

操作文档

文件夹目录

美术资源目录

\arts	美术资源总目录
\character	角色动画
exportAll.py	导出脚本 递归导出目录下的所有文件
\hero	主角
move.aseprite	若干个动画的文件
export.py	导出脚本
idle.png	单个动画的导出文件
\monster	怪物
...	
\bullet	子弹
\hero	主角的子弹
\effect	特效
\scene	场景
\object	场景内的物体
\terrain	场景地形
level01.aseprite	level01的地形数据 多个图层
exportTerrain.py	导出地形数据的脚本
level01_block.png	导出的地形数据 空气墙
level01_static.png	导出的地形数据 静态物体
level01_terrain.png	导出的地形数据 动态生成的物体
level01_terrain_r.png	地形R通道的贴图 手动绘制
level01_terrain_g.png	地形G通道的贴图 手动绘制
level01_terrain_b.png	地形B通道的贴图 手动绘制
\ui	ui素材

工程资源目录

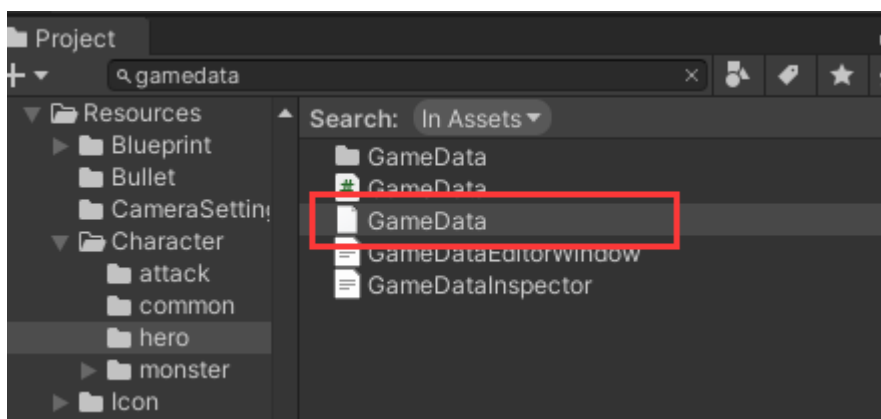
\client\Assets\Resources	资源总目录
\Blueprint	蓝图
\Bullet	子弹
\Character	角色的动画资源
\Icon	所有图标
\Prefab	所有预制件
\Bullet	子弹的预制件(不用管)
\Common	通用的预制件 比如影子
\Effect	特效
\Interact	交互物
\Level	关卡
\Monster	怪物
Player.prefab	主角
\SceneAsset	场景资源

\Object	场景内的物体
\Terrain	地形
\Scenes	unity场景(不用管)
\Texture	不好分类的贴图
\Timeline	时间轴
\UIRes	FGUI导出路径

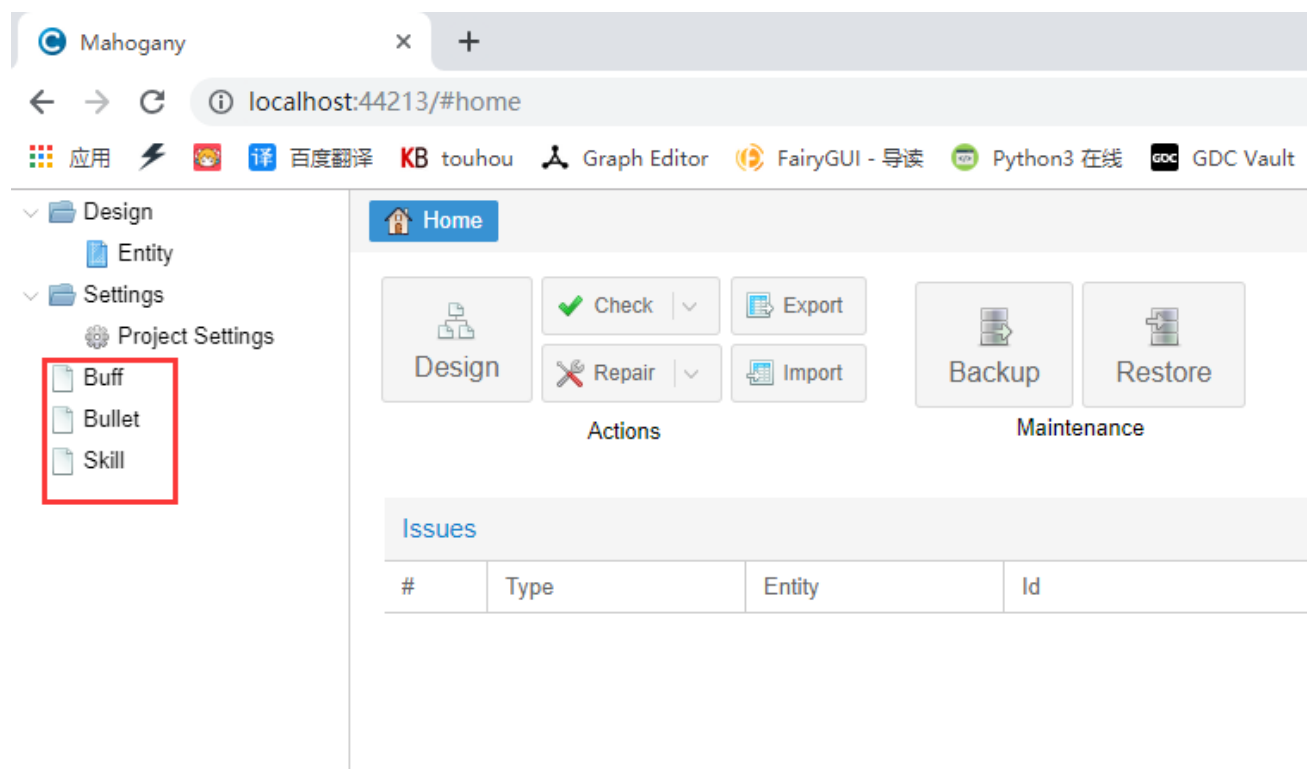
导表

导表编辑器

在unity资源管理器中搜索 gamedata



双击这个文件，会打开一个网页

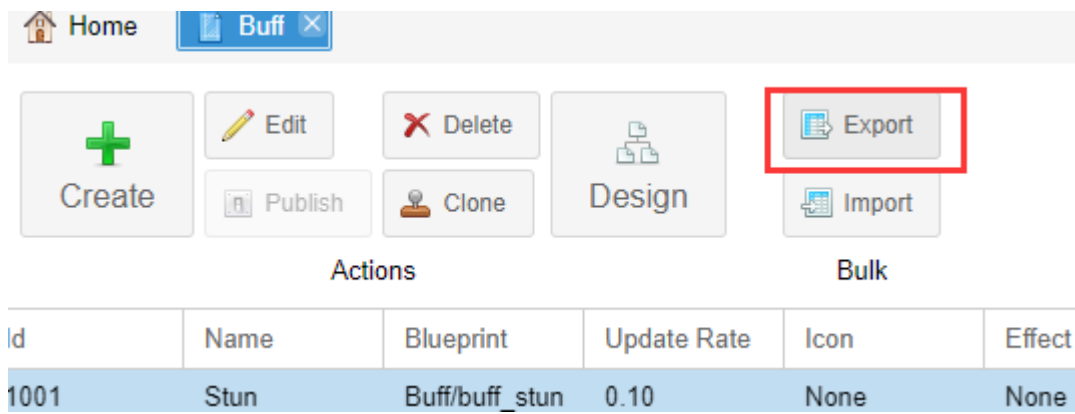


框选部分就是我们项目中使用的数据档。

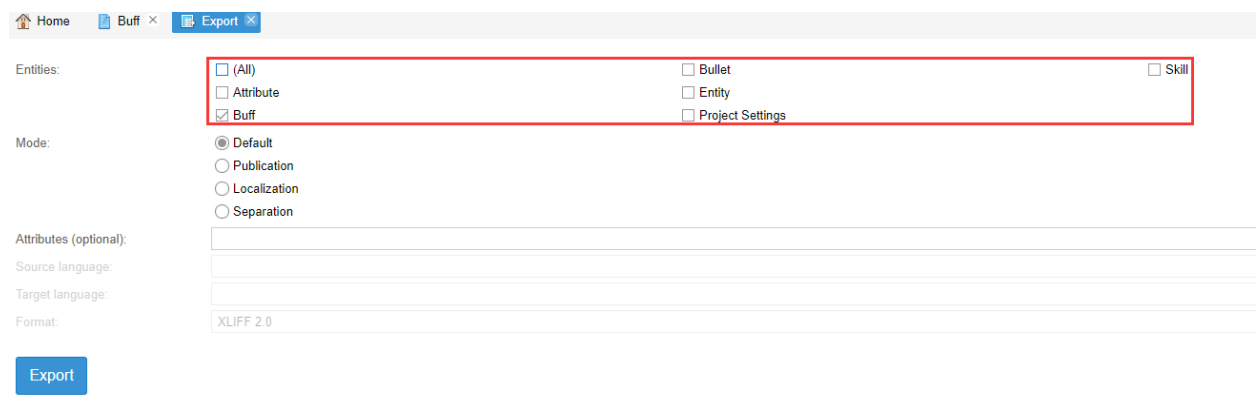
在excel下编辑

1.导出

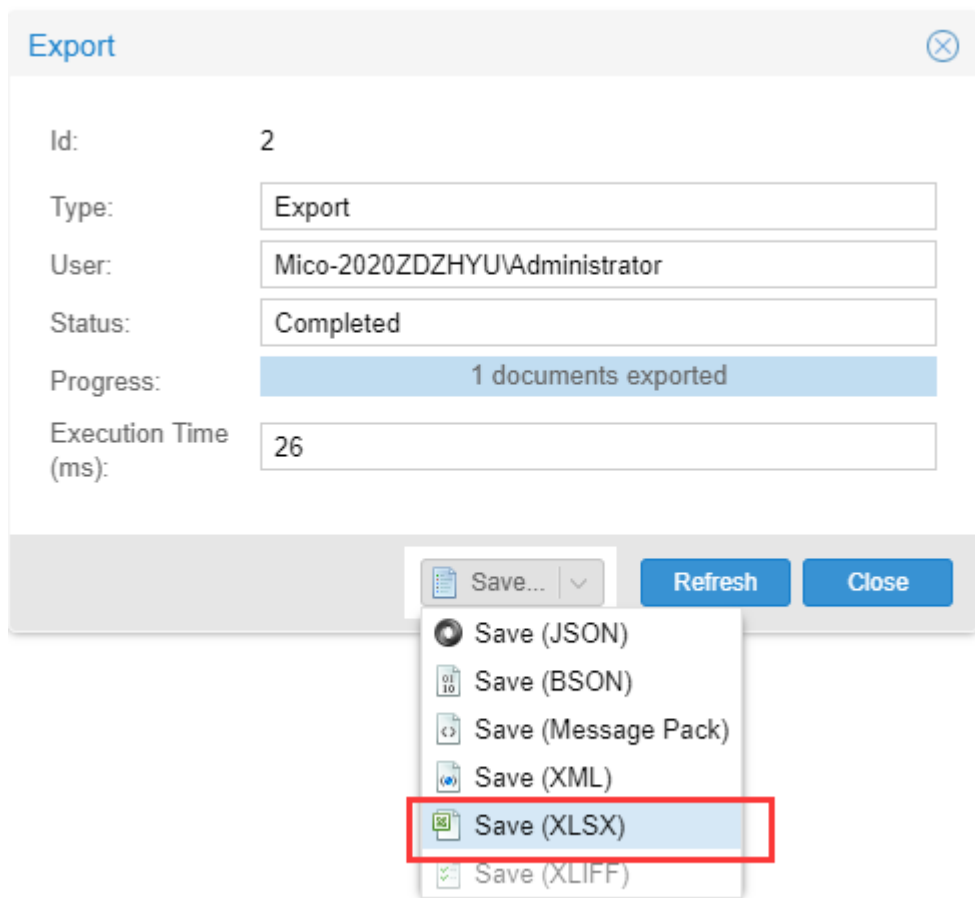
1. 在导表编辑器任意界面点击 Export`按钮



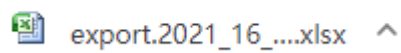
2. 勾选需要导出哪些表



3. 点击 Export 按钮，选择导出格式为excel

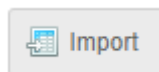


4. 浏览器下载列表会出现这个excel文件，编辑这个文件即可。

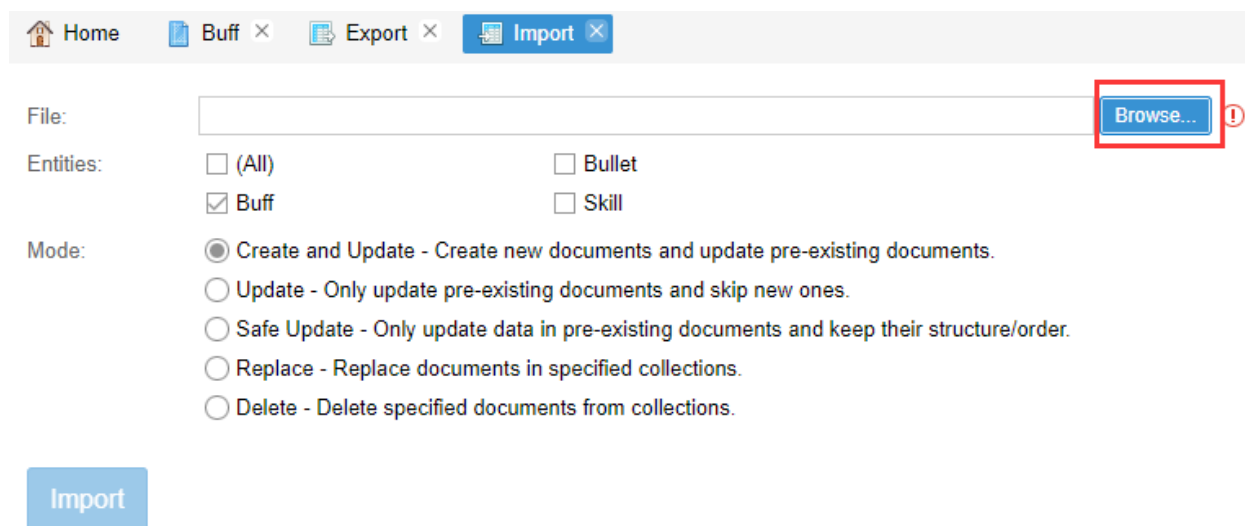


2.导入

1. 任意界面点击 Import 按钮



2. 选择excel文件和要导入哪些表



3. 选择导入方式，一般选Replace直接覆盖

- Mode:
- ☐ Create and Update - Create new documents and update pre-existing documents.
 - ☐ Update - Only update pre-existing documents and skip new ones.
 - ☐ Safe Update - Only update data in pre-existing documents and keep their structure/order.
 - ☒ Replace - Replace documents in specified collections.
 - ☐ Delete - Delete specified documents from collections.

4. 点击Import按钮

其他

他这个编辑器有点难玩，不如直接用excel编辑。主要是有自动生成代码的支持。后续可能改为直接编辑excel。

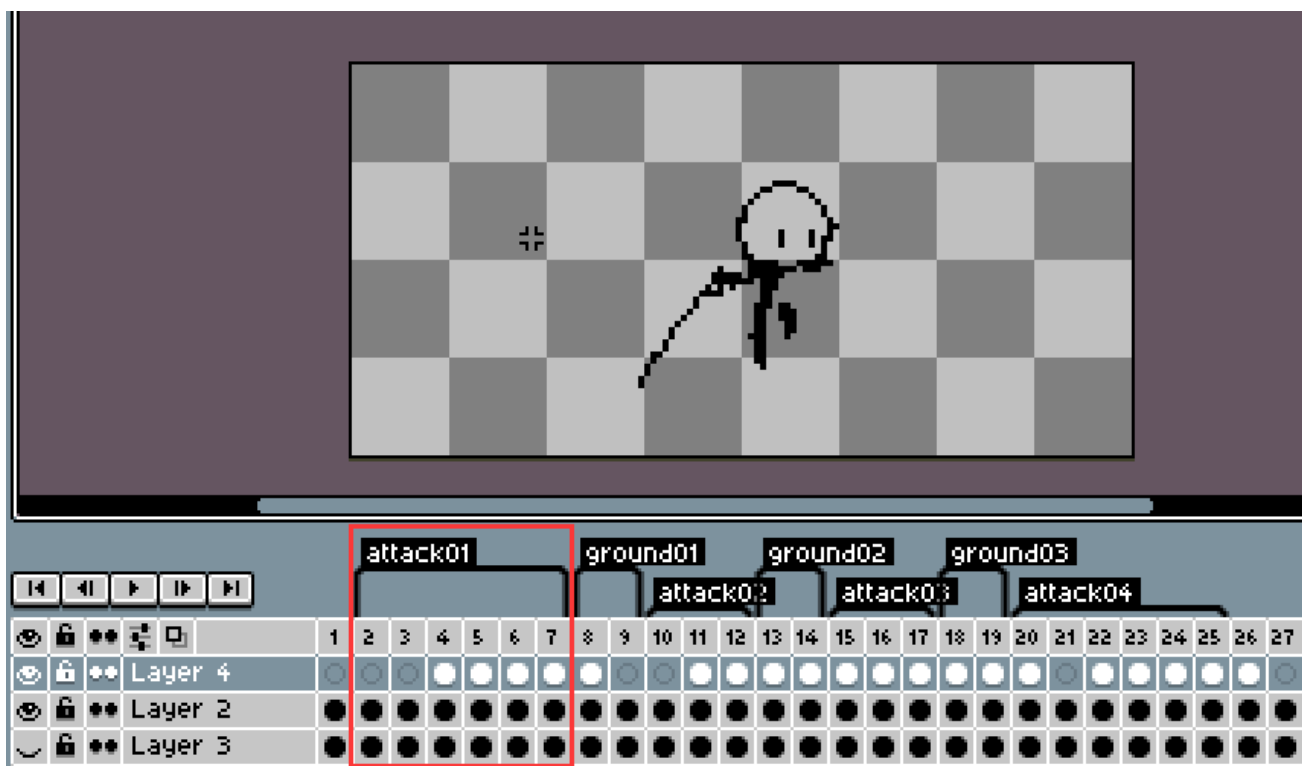
制作角色动画

Step1.绘制动画

1. 在对应的美术资源目录下新建aseprite文件。
 - 对于一般角色，规定文件尺寸为 128*64
 - 一个aseprite文件对应若干个动画，用tag区分每个动画的名字
2. 绘制动画

如图，2-7帧为attack01动画的帧，用tag区分

对于图层无要求，导出时会导出所有可见图层。目前分3层，上层是攻击特效(测试用)，中层是动画，下层是背景。导出前将背景设为不可见。



Step2.导出动画文件

动画文件会根据tag，从aseprite导出为若干个png，生成到aseprite文件所在目录。

操作：

1. 安装python3
2. 使用文本编辑器打开对应的导出脚本py文件，修改 `ASEPRITE_PATH` 字段，指向本机的aseprite软件的exe文件。（不能是steam的快捷方式文件）

```

exportAll.py E:\...effect X BuffSystem.cs BulletSystem.cs EffectPoolManager.c
E: > Mahogany > arts > effect > exportAll.py > ASEPRITE_PATH

1  import os
2
3  ASEPRITE_PATH = "D:/Steam/steamapps/common/Aseprite/aseprite.exe"
4
5  FILE_NAME_FLAG = "[FILE_NAME]"
6  TAG_NAME_FLAG = "[TAG_NAME]"
7  OUTPUT_NAME_FLAG = "[OUTPUT_NAME]"
8  CMD_QUERY_TAGS = ASEPRITE_PATH + "-b---list-tags-" + FILE_NAME_FLAG
9  CMD_EXPORT = ASEPRITE_PATH + "-b---tag-" + TAG_NAME_FLAG + \
10 | ..... + FILE_NAME_FLAG + "-sheet-" + OUTPUT_NAME_FLAG + ".png"
11

```

3. 使用命令行切换到导出脚本所在目录，运行脚本文件：

```
python exportAll.py
```

4. 后续脚本完善就改为导出为exe执行，不需要其他操作。复制这一步也可以改为脚本执行。

Step3.将导出的png文件复制到对应工程目录

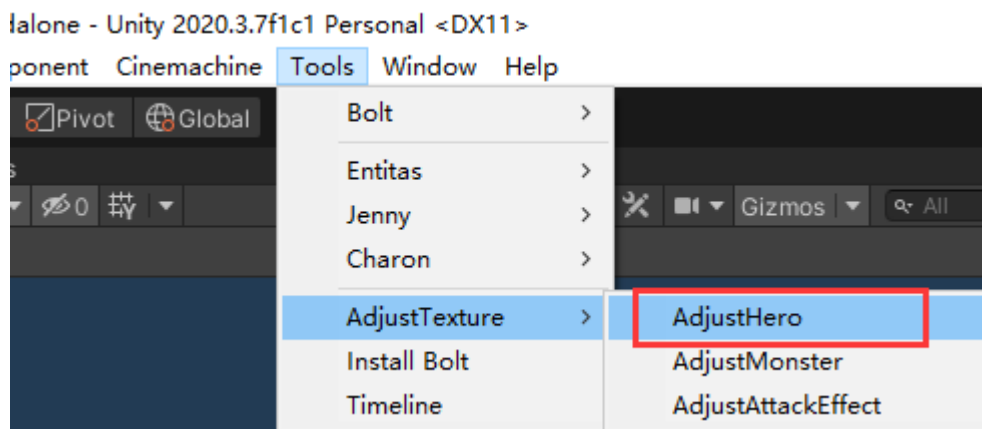
例如：主角的动作文件 attack01.png

从 \arts\character\hero\ 复制到 \client\Assets\Resources\Character\hero\

这里推荐使用文件浏览器软件：Q-Dir

Step4.从png生成anim

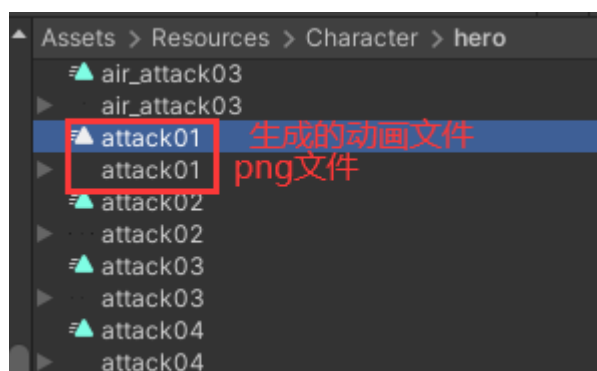
进入unity编辑器，使用生成工具进行生成



作用为：

1. 自动调整texture属性
2. 自动切片
3. 按切片文件生成 .anim 动画文件

点完之后会在texture同目录下生成 .anim 动画文件，这个文件就是可以直接在时间轴、动画状态机等地方使用的文件。



制作角色技能

角色技能比怪物技能承载了一些额外的信息，涉及连招和技能热拔插机制，需要在 Skill 表里配置。怪物的技能只需要制作 Timeline

Step1.配表

在技能表里新加一行，参数意义如下：

Id	Name 名字	开始姿态 StartStance	默认结束姿态 EndStance	释放类型 TriggerType	时间轴文件 Timeline	是否无视锁定 FreeLock	是否缓存姿态 Stance Cache	Quality	Element	Performance	Icon	Describe
1001	Attack01	0	1	0	hero/Attack...	No	No	0	None	None	None	Desc_Attac...
1002	Attack02	1	2	0	hero/Attack...	No	No	0	None	None	None	Desc_Attac...
1003	Attack03	2	3	0	hero/Attack...	No	No	0	None	None	None	Desc_Attac...
1004	Attack04	3	1	0	hero/Attack...	No	No	0	None	None	None	Desc_Attac...
1005	None	0	0	0	None	No	No	0	None	None	None	None
1006	None	0	0	0	None	No	No	1	None	None	None	None
1007	None	0	0	0	None	No	No	2	None	None	None	None

需要配置的有：

- 1. 开始姿态
- 2. 结束姿态
- 3. 释放类型
- 4. 时间轴

姿态枚举说明：

0	Idle,	地面待机
1	Ground1,	地面姿态1
2	Ground2,	地面姿态2
3	Ground3,	
4	Ground4,	
5	Ground5,	
6	Ground6,	
7	Ground7,	
10	AirIdle = 10,	空中待机
11	Air1,	空中姿态1
12	Air2,	空中姿态2
13	Air3,	
99	None = 99,	没有(只出现在结束姿态,技能没有后续连招)
100	Any = 100,	任意(只出现在开始姿态,任意姿态都可以释放这个技能)
101	Cache = 101,	没用

释放类型枚举说明：

0	LightAttack,	轻攻击按下
1	HeavyAttack,	重攻击按下
2	LightAttackRelease,	轻攻击抬起
3	HeavyAttackRelease,	重攻击抬起
4	Flash,	闪避
5	Jump,	跳跃

例如：

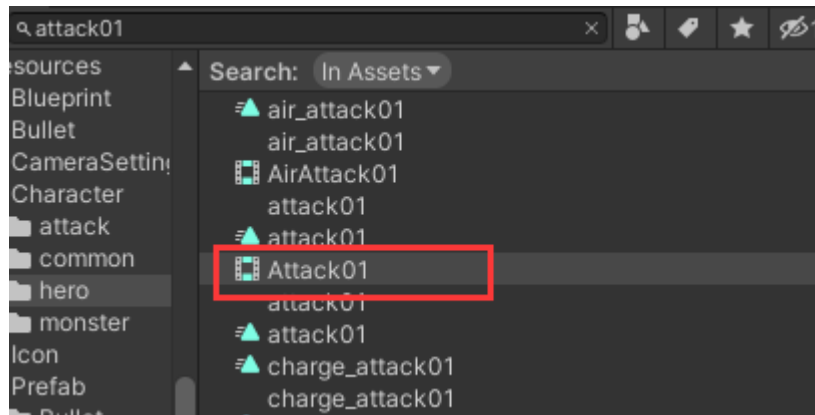
id	startStance	endStance	triggerType	timeline
2001	1	2	1	hero/Skill01

即：在角色姿态为 1(地面姿态1) 时触发 1(重攻击按下)，即可执行时间轴 hero/Skill01，执行进入后摇阶段时姿态为 2(地面姿态2)

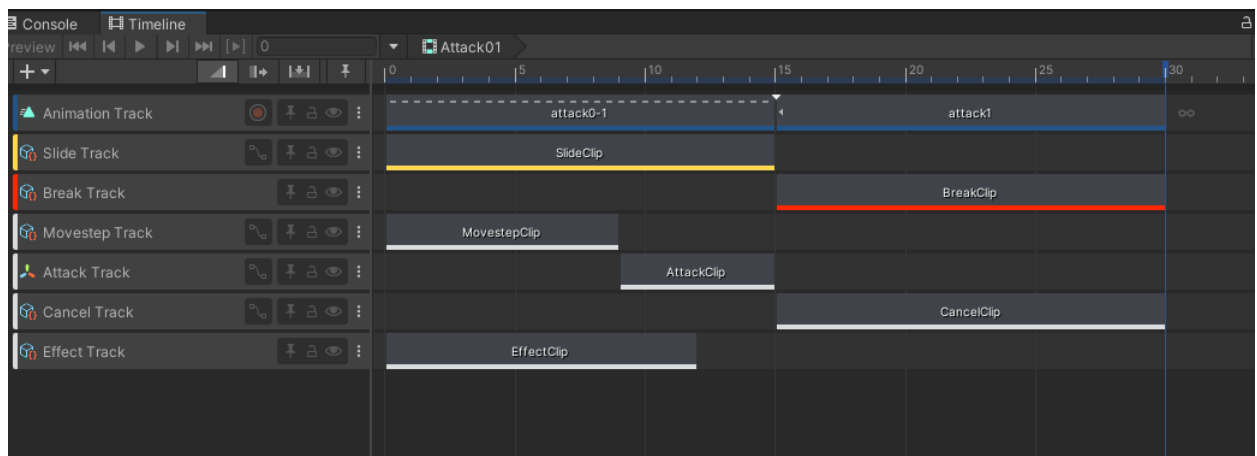
Step2.配置时间轴

以主角技能为例

1. 进入目录 \client\Assets\Resources\Timeline\Skill\hero，复制一个 .playable 文件，重命名为自己想要的名字
2. 进入unity资源管理器找到这个文件，并双击

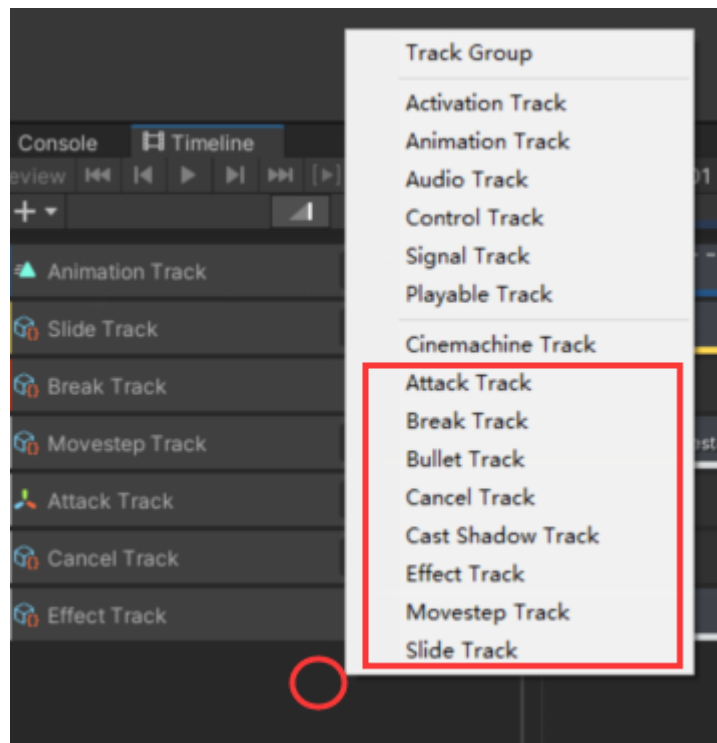


3. 此时会打开timeline编辑器。即可对时间轴进行编辑。



4. 编辑操作

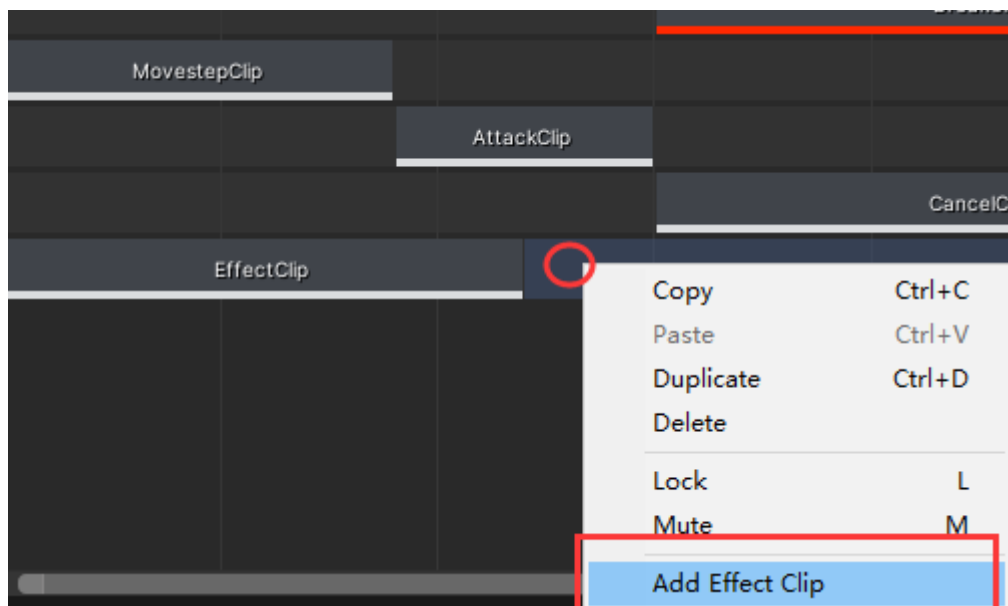
1. 添加Track：需要增加某一类型的操作时，在左边Track栏空白处点右键，选择需要添加的Track



红框中为项目中使用的自定义Track。

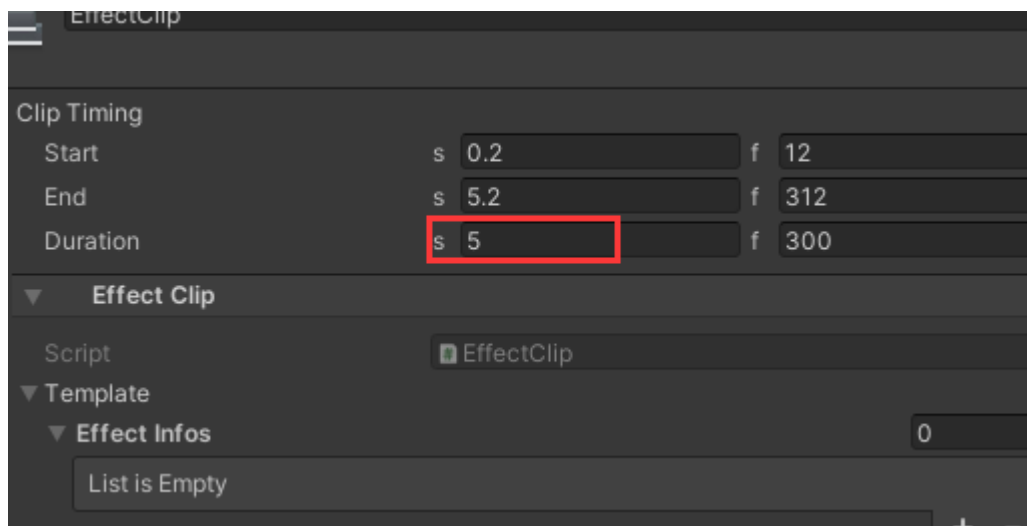
点击后即可添加一个空的Track轨道。

2. 添加Clip：在某个Track轨道的空白处点右键，添加Clip

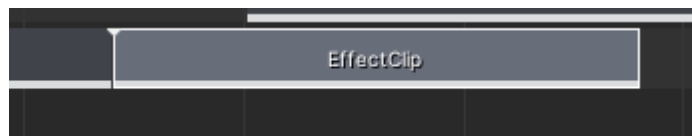


unity添加clip的默认长度值很长，可以先手动修改一下clip的长度。

单击clip，在右侧 inspector 页面中修改 duration 字段：

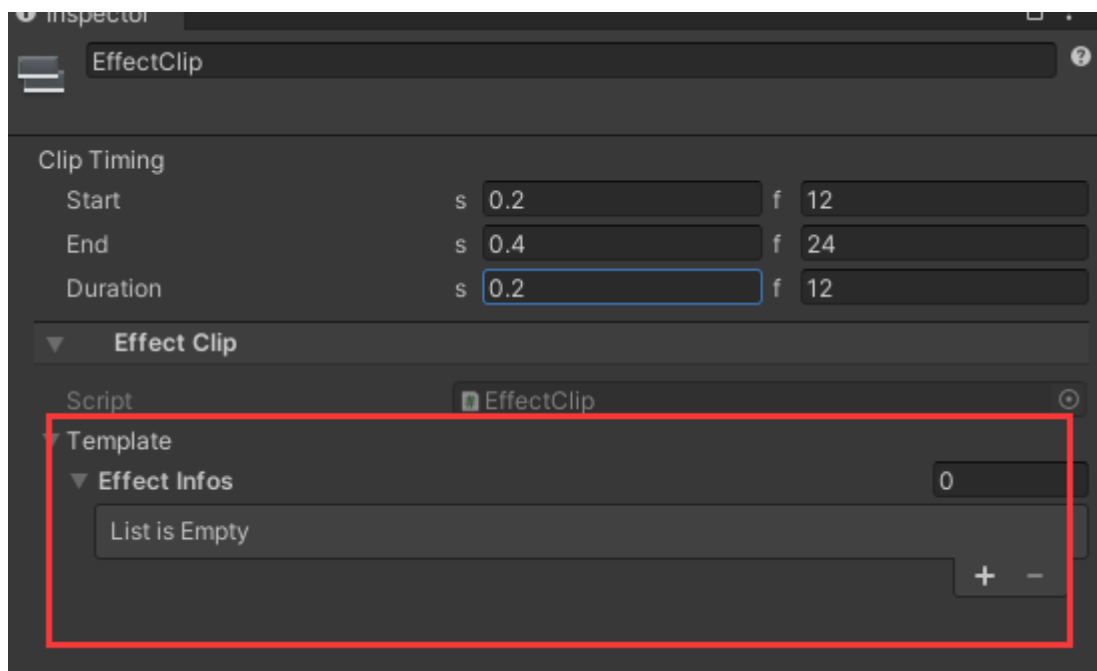


s指的是秒数，f指的是帧数，修改任意一个即可。



3. 修改Clip参数

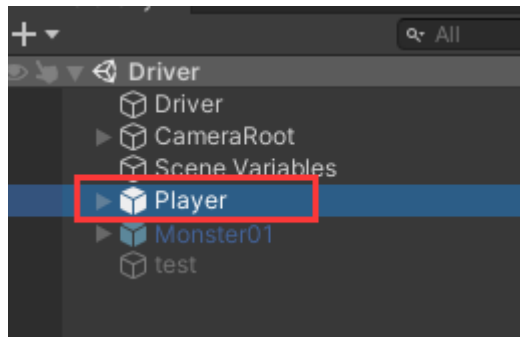
单击clip，在右侧 inspector 页面中修改 Template 下的字段即可，每种clip的字段不一样，按具体含义修改。



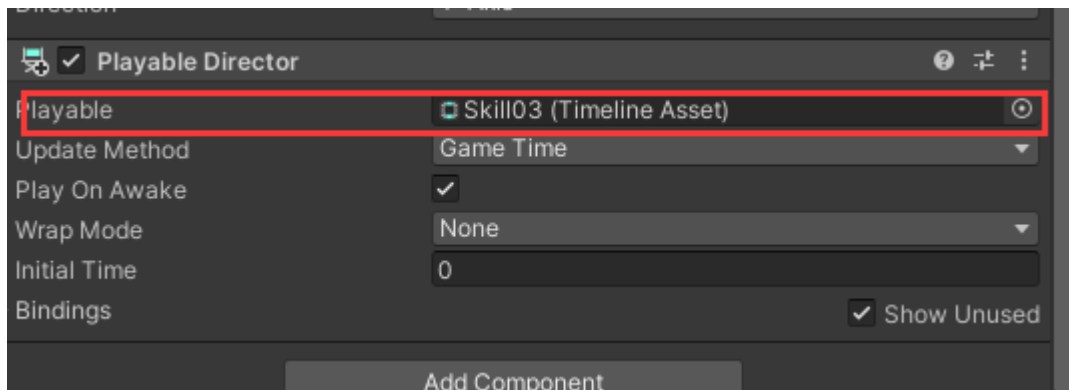
Step3.测试时间轴（可跳过）

这一步的目的主要是方便快速检测动画和判定框。这两种Clip比较需要实时检测。

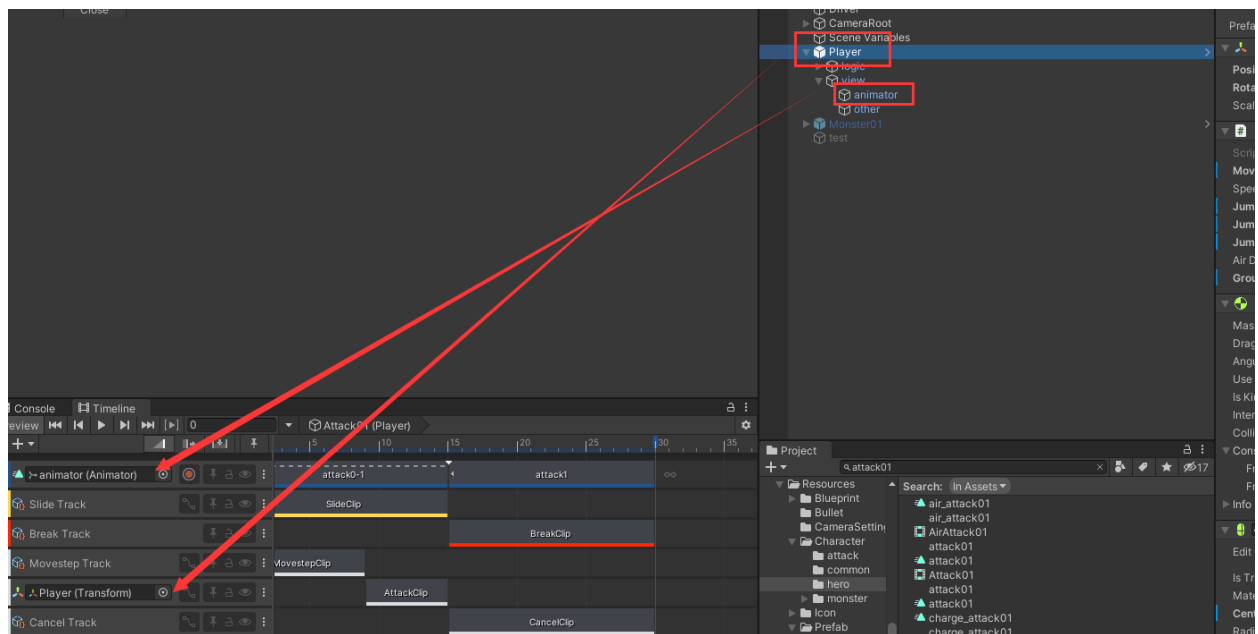
1. 选中Driver场景中的 Player 物体



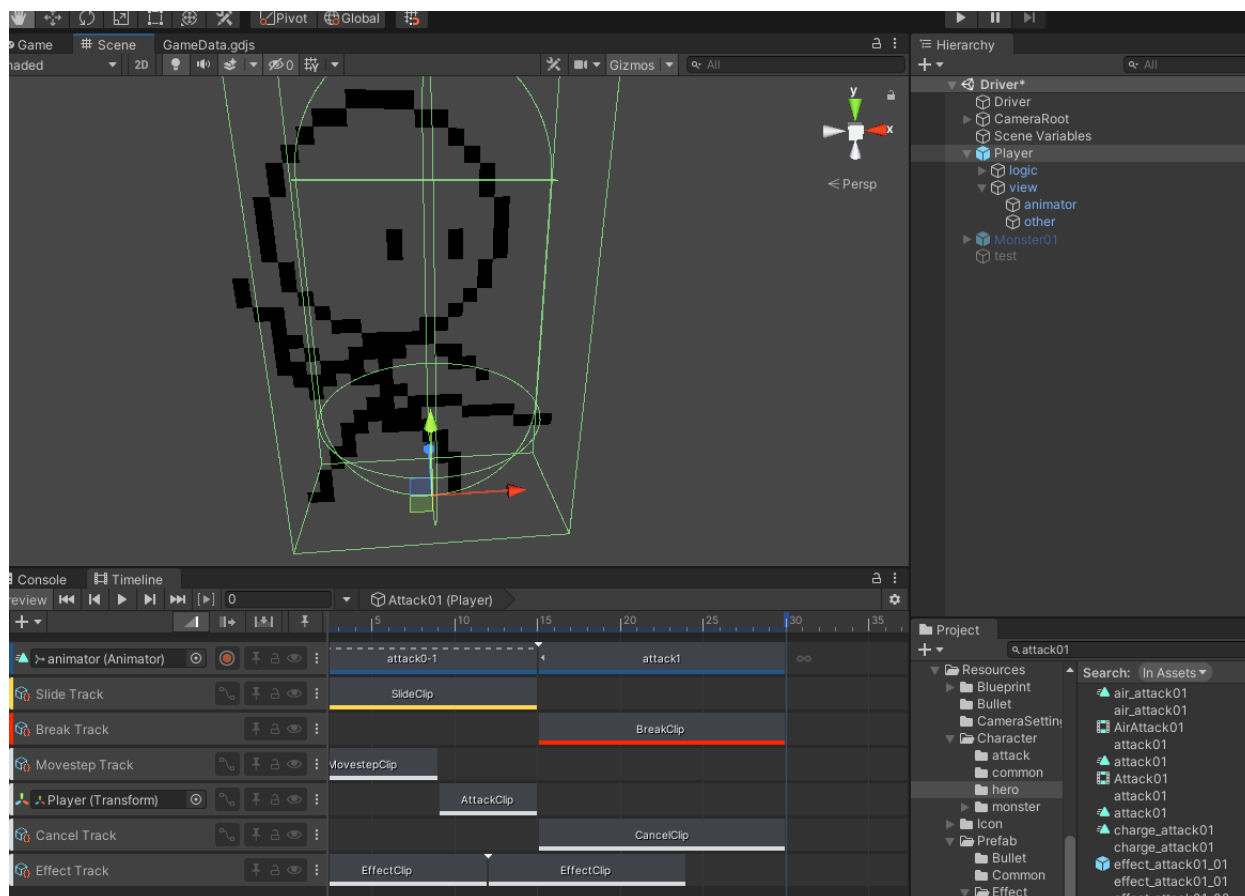
2. 将需要检测的timeline文件(.playable)拖入右侧 inspector 页面中的 PlayableDirector



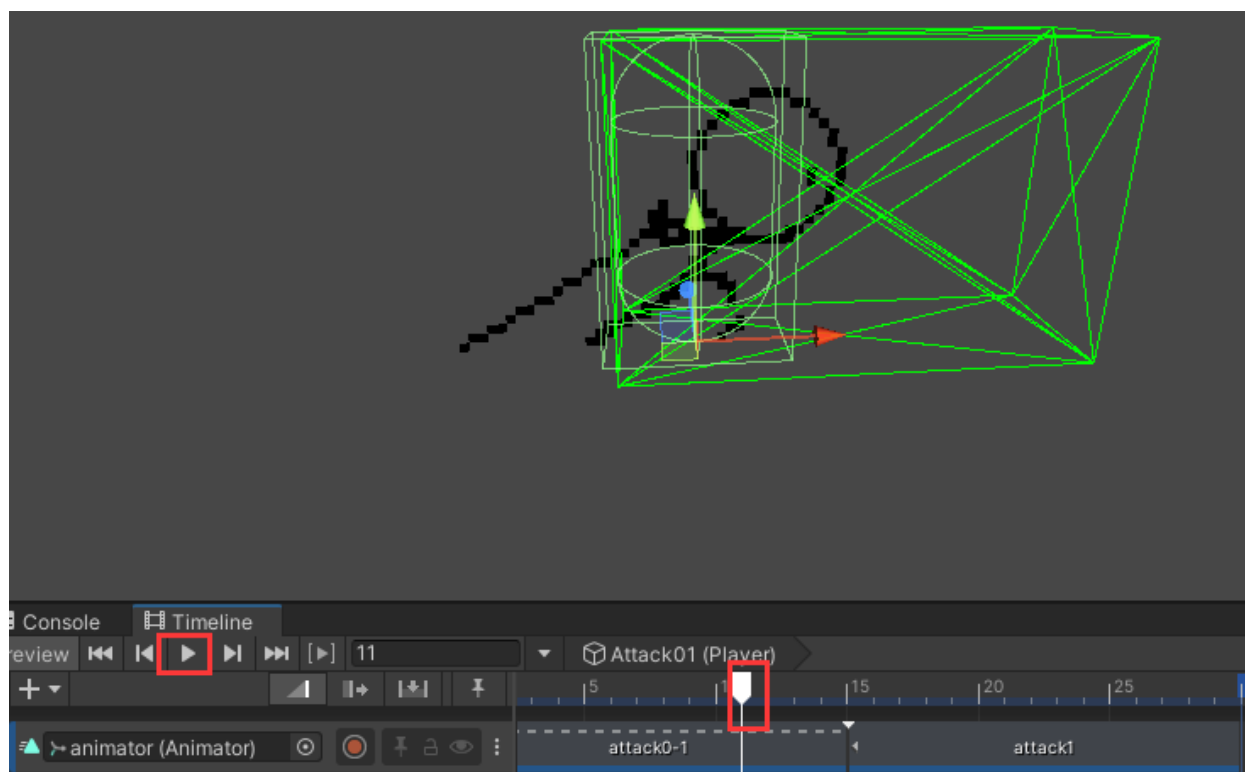
3. 将这图中两个物体拖入打开的时间轴编辑器内对应的Track中



4. 打开Scene页面，双击Player物体，使之位于屏幕中央

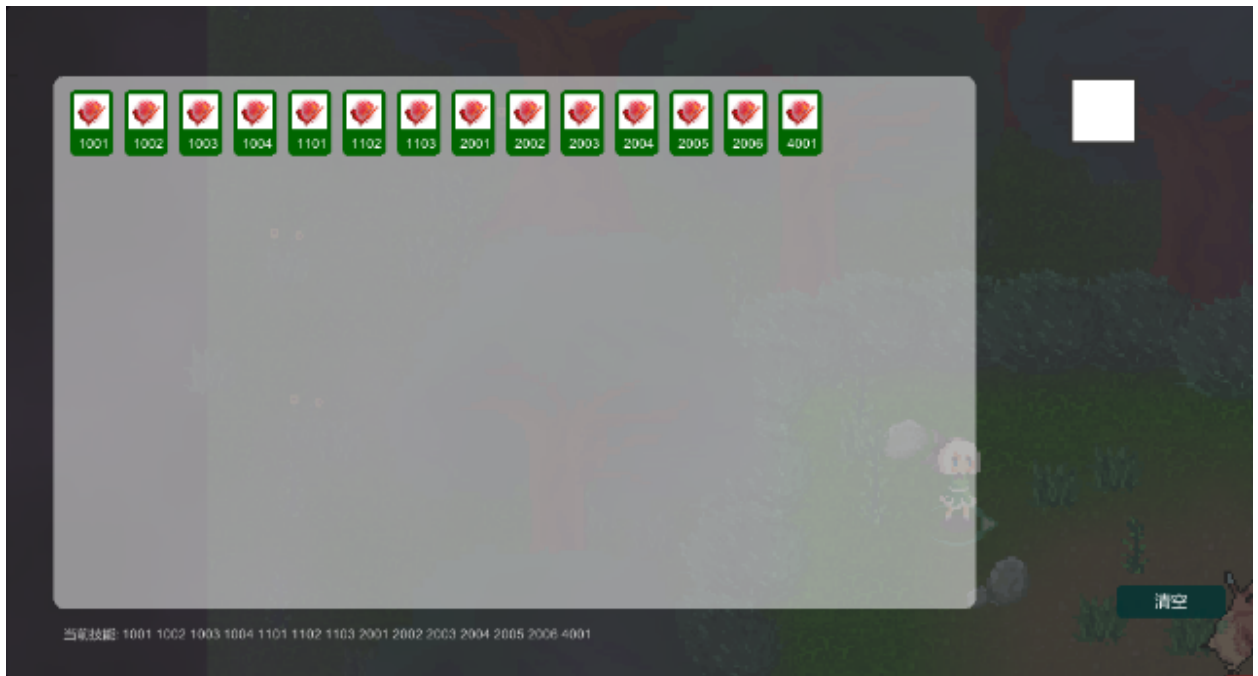


5. 预览时间轴效果。可以点击时间轴编辑器的播放按钮，也可以拖动时间轴上的锚点进行逐帧查看



Step4.测试技能

1. 进入游戏后按P呼出技能配置UI。



2. 列表中是所有技能，绿色为已装备，灰色为未装备。点击自己要测试的技能装备上，即可按照连招树打出技能。

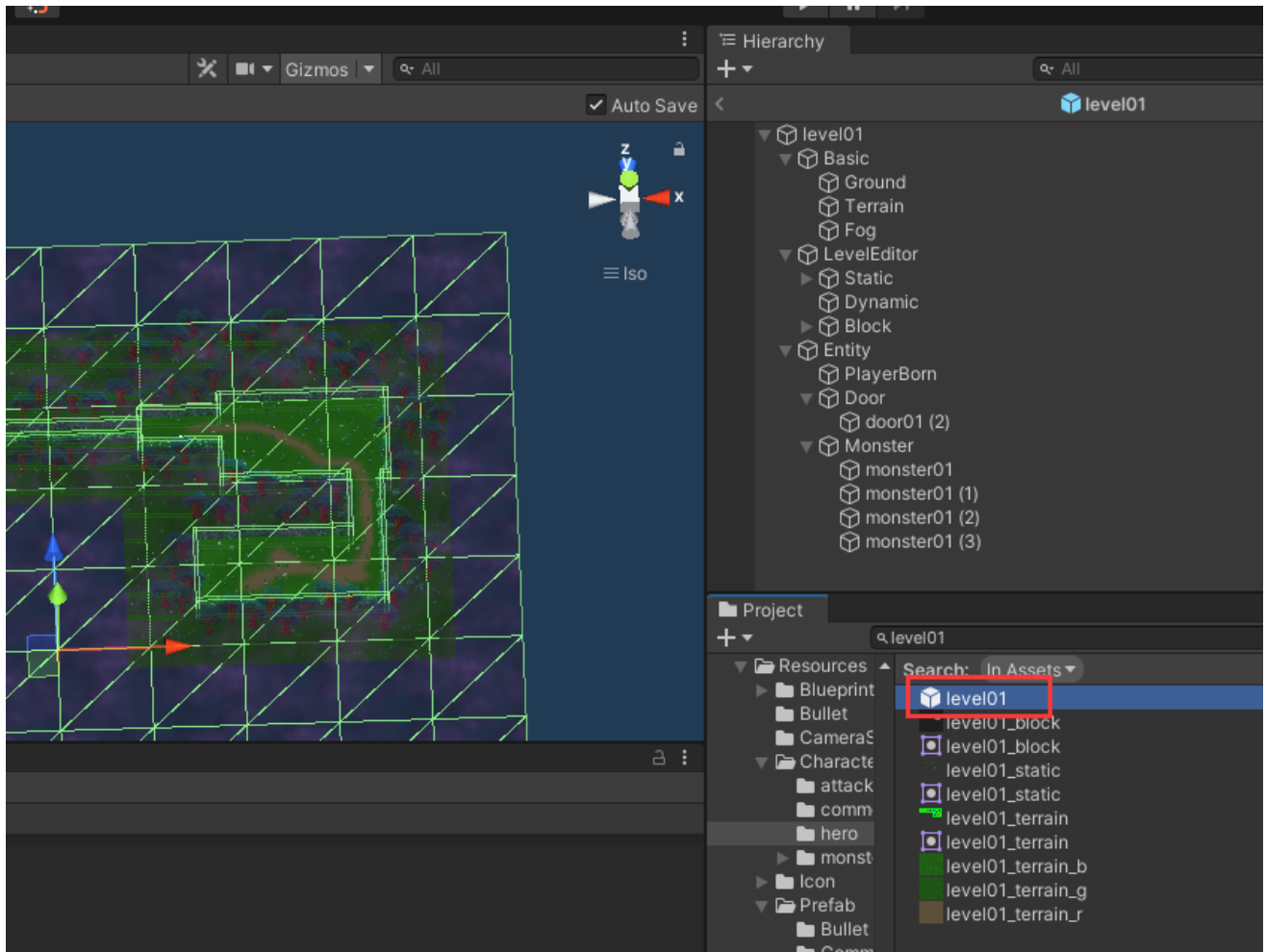
制作关卡场景

Step1.复制一个场景文件

场景文件在 `\client\Assets\Resources\Prefab\Level`，复制 `level01.prefab` 到同目录，重命名为想要的名字。

Step2.制作场景预制件

在unity资源编辑器中打开场景的prefab文件



场景prefab结构是这样的

1. Basic是地表的collider和sprite，以及雾效，这个不动就行了
2. LevelEditor生成场景物件，比如树，草
3. Entity 就是场景内的可交互实体，比如怪物，传送门，可破坏的箱子之类的

Step2.1.制作场景地表和物件

绘制场景物件

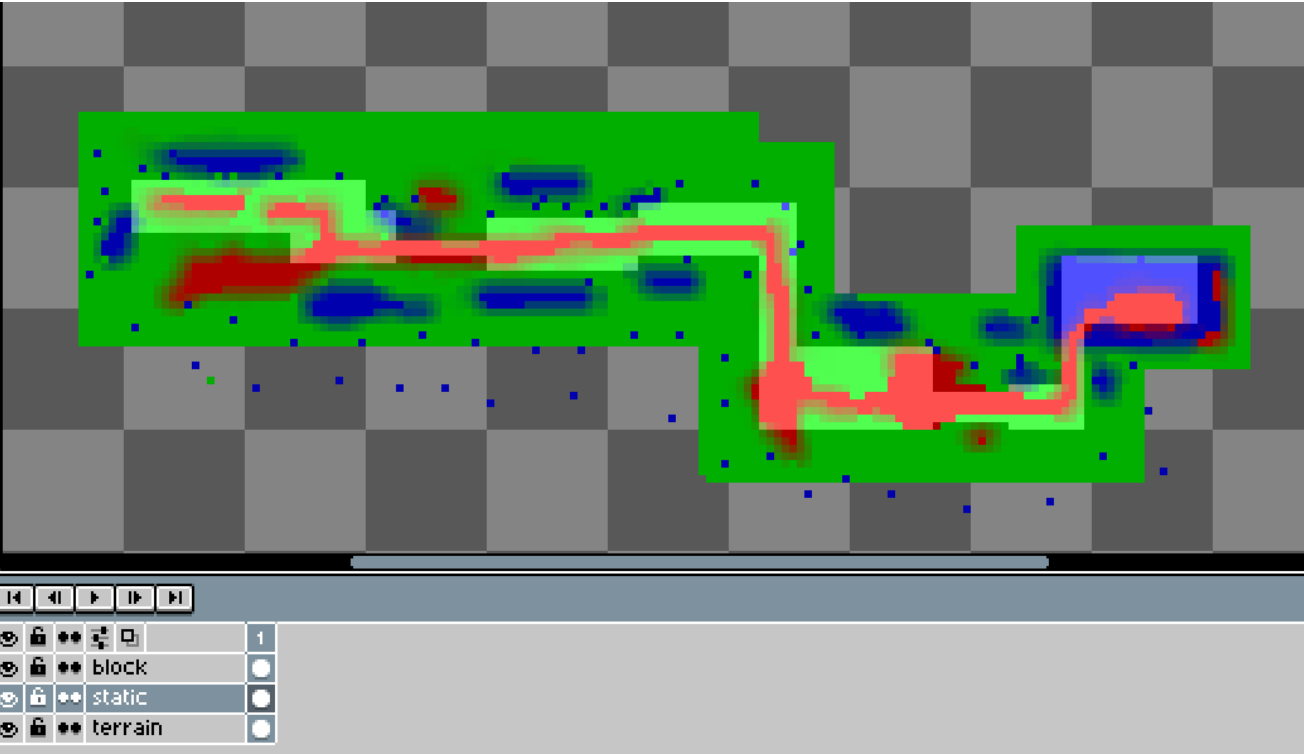
场景物件路径在 `\arts\scene\object`，没有特殊要求，绘制完毕之后复制到 `\client\Assets\Resources\SceneAsset\Object\` 即可。

绘制场景贴图

场景物件路径在 `\arts\scene\terrain`，文件含义为：

文件	含义	来源
level01.aseprite	贴图的原始文件	手动绘制
level01_block.png	空气墙贴图	自动导出
level01_static.png	静态物件贴图	自动导出
level01_terrain.png	地表贴图	自动导出
level01_terrain_r.png	地表贴图R通达代表的贴图	手动绘制
level01_terrain_g.png	同上	手动绘制
level01_terrain_b.png	同上	手动绘制
exportTerrain.py	导出脚本	无

主要需要制作的的就是 level01.aseprite 文件：



场景地编图分三层：

1. block用于规定真实关卡范围，画空气墙
2. static用于用像素点来画静态景观的位置，比如树和灌木
3. terrain用来生成混合地表贴图，以及控制动态装饰物的生成，比如草

场景地编图规定大小为256x256

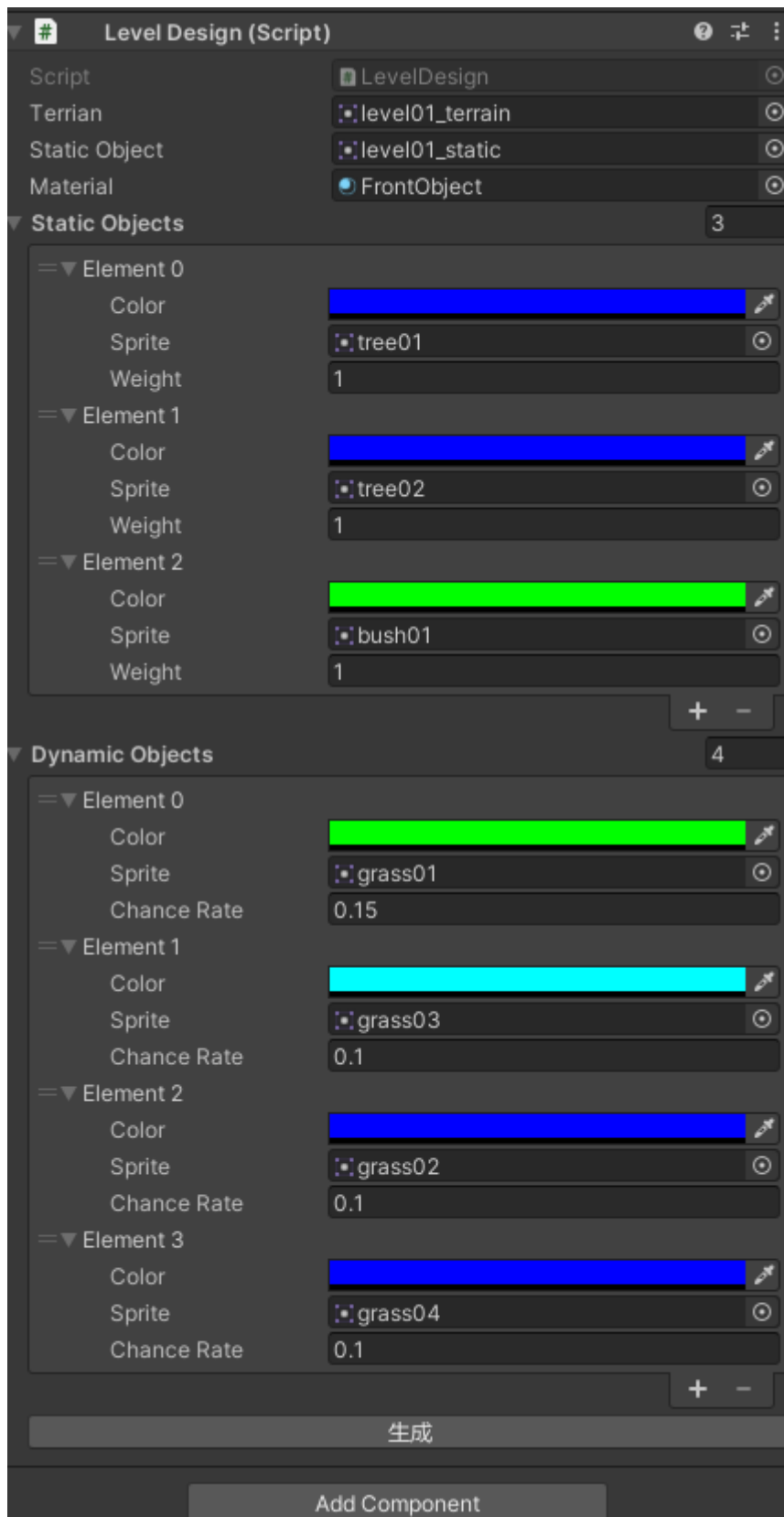
导出场景贴图

目录下运行 exportTerrain.py 进行导出。方法同角色动画。

复制贴图到工程目录

\client\Assets\Resources	资源总目录
\SceneAsset	场景资源
\Object	场景内的物体
\Terrain	地形

编辑场景物件



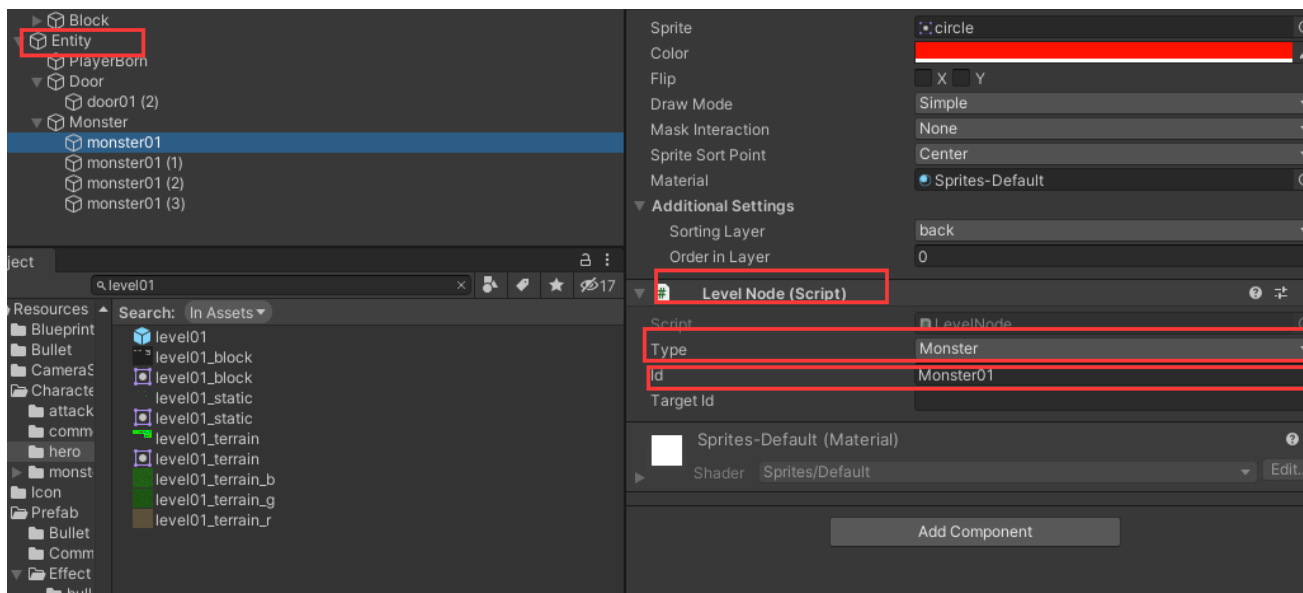
levelEditor节点挂了一个LevelDesign组件,

1. 把三张生成的地表图拖到这里 (每个关卡不一样)
2. material是透视效果需要的材质, 这个不用动
3. static object 就是固定点的树, 这里配置方式为 颜色 (一定要rgb完全一样), 贴图, 权重, 也就是一个颜色的点, 对应权重随机选一个贴图来刷

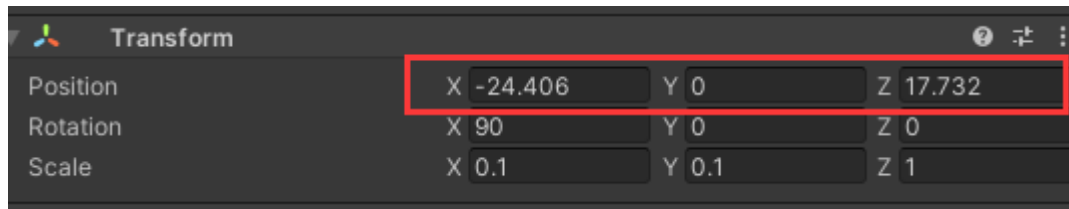
- dynamic object 就是地表上的随机装饰物，规则为：对于一个贴图，选择一个最可能刷这个装饰物的目标地表颜色，对于每一个地块读颜色，和这个目标颜色rgb作差，越接近的刷的概率越大，最后乘一个chance就是这个地块刷这个装饰物的概率
- 最后点生成按钮就会在LevelEditor节点下生成object

编辑场景实体

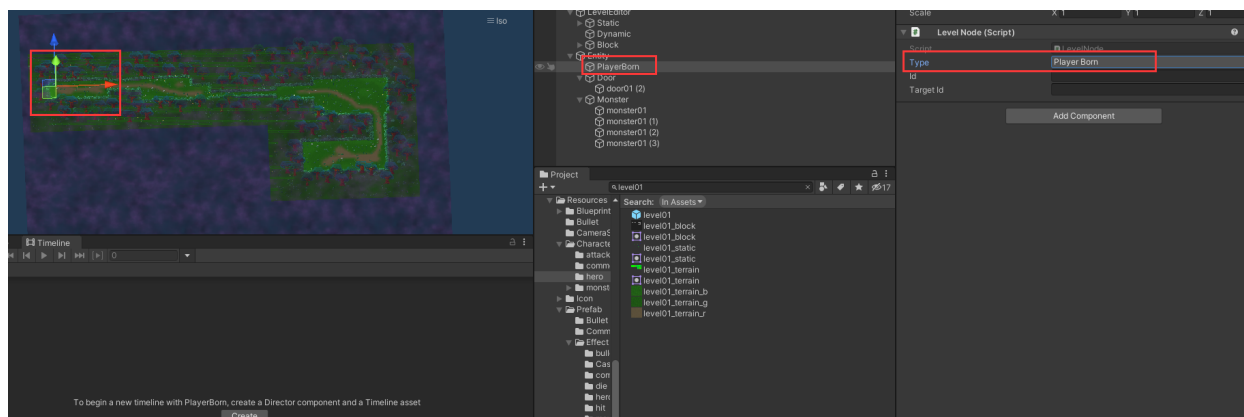
- 在Entity节点下创建新的节点(结构随意)，挂载 LevelNode 脚本



- 编辑新增节点的位置信息。注意y值为0



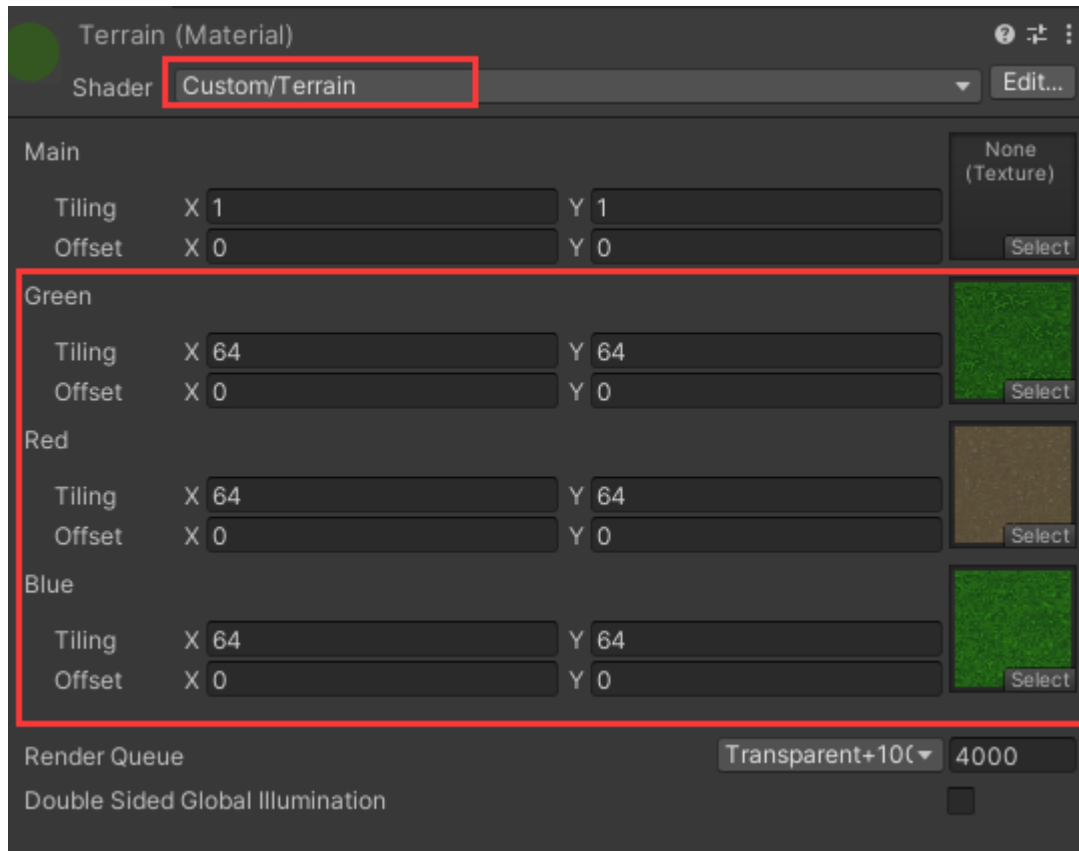
- 在 LevelNode 脚本中设置这个节点的类型(如怪物)，Id(如怪物id)，TargetId(额外信息，如传送门的目标关卡Id)。
- 一定要包含一个Type为 PlayerBorn 的节点，作为角色进入关卡时的默认位置。



编辑地表材质

不同关卡之间使用的地表材质融合方式是一样的，但是材质可能不一样。例如森林地图里材质贴图的RGB代表了泥土、浅草和深草，到了雪地地图可能变为石头、冰面、雪地。如果有这种需求需要对材质进行编辑。

1. 新建材质，shader选择 Custom/Terrain



2. 将RGB通道代表的材质拖到对应的参数内。
3. 设置材质的Tiling，即材质采样时的缩小倍率。64x64的贴图，Tiling为64x64；32x32的图，Tiling为128x128，以此类推。