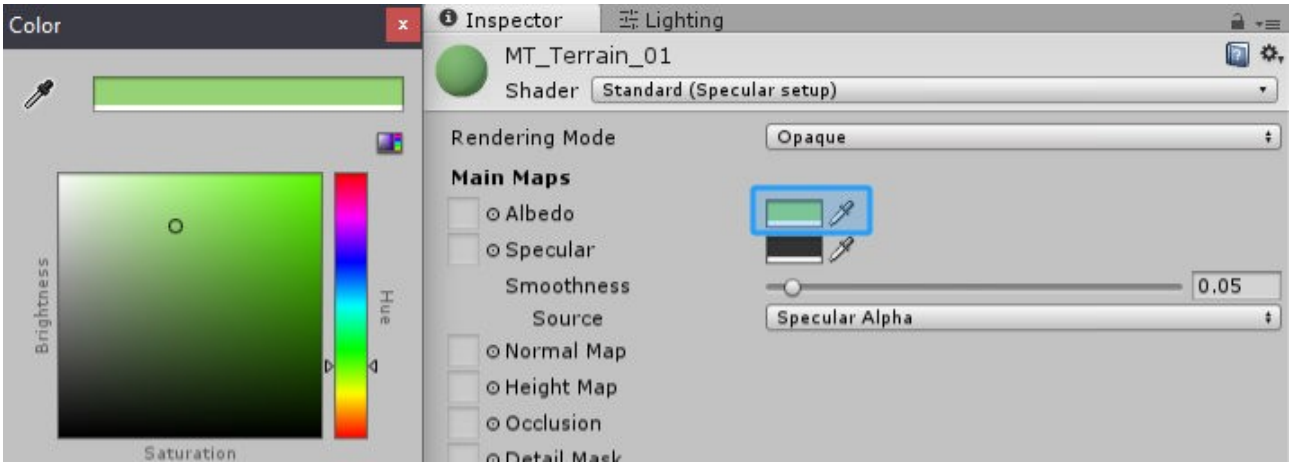
MT – Material & Texture. All MT prefabs use 1 material. You can also apply a seamless texture.

Watch the VIDEO TUTORIAL at **11:35** or follow the steps below.

Go to *Low Poly Modular Terrain Pack/Terrain_Assets/Materials* - here you will find **6** materials which are used for **MT** Prefabs:



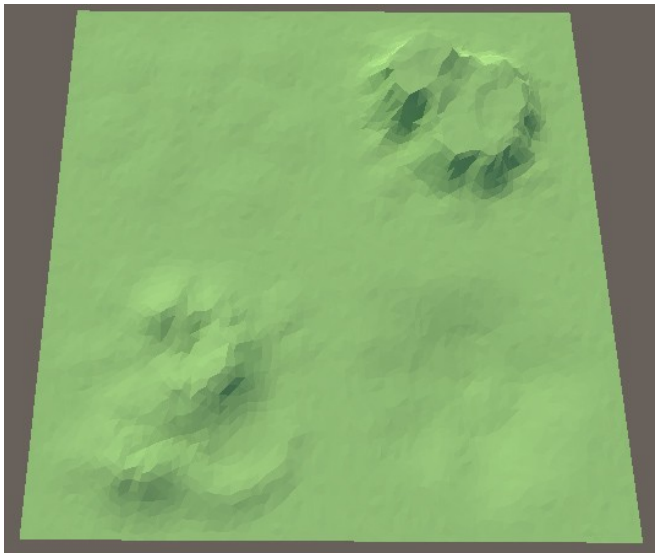
MT_Terrain_01 is applied to almost all **MT** Prefabs by default. To change the color of **MT Prefabs**, apply any of these materials, select material and change **Albedo** color:



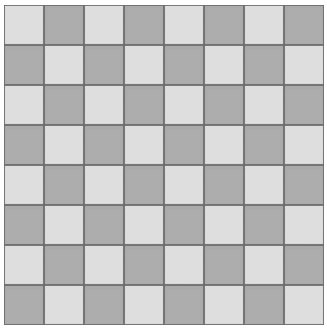
MT Terrain Texture

Watch the VIDEO TUTORIAL at **12:40** or follow the steps below.

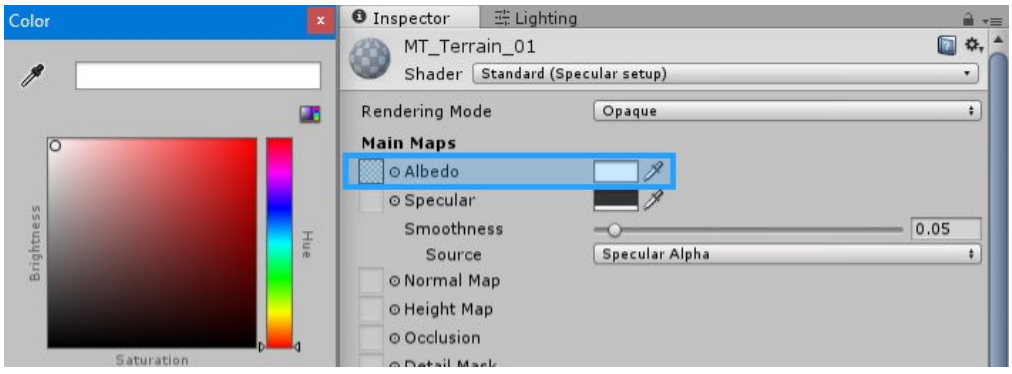
You can apply any tileable **Texture** to all **MT** Prefabs. Here are **4 MT_Terrain** Prefabs added to the Scene (Terrain use default **MT_Terrain_01** material):



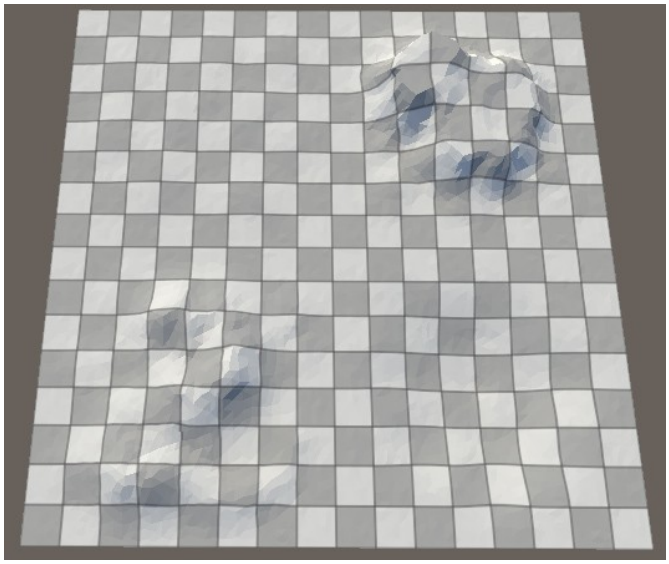
I will apply a simple Grid Texture to the **MT_Terrain_01** Material (**Albedo** slot):



And change material **Albedo** color to **White**:



Now Terrain looks like this:



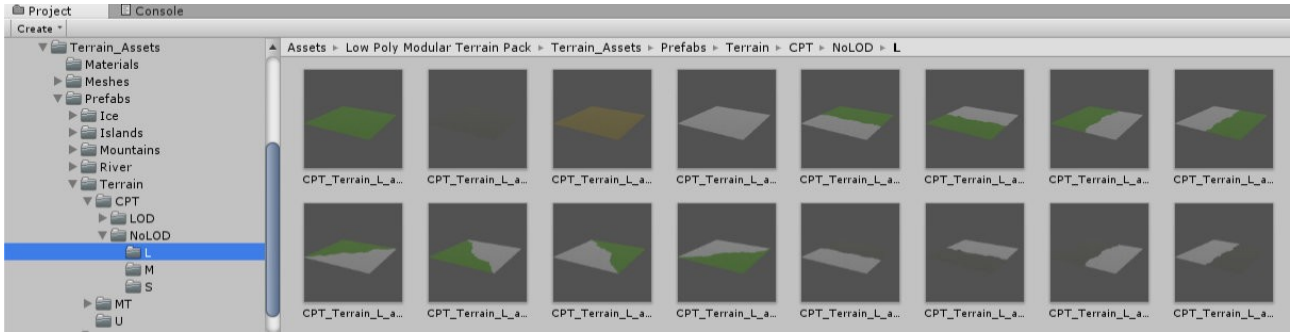
U Terrain

U – Unity Terrain. You can edit the terrain shape, paint textures, draw grass, trees, etc.

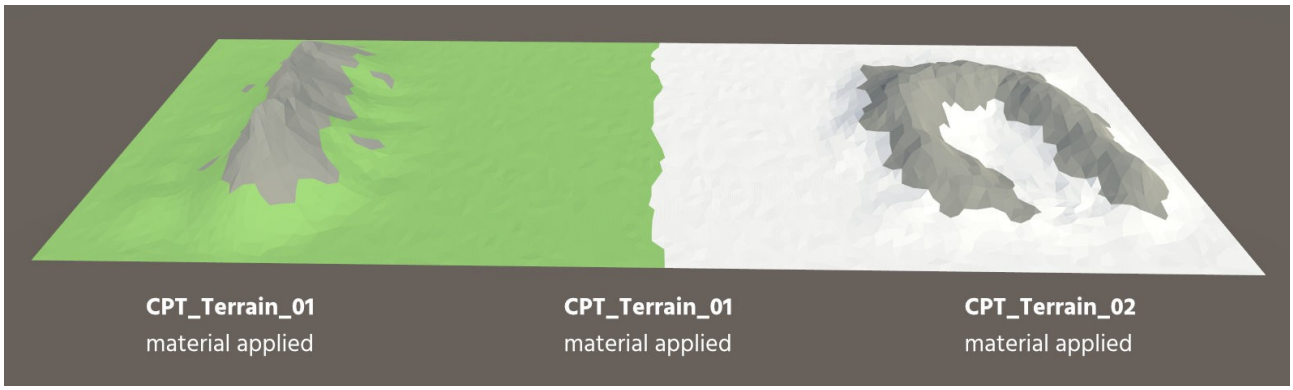
Watch the VIDEO TUTORIAL on how to use it at **13:45**

How to Use CPT Terrain Transition Parts

If you go to *Low Poly Modular Terrain Pack/Terrain_Assets/Prefabs/Terrain/CPT/NoLOD/L* – you will see there are a bunch of terrain transition parts. They can be used to make a transition from a grass to a snow terrain, mud to grass, and so on.

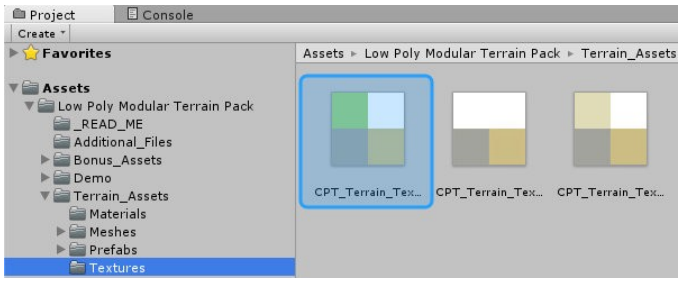


Here is an example of 3 MT Terrain prefabs in the scene. Terrain prefab on the left, transition Terrain in the center, and another Terrain on the right.

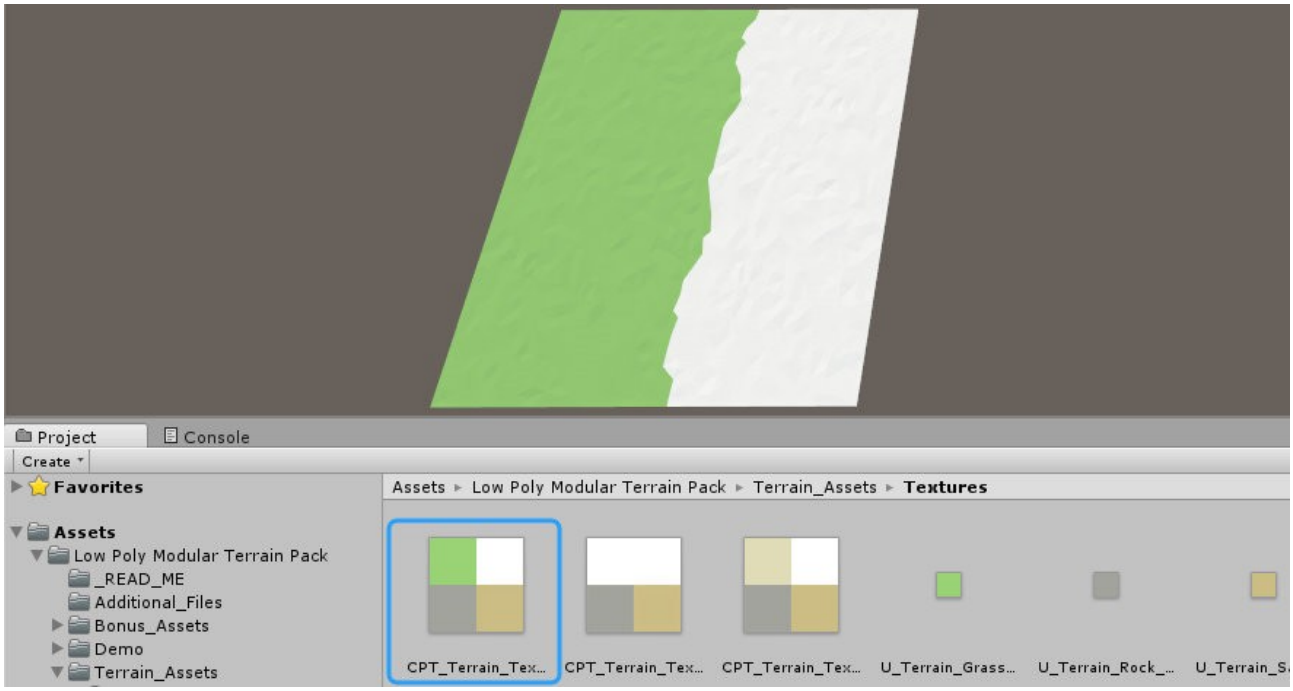


Change Transition Terrain Color

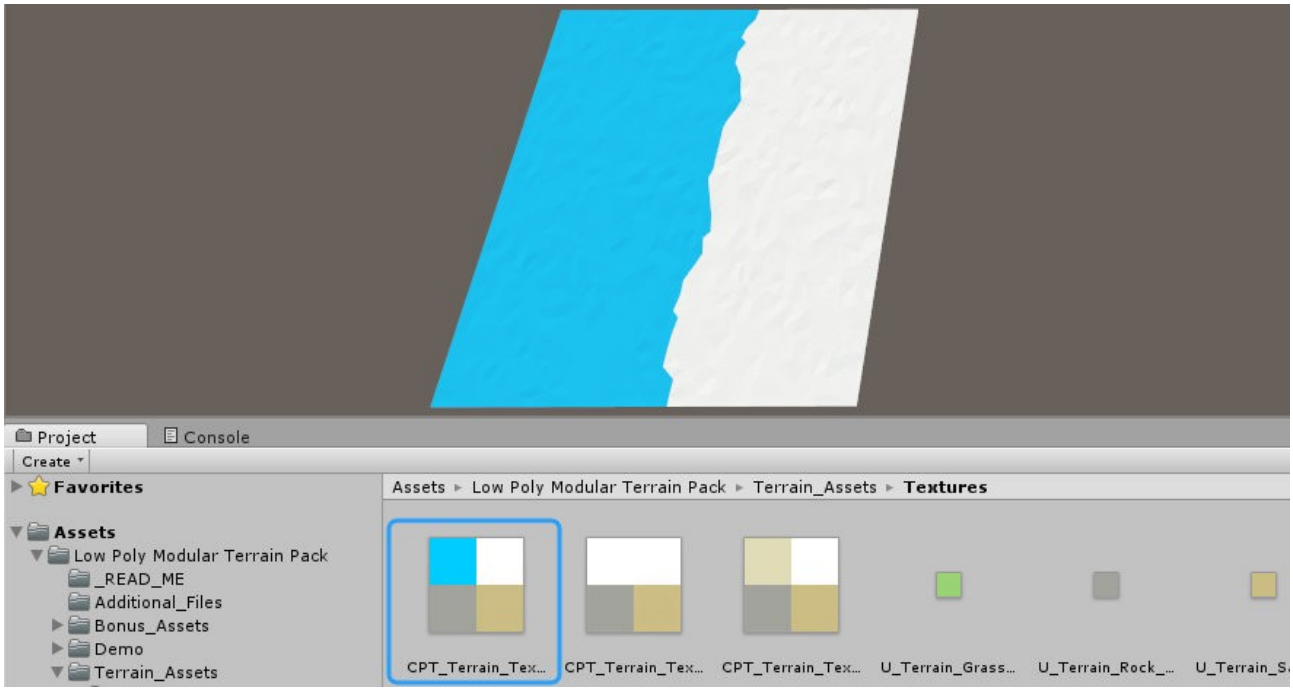
Go to *Low Poly Modular Terrain Pack/Terrain_AssetsTextures* – **CPT_Terrain_Texture_Atlas_01.png** is used for all Terrain transition parts by default.



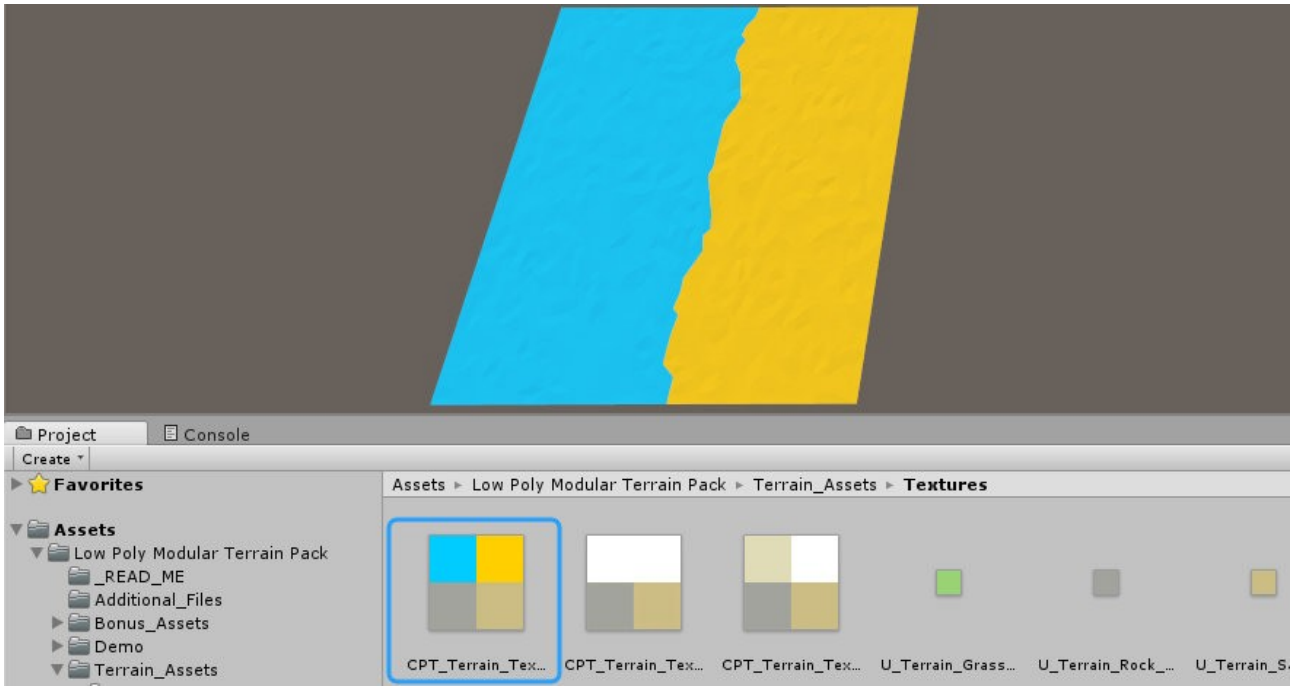
So open **CPT_Terrain_Texture_Atlas_01.png** in any image editing software and change the color you want. Here is the original color:



Changed the first color block to **Blue**:



Changed the second color block to **Yellow**:

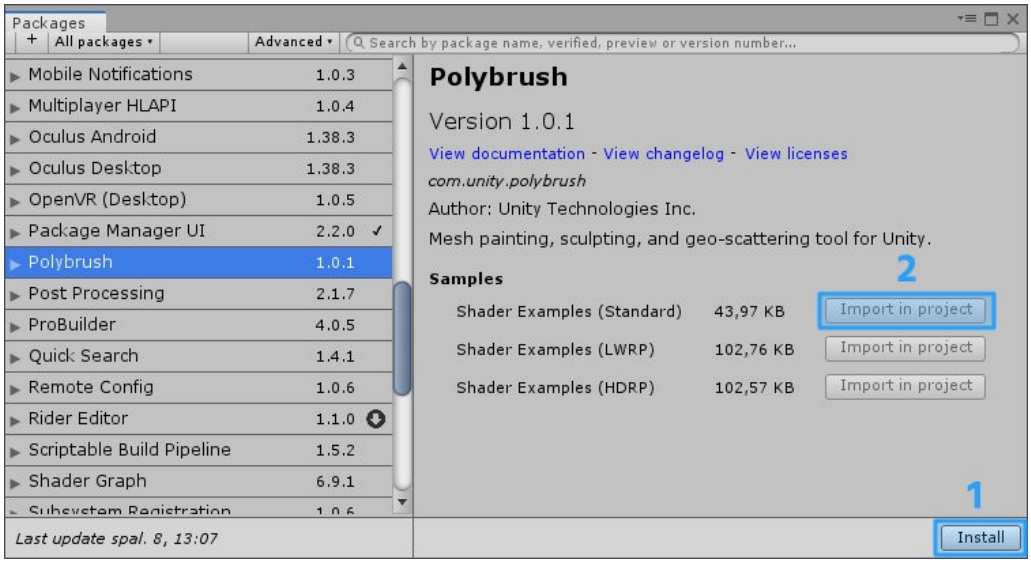


How to Paint Vertex Color And Textures on MT Terrain Using Polybrush

UPDATE! Watch my new Polybrush Tutorial on how to use it properly (including texture painting, prefab painting on the terrain, and mesh/MT_Terrain sculpting)!

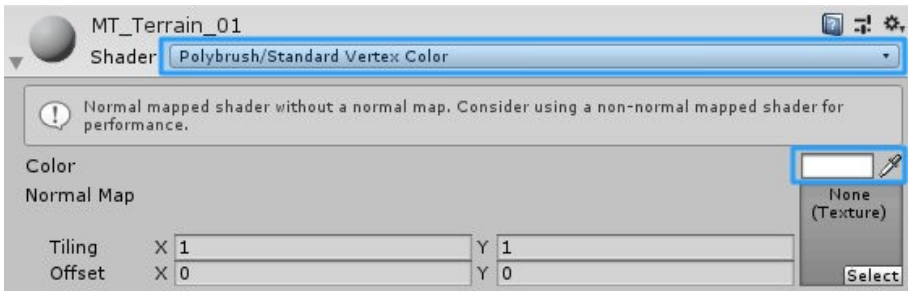
1. Import **Polybrush**.

If you are using **Unity 2019.1** and above, you can install **Polybrush** from the **Package Manager**! *Window > Package Manager* – Polybrush. Click on **Install**, after installing it, click on **Import in project** (to import example shaders we need to paint on terrain mesh).



2. Setup **MT Terrain** for Painting **Vertex Color**

Go to *Low Poly Modular Terrain Pack/Terrain_Assets/Prefabs/Terrain/MT/NoLOD/L* – and import any **MT terrain** to the scene. Select **MT Terrain** from your scene and change material Shader to **Polybrush/Standard Vertex Color** and set **Color** to **White**:

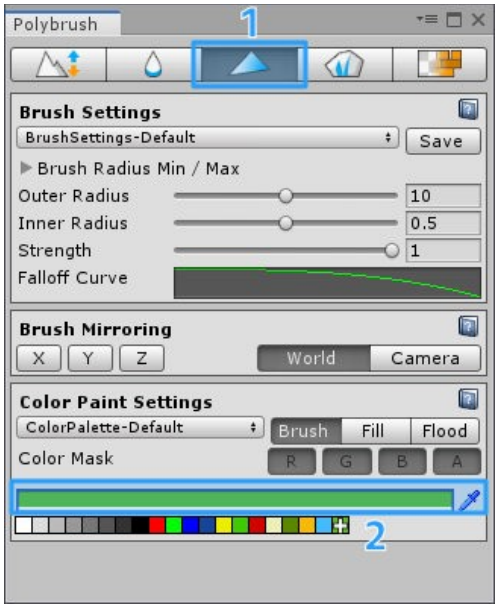


3. Setup and use **Polybrush** to paint **Vertex Color** on any **MT** prefab.

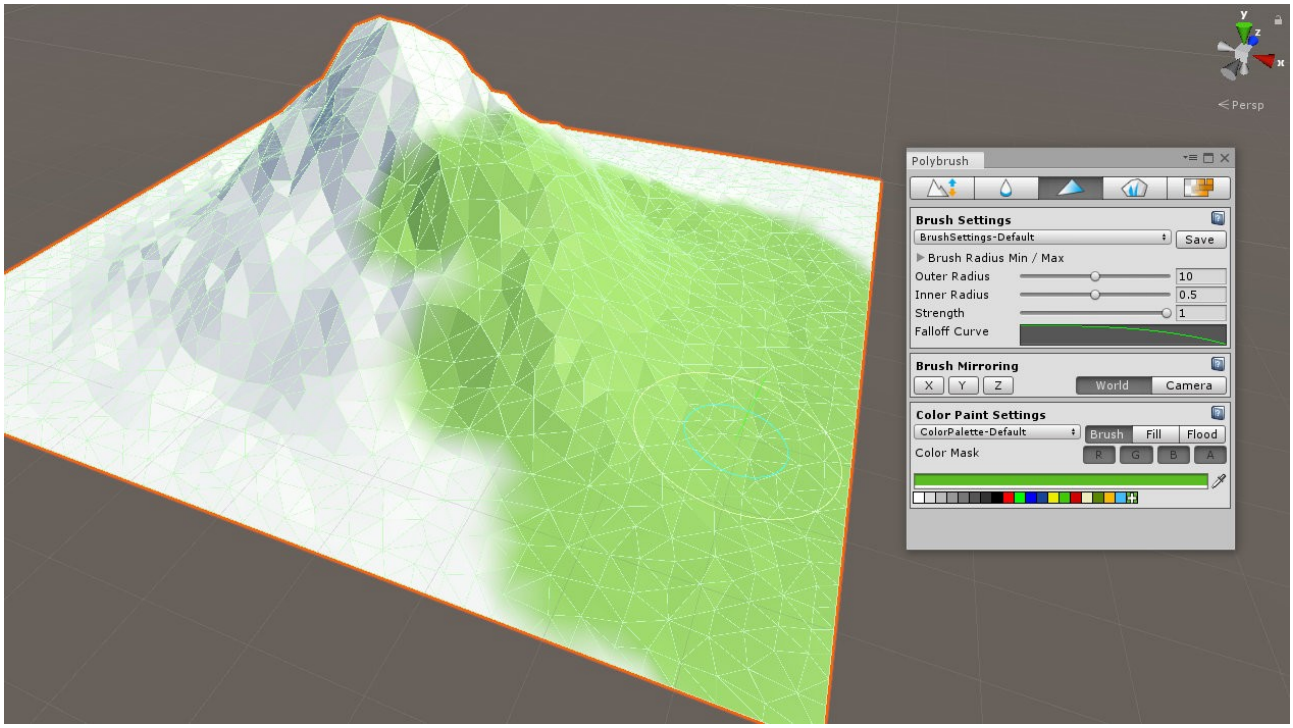
Go to *Tools > Polybrush > Polybrush Window*

You should see a **Polybrush** window.

1. Open **Paint vertex colors on meshes** tab
2. Choose a **Color** to paint

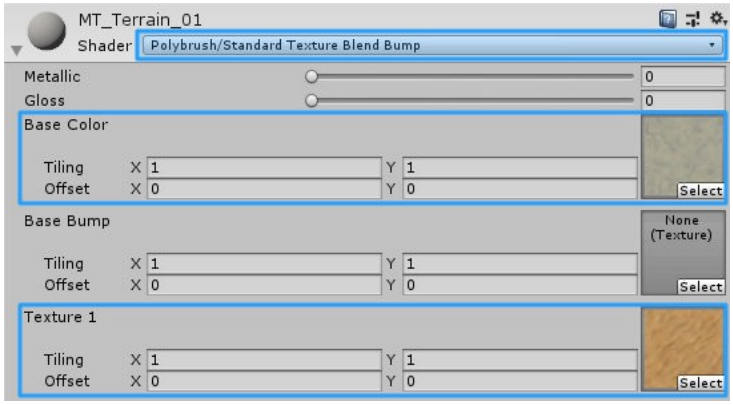


3. Select **MT Terrain** in the scene and **Paint**



4. Setup **MT Terrain** and use **Polybrush** to paint **Textures** on.

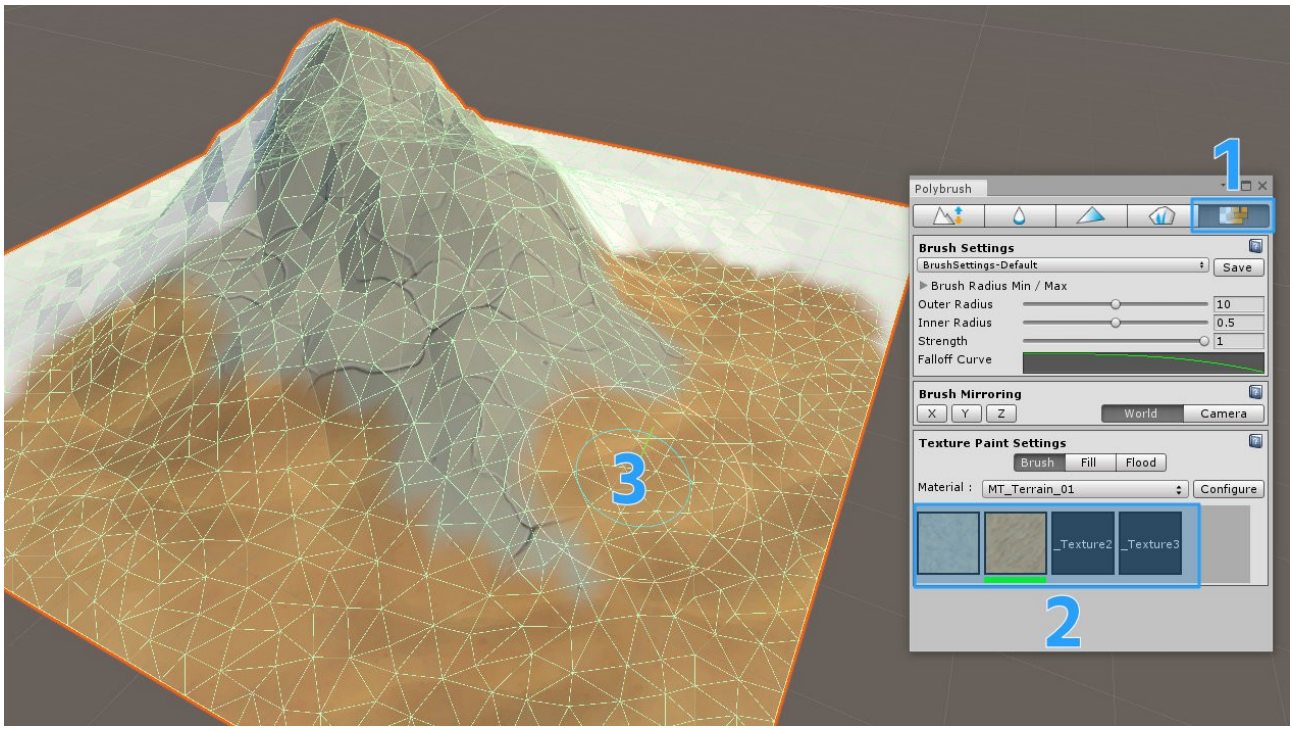
Select **MT Terrain** from your scene and change material Shader to **Polybrush/Standard Texture Blend Bump**. Apply **Textures** you want to paint as **Base Color**, **Texture 1**, **Texture 2**, etc.

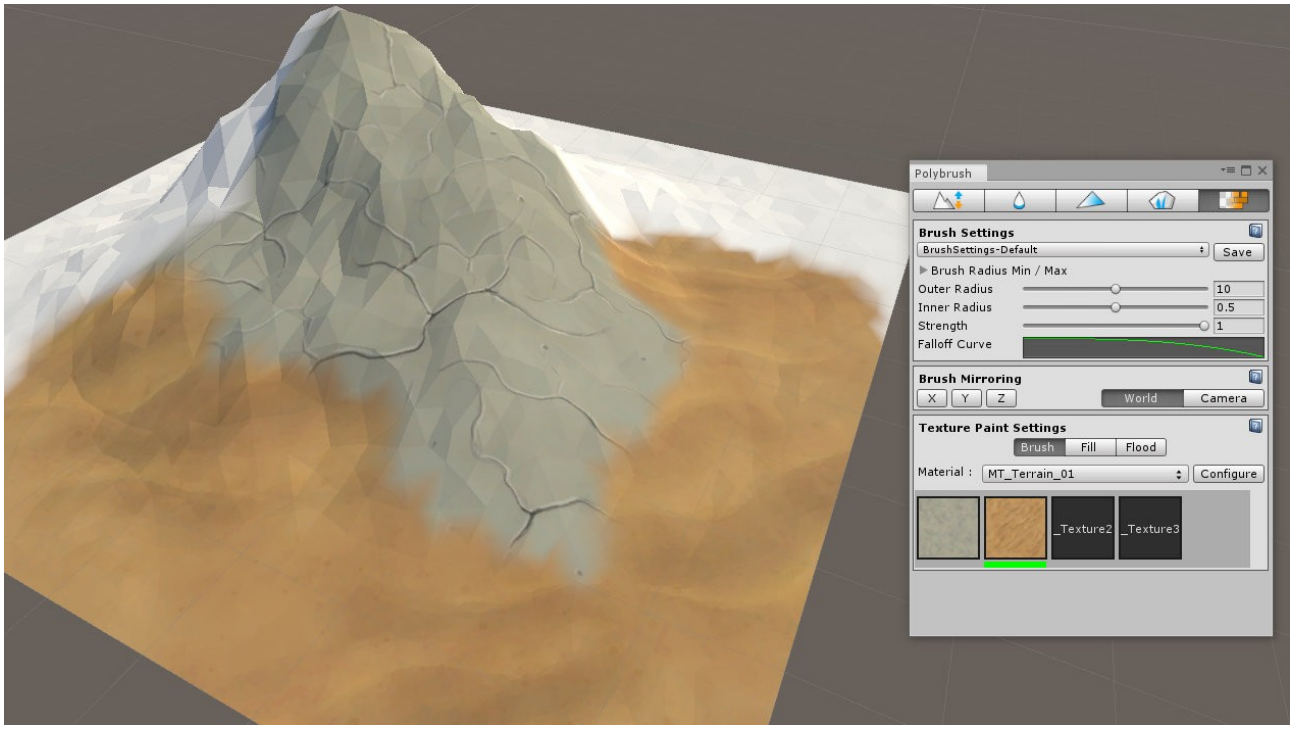


*Textures I've used in this demo are not included in the package!

Now inside **Polybrush**

1. Open **Paint textures on meshes** tab
2. Choose the **Texture** to paint
3. **Paint** on the **MT Terrain**




Polybrush
Brush Settings
BrushSettings-Default
Save
Brush Radius Min / Max
Outer Radius
10
Inner Radius
0.5
Strength
1
Falloff Curve
Brush Mirroring
X
Y
Z
World
Camera
Texture Paint Settings
Brush
Fill
Flood
Material :
MT_Terrain_01
Configure
_Texture2
_Texture3

Additional Info

Naming Conventions

Prefab name example 1: **CPT_Terrain_L_b_27**

- **CPT** – Color Palette Texture (All CPT Prefabs use 1 Material + 1 Color Palette Texture Atlas 64x64)
- **L** – Large 100m x 100m Terrain Size
- **b** – Just a Terrain type letter,
- **27** – Prefab number

Prefab name example 2: **Ice_H_BT_01**

- **H** – Huge size
- **BT** – With Bottom (ice has bottom faces and can be seen from both sides)

The same Terrain Prefabs comes in 3 different types:

- **CPT** – Color Palette Texture (All CPT Prefabs use 1 Material + 1 Color Palette Texture Atlas 64x64)
- **MT** – Material and Texture (All MT Prefabs use 1 Material. You can also add any seamless Texture to it!)
- **U** – Unity Terrain (You can edit the terrain shape, paint textures, draw grass, trees, etc.)

Almost all Prefabs come in 2 versions:

- **LOD** - Prefabs with 3 LOD levels: LOD0, LOD1, LOD2
- **NoLOD** –Mesh Prefabs without LODs

Prefabs come in 4 Sizes:

- **H** – Huge ~500x500m
- **L** - Large 100m x 100m Terrain Size
- **M** - Medium 50m x 50m Terrain Size
- **S** - Small – 25m x 25m Terrain Size

*Keep in mind that every terrain mesh is different, no matter is it small or large.

You can also find letter **R** at the end of the River part names. This means **Reversed**!

Scripts

Every scene **Camera**, **Directional Light,** and **_Clouds** (an empty game object which contains all clouds on the scene) have movement controls.

For, example, select **Camera** and on **Inspector** scroll down to the bottom, you will see **Modular Terrain Camera Control (Script)** attached to it. Here you can control **Camera Movement Speed** using sliders.

Same with **Direction Lights (Sun)** and **_Clouds**.

Contacts

If you have any questions, suggestions on what to improve or create. Maybe found any bugs, please send me an e-mail!

E-mail: justinas@lmhpoly.com

Website: https://lmhpoly.com/contact/

Follow me on **Twitter** to see what I'm working on right now:

https://twitter.com/lmhpoly

Don't miss out and be the first!

Get notified about the new "Low Poly Vegetation Pack" and other asset updates + my new game asset releases straight to your inbox.

Subscribe to LMHPOLY Game Asset Newsletter