

MBS

Assets Creation Tutorial

Roman Ponomarenko | roman.indiedev@gmail.com

questions? bugs? wishes? send me an email

1. Introduction
2. How does it work?
 - 2.1. Grid System
 - 2.2. Walls
 - 2.3. Floors
3. Wall assets creation
 - 3.1. Pivot Point (Origin)
 - 3.2. Face direction
 - 3.3. Corners coloring
 - 3.4. Windows and Doors Coloring
 - 3.5. Windows and Doors Spacing
4. Floor assets creation
 - 4.1. Pivot Point (Origin)
 - 4.2. Corner tile orientation
5. Import Assets
 - 5.1. Model import requirements
 - 5.2. Prefab layout
 - 5.3. Adding Assets to the MBS

Introduction

From developer:

I'm trying to reduce any routine work with the plugin and automatize as many processes as I can, but at the same time, I want to give you as much flexibility as it can be useful.

What does it mean? It means you can create any models you want, with any geometry, vertex count, texture, material, render pipeline, attach any scripts, and so on.

But at the same time, this means that all responsibility for the performance of your assets remains on you. Which materials will you use (PBR, tessellation, smoothness), which size of textures will you use, will you use texture atlas or separate texture for each object, will you combine meshes or not, and so on. You can set all these things as you want, the MBS does not touch it.

* Shortly there will be Mesh Combining built-in function but this is not implemented yet.

How does it work?

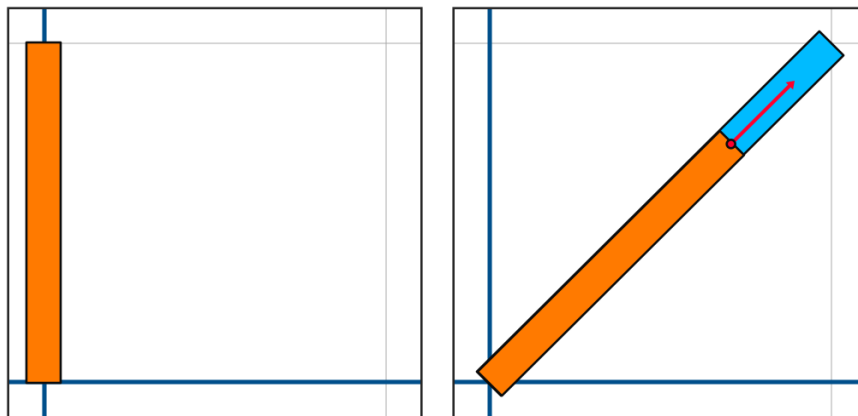
To design better assets it is good to know how the plugin works.

1. Grid System

Plugins work based on a grid system. A grid-based system is good enough for creating basic closed shapes out of modular parts. But it has its nuances.

- **First**, angle constraint. Walls can be rotated only with 45° steps.
- **Second**, when the wall is placed under 45/135 degrees it stretches up to the nearest diagonal point (0,0) -> (1,1). The amount of stretching equals this: $1 / \sin(45^\circ) = 1.4142^*$. It needs to be kept in mind.

NOTE: some textures can deteriorate a lot with this stretch, and in this case, it can be better to use triplanar materials instead of textures.

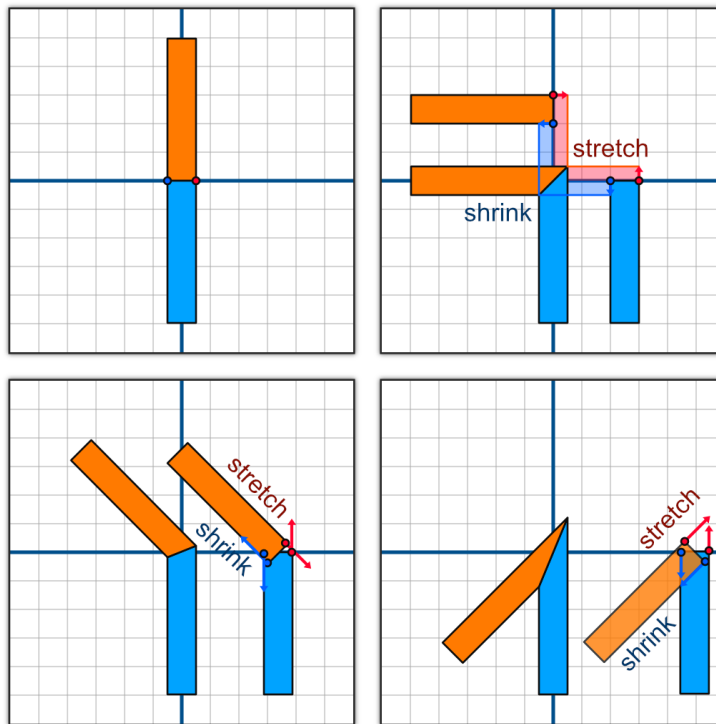


- **Third**, another stretching/shrinkage occurs when the walls are connected with a nonzero angle. In this case, stretching/shrinkage occurs not with the entire model, but with the corners that participate in the connection.

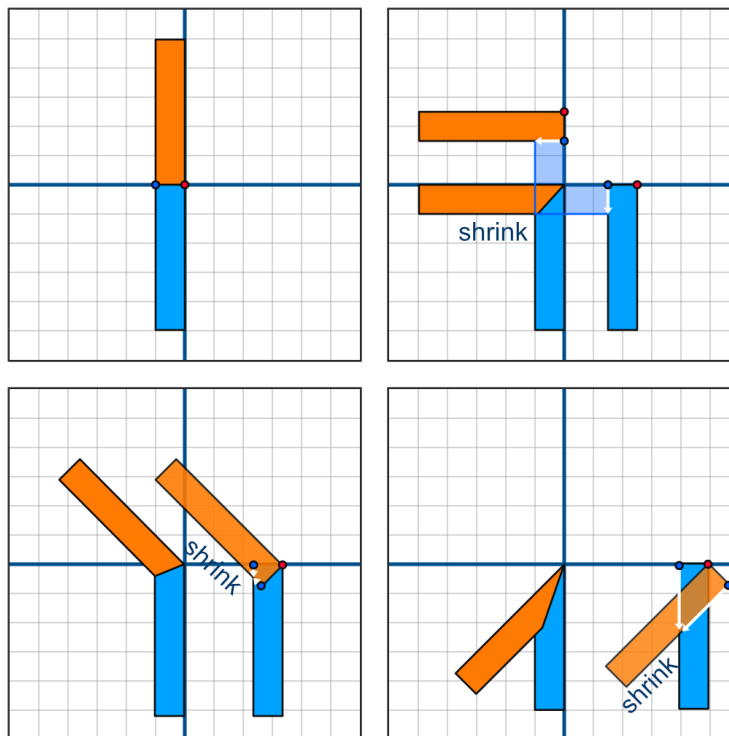
The amount of stretching/shrinkage depends on the offset of the vertex relative to the pivot point (origin).

NOTE: By default, the pivot point should be in the center of the bottom, but if you shift it you'll shift the ratio of stretching/shrinking of the wall sides.

Stretching/shrinkage with default pivot point.

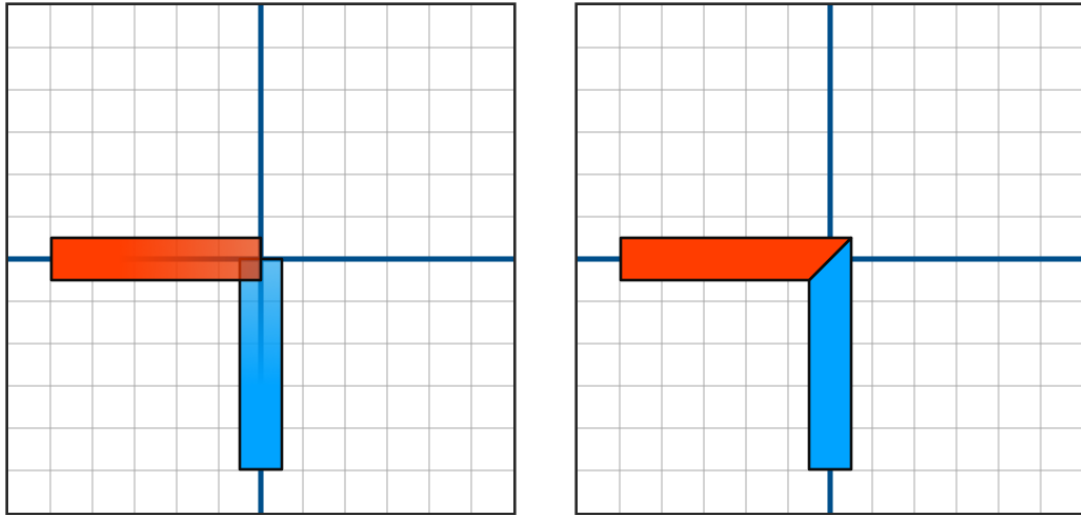


And with the shifted pivot point. In the shown case, the outer side of the wall will not be stretched at all, but the inner side of the wall will be shrunk 2 times more because the ratio was shifted with the shifting of the pivot point.

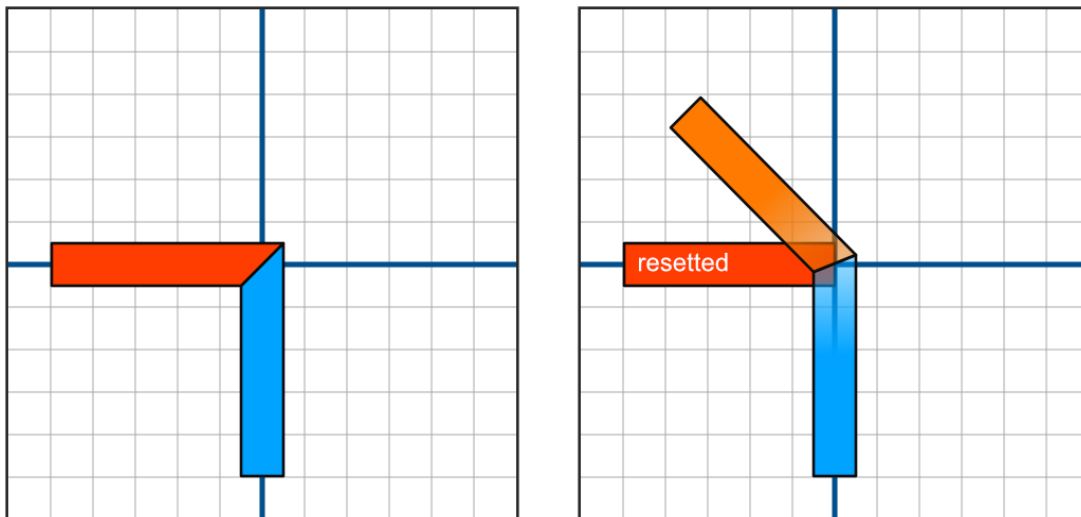


2. Walls

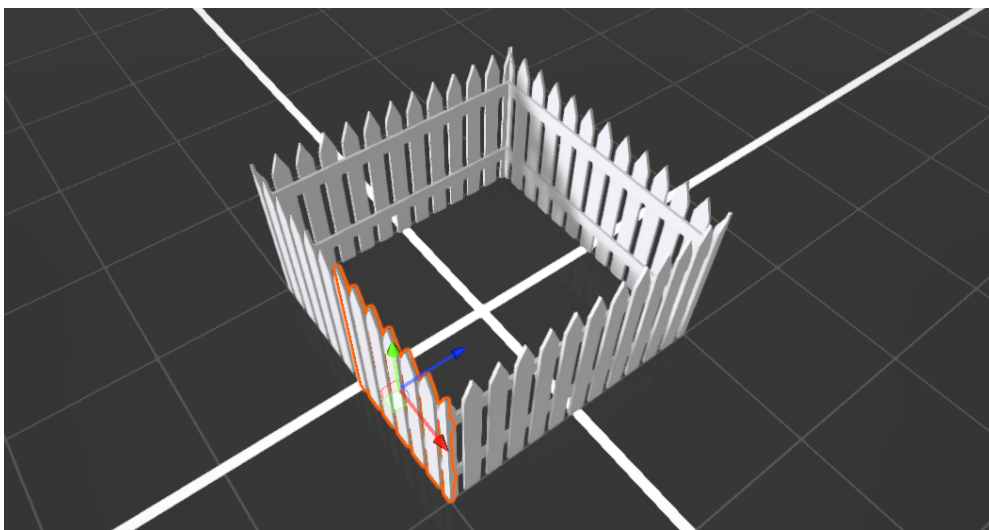
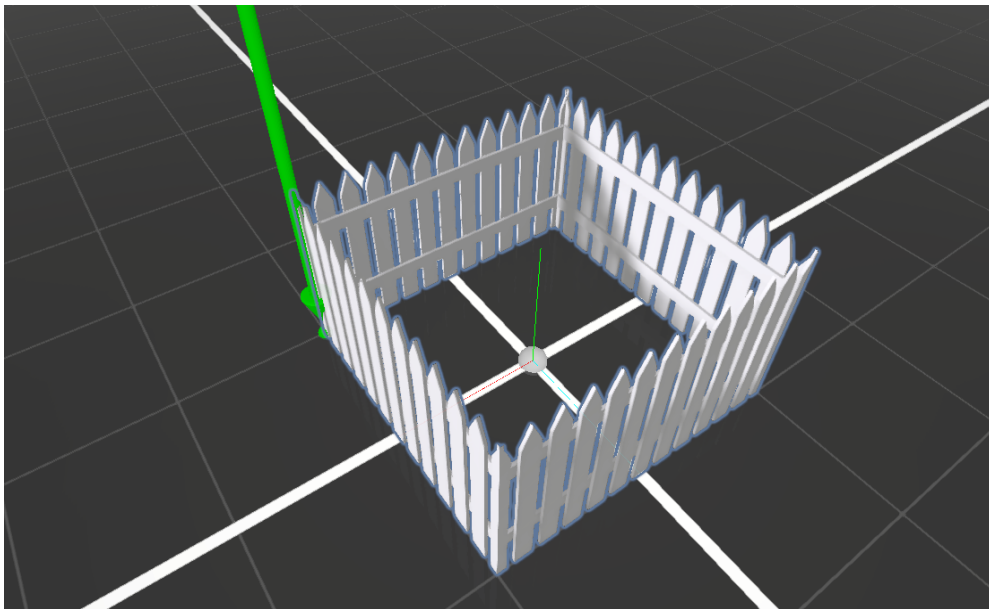
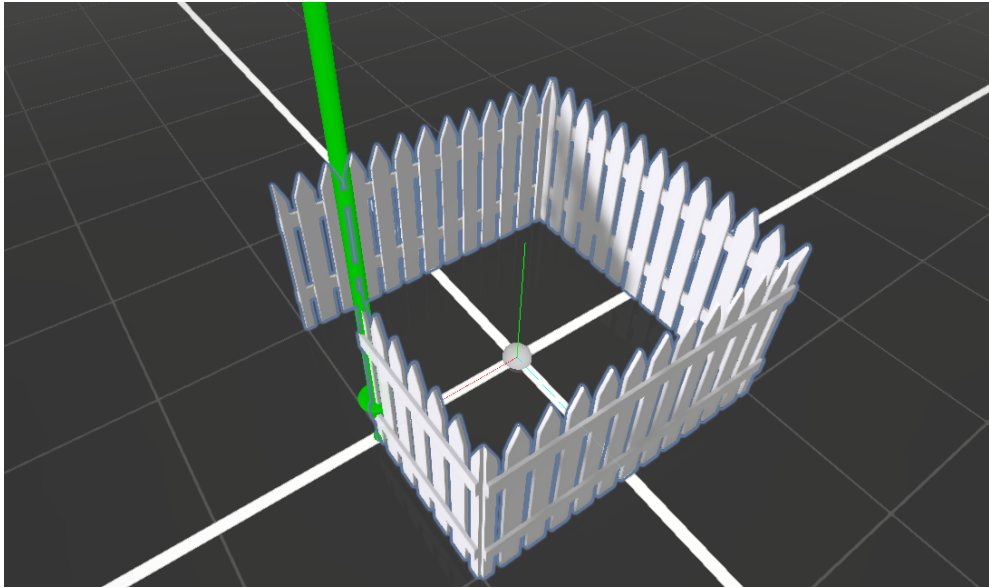
If walls are prepared correctly to work with MBS then at connection points (between two or more walls) they could be modified to fill the gaps and fix overlaps.



To reduce stretching/shrinking walls, MBS modifies only two walls in connection where an angle between is bigger, and the sides of the other walls in connection reset.



When walls are connected into the closed loop they will automatically change their facing inside the formed area. It means that their **forward direction / z-axis / blue arrow** will point **inside** the area they form.



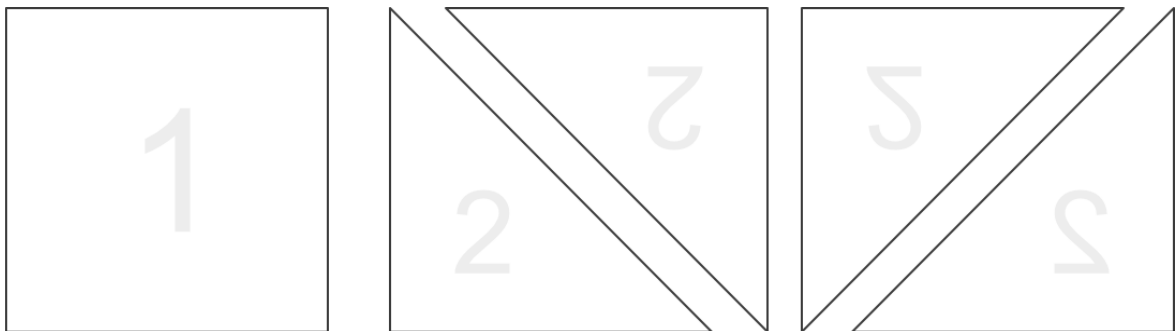
3. Floors

To place the floor tiles MBS is using two different models,

- one - square tile
- second - square cut in half = triangle = corner tile.

Instead of using procedural boolean cuts on square tiles, this method gives more flexibility (in terms of tiles geometry and texturing) and does not generate extra vertices or triangulation errors.

NOTE: to place **corner tile** in 4 different positions MBS does not rotate it, but “flip” it. This method allows to preserve the direction of the applied texture (for example, a wooden floor) but can cause duplication of the pattern at the seams.

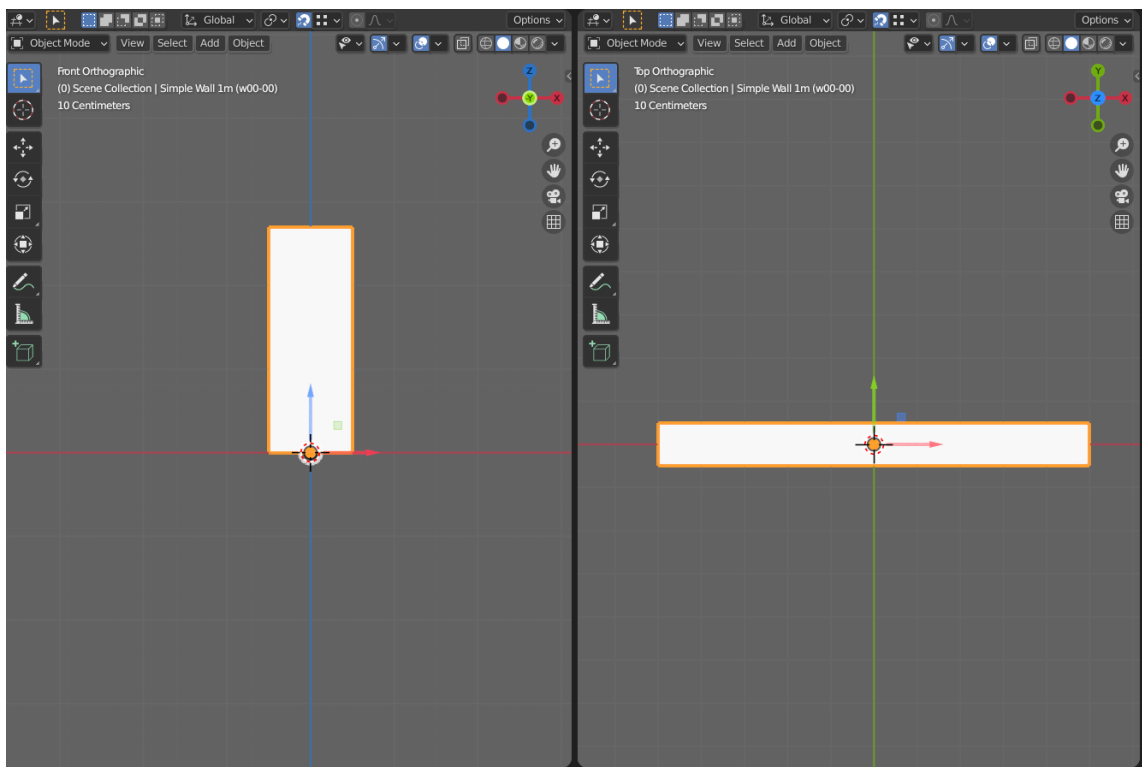


Wall assets creation

To prepare walls to work in MBS please follow next rules:

1. Pivot Point (Origin)

The **default** position of the pivot point (origin) is the **bottom center**. You can change it to achieve special results.



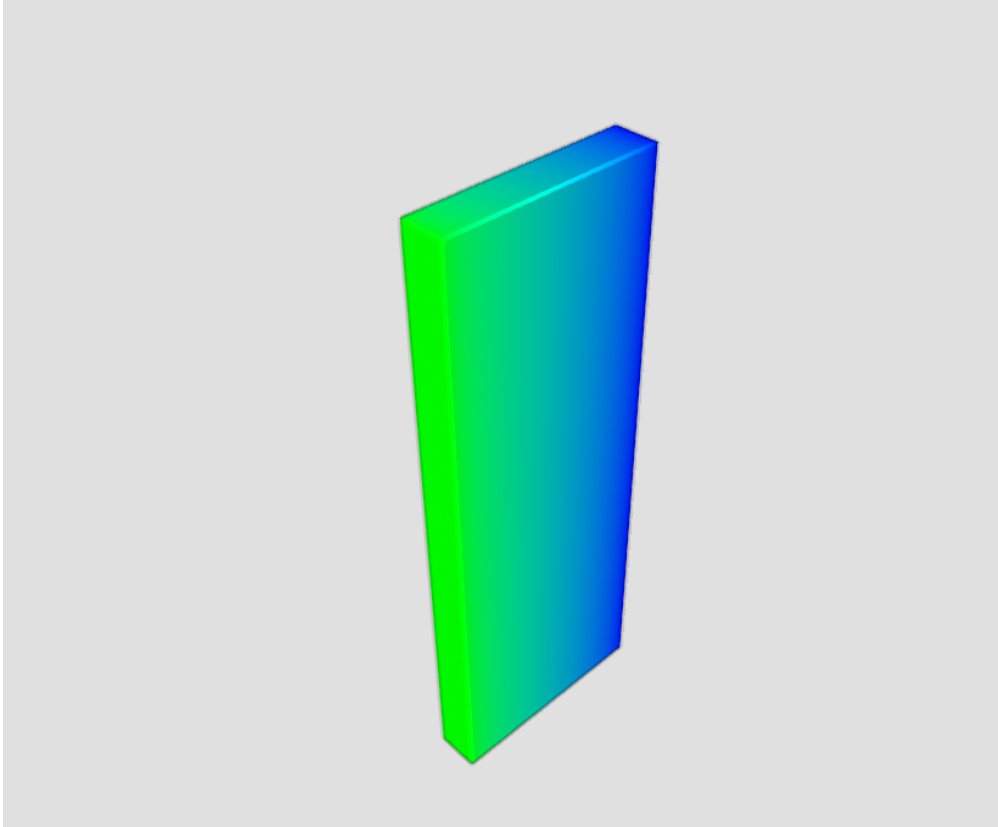
2. Face direction

If you're creating a model with different sides (outside-inside) make sure that the **inner side** is facing the **+Z axis** (-Y Blender).

3. Corners coloring

Vertices of each side of the wall model should be colored (**vertex color**) in any unique color except **black** (RGB (0,0,0)) and **white** (RGB (255,255,255)).

One side - one color.



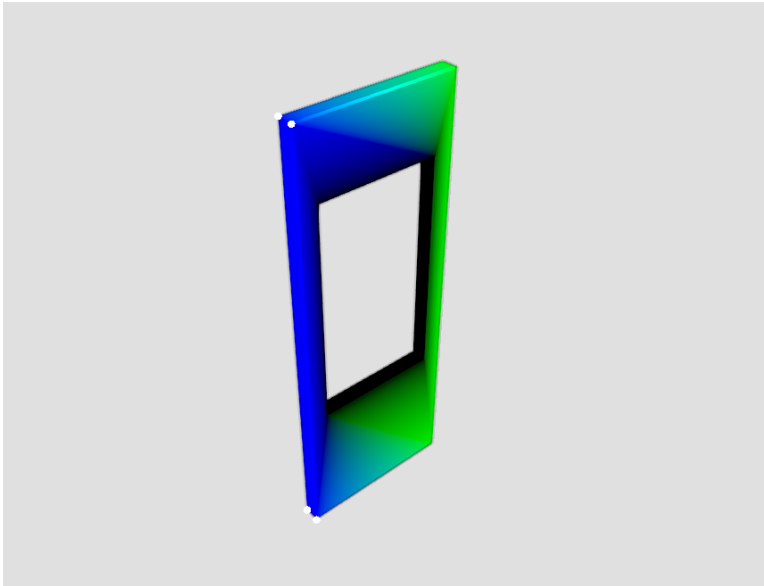
This is necessary to modify the corners of the mesh at the connection of two or more walls.

NOTE: make sure that vertices are colored in the same RGB color. If there is a difference even in one digit, for example, **RGB (255, 0, 0)** and **RGB (254, 0, 0)**, the MBS will treat this as different sides.

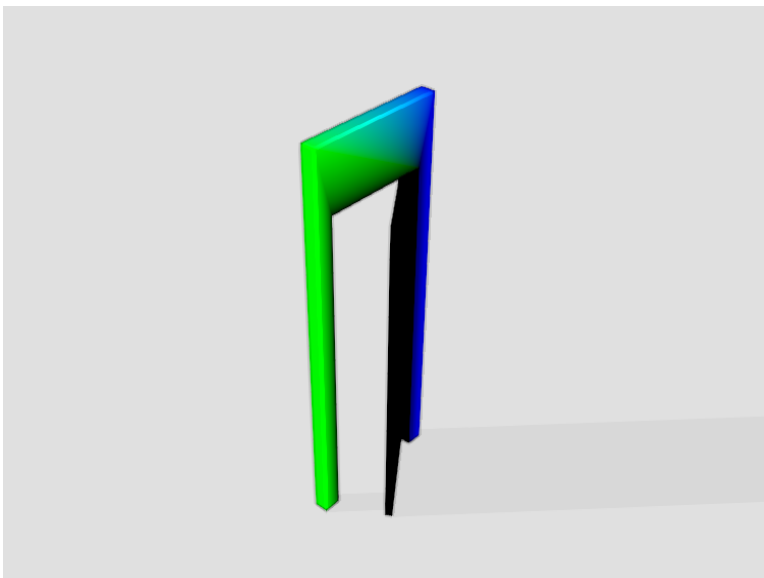
4. Windows and Doors Coloring

Since MBS stretches wall models that are placed under 45° (grid space) stretching doors, windows, and similar things can negatively affect the visual or functional part. Because of it, there is **“no effect color”** - **black** RGB(0, 0, 0).

If your model has parts that shouldn't be modified due to stretching or something else, color them in black.



If your model has children's models such as doors, windows, or any interactive parts and so on, color them too.



5. Windows and Doors Spacing

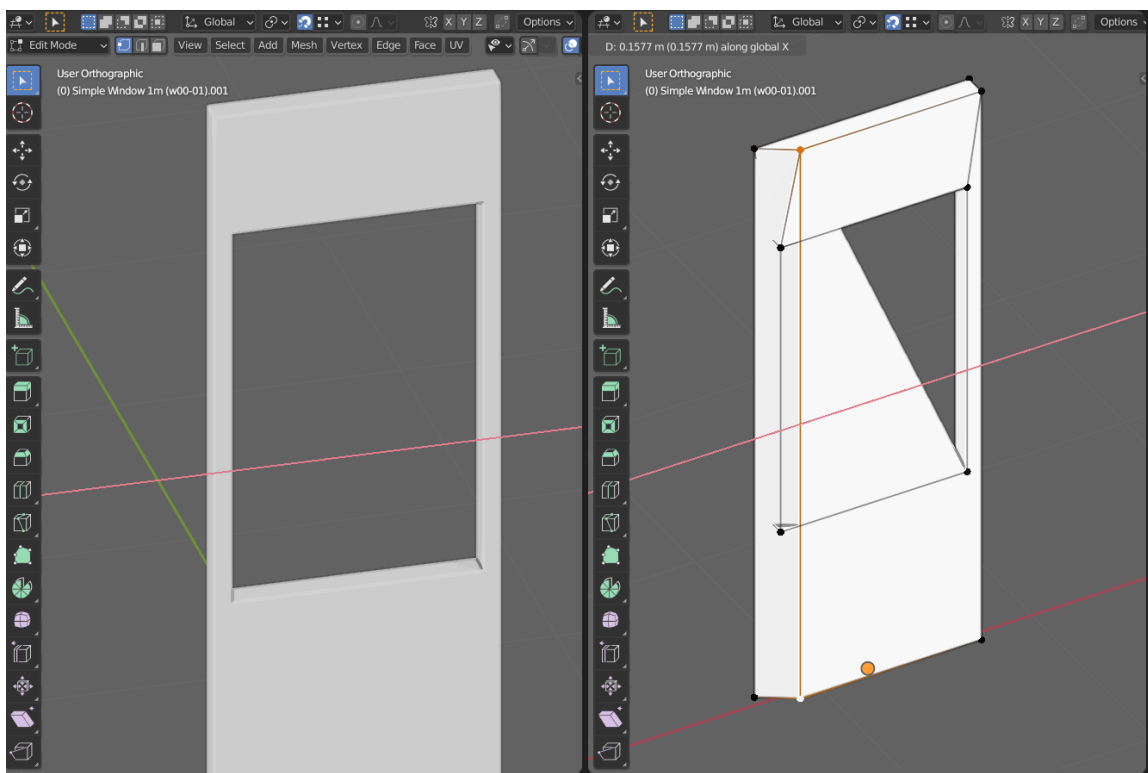
As mentioned earlier, how much the vertices will be displaced depends on the offset of the vertex along the z-axis relative to the pivot point, or, in other words, it depends on the wall thickness. This means that **the thicker the wall, the greater the offset** of the points.

For 3 angles this equals to:

- right angle = half width
- obtuse angle = (half width / $\sin(45^\circ)$) - half width
- sharp angle = (half width / $\sin(45^\circ)$) + half width

So.. when you are creating a window hole or doorway or any part that **does not participate** in corner modification make sure the wall has **enough space** from the corner up to this part to move the corner without **overlapping** or any other issues.

For example: *on the screenshot below you can see there is not too much space on the sides of the window and if we try to move the corner (as MBS will do) we get overlapping issues.*

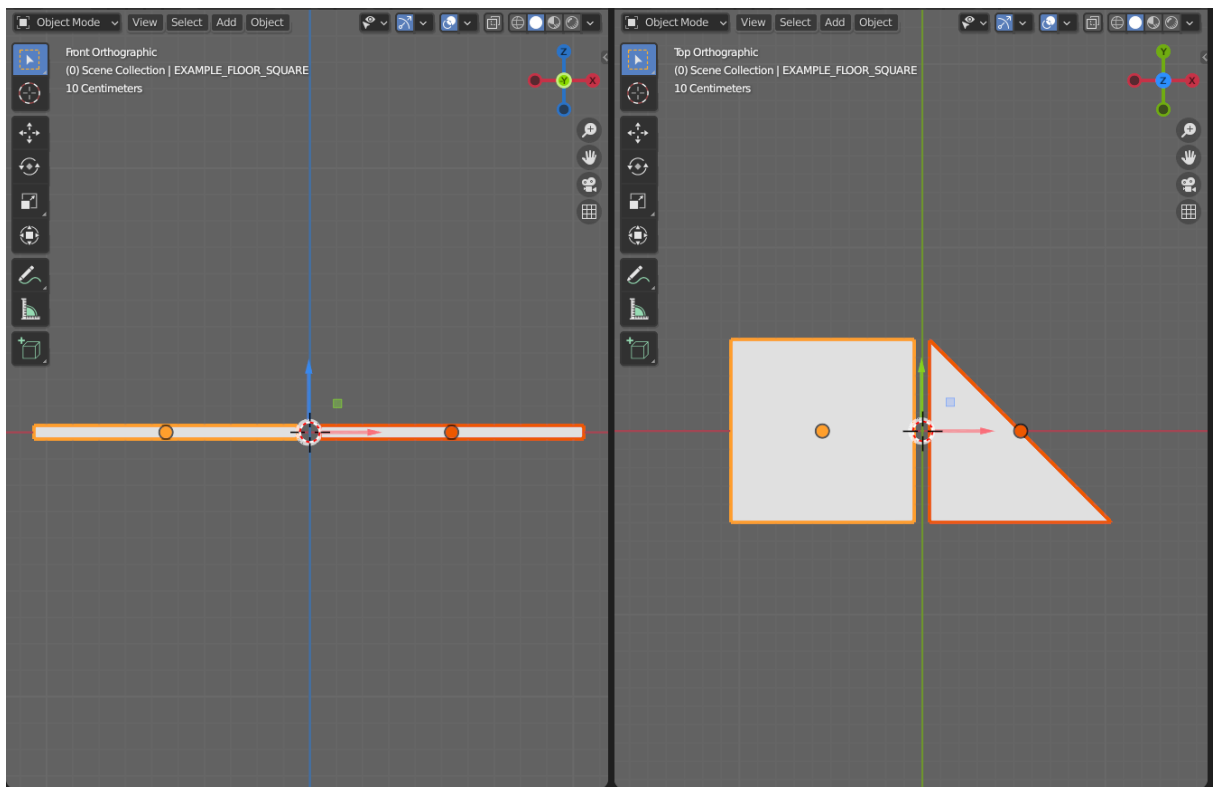


Floor assets creation

To prepare floors to work in MBS please follow next rules:

1. Pivot Point (Origin)

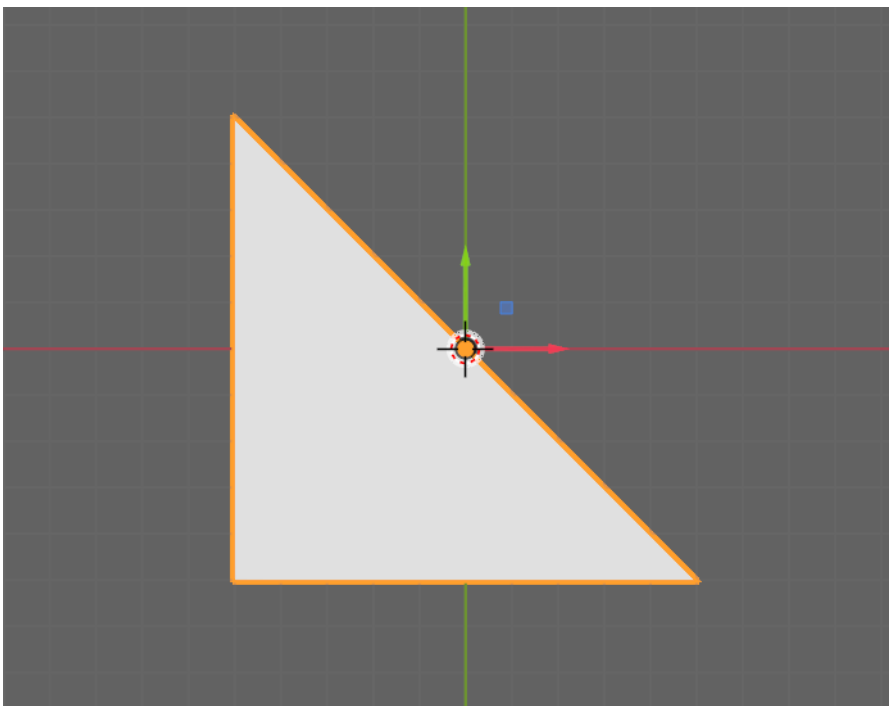
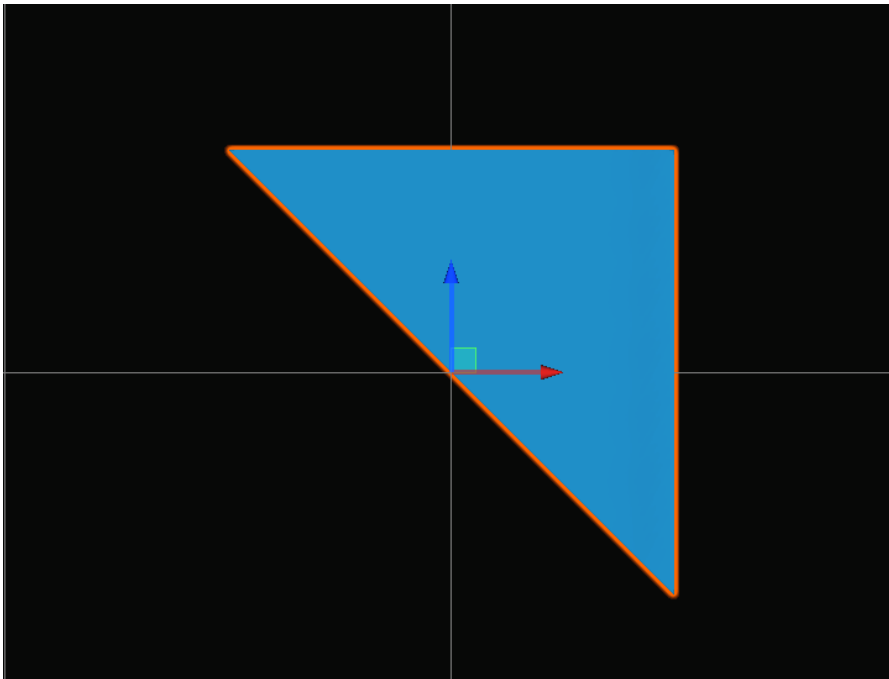
The pivot point of floor tiles should be in the **center** (of the square) horizontally for **both** square-tile and corner-tile. But vertically you can change it to achieve some special results. The best practice is to set it in the middle vertically.



2. Corner tile orientation

There is no special setting for **square tile**, this is just x-z (x-y blender) square with needed height.

But for the **corner tile**, make sure that the remaining half of the cut is lying in **+X** and **+Z** space (-X and -Y Blender).



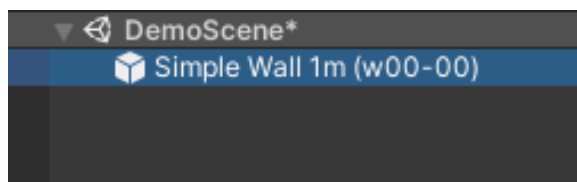
Import Assets

1. Model import requirements

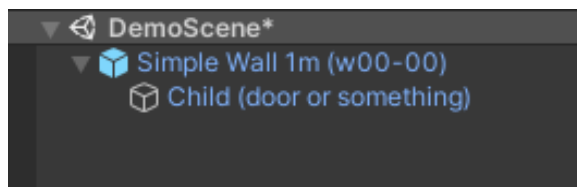
- 1.1. Scale changes are applied to the model and equal (1, 1, 1)
- 1.2. Rotation changes are applied to the model and equal (0, 0, 0)

2. Prefab layout

- **Single-object prefab:** an object should have a MeshFilter component with Mesh attached.



- **Multi-object prefab:** one object (the biggest one) should be a parent with MeshFilter attached, and the rest objects should be the children of that object.

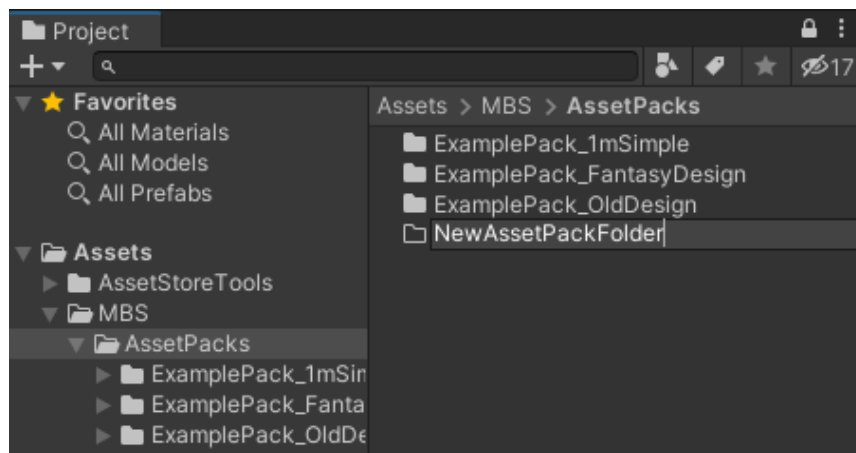


- **LODGroup prefab:** the parent object should have a LODGroup component, and **should not** have a MeshFilter component. LODs should be the children of that object and must be done according to the rules described above for **Single-object** or **Multi-object** prefab.

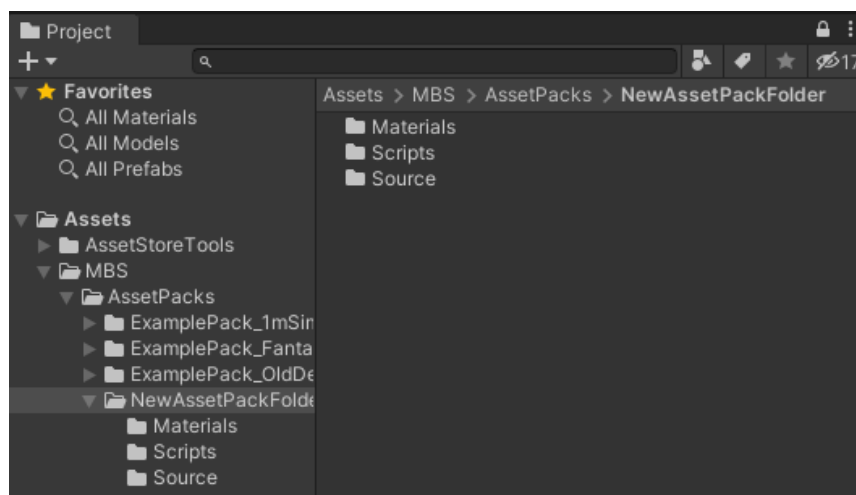


3. Adding Assets to the MBS

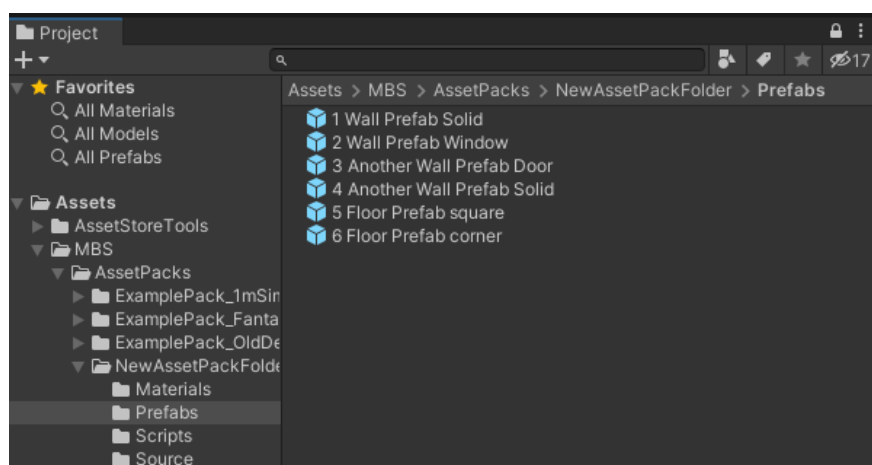
- 3.1. Create a **folder** for your assets inside **MBS / AssetPacks**



- 3.2. Import your models, textures, script, and so on into **your folder**



- 3.3. Create a **Prefabs** folder inside **your folder**, and add there all prefabs that will be used.

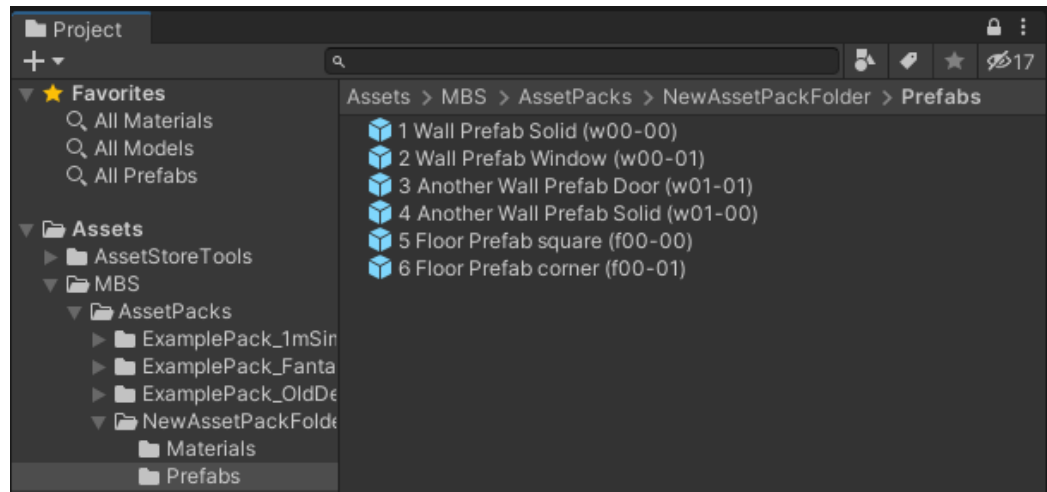


- 3.4. Add to the end of the name of all prefabs this construction

- For walls: **wAA-BB**
- For floors: **fAA-BB**

where **AA** - asset (group) number, and **BB** - prefab index.

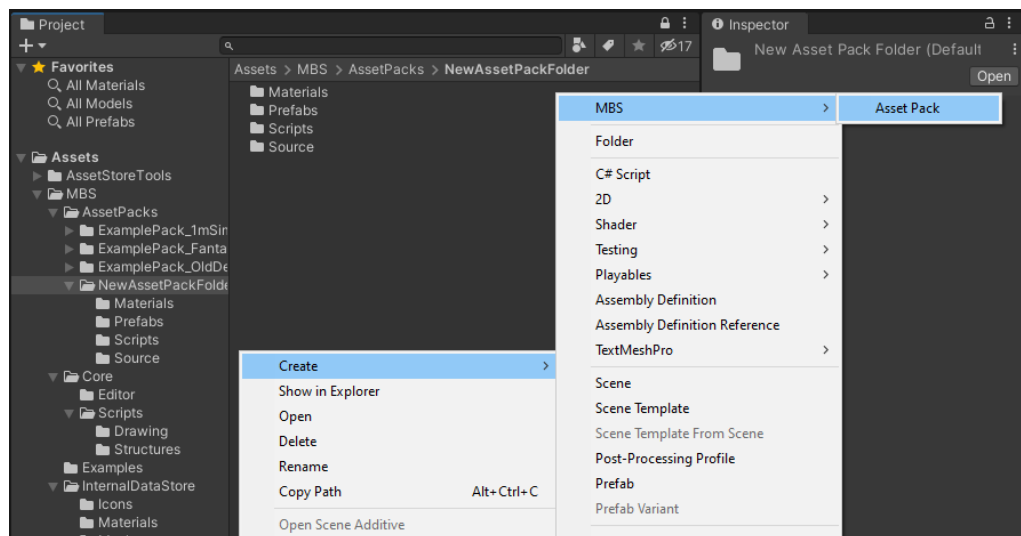
For walls, **BB** can be **00-99**, but **for floors only 00 and 01**, where 00 is used for square floor tile, and 01 for corner floor tile.

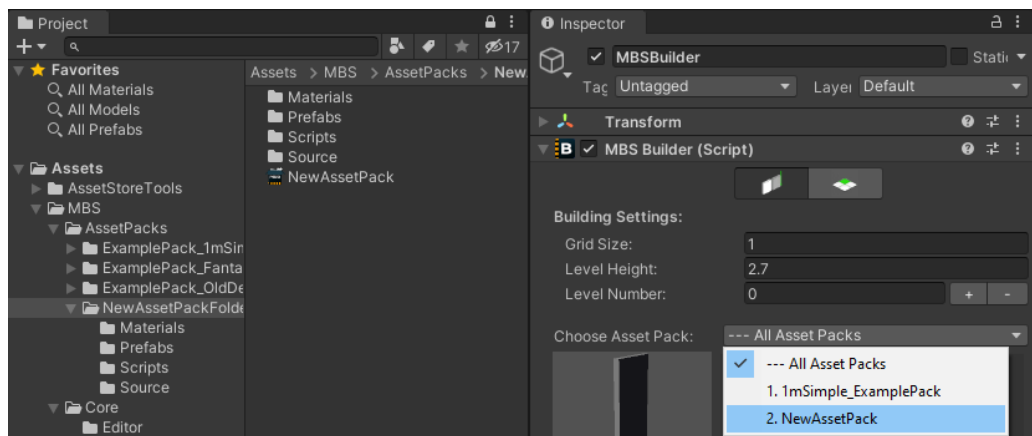
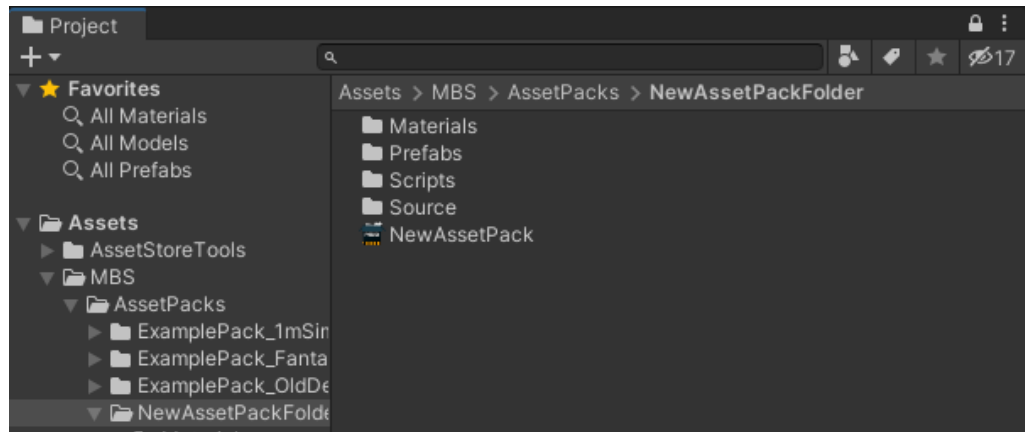


** You can use additional folders inside the **Prefabs** folder to sort prefabs as you needed.*

- 3.5. Create an **MBS Asset Pack** file inside **your folder** via the context menu and rename the file, the name of the file will be shown in the Asset Packs selection dropdown menu (MBSBuilder Inspector).

- (Project) Right Click -> Create -> MBS -> Asset Pack





Done! Asset imported. Now you can use it in MBS.