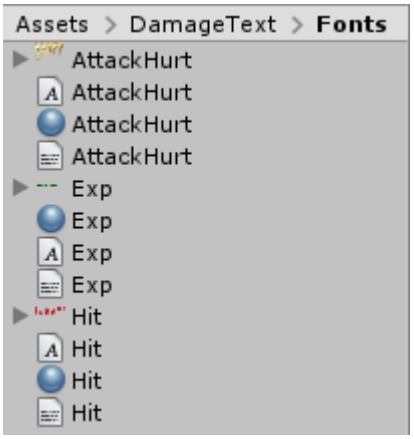
How to use

Steps：

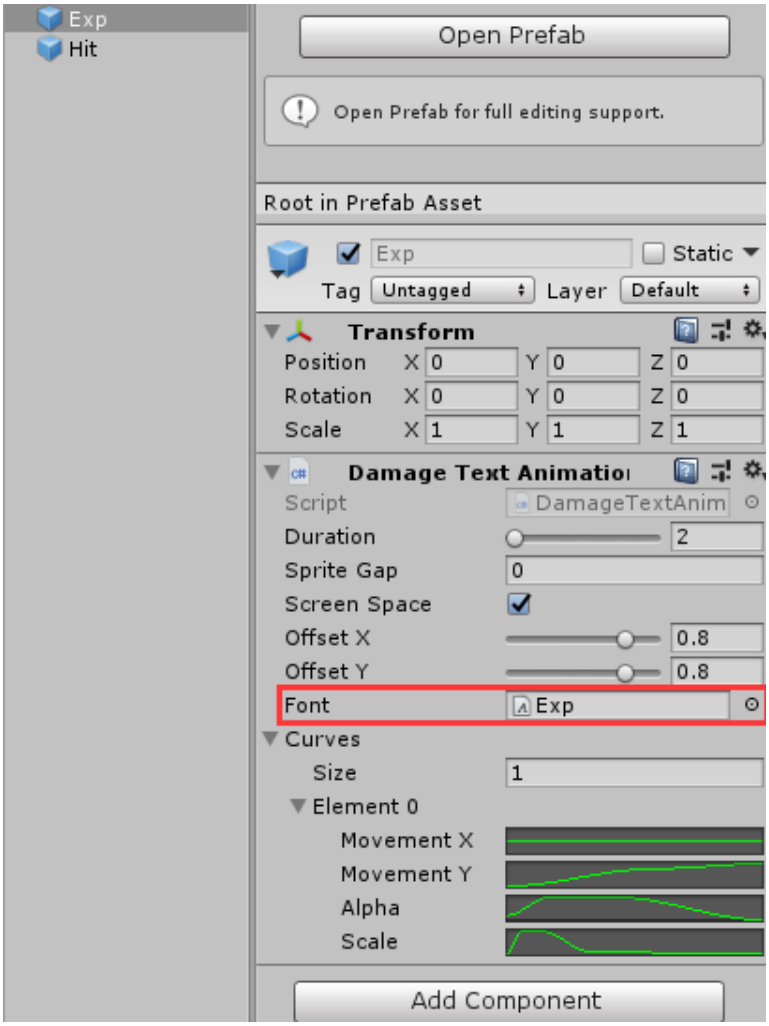
Step 1.

Import your font files.



Step 2.

Edit damage text animation prefabs(DamageTextAnimation.cs)



Duration:

animation time

Sprite Gap:

sprite gap

Screen Space:

Offset X:screen position horizontal offset, range 0 to 1(screen left to right).

Offset Y:screen position vertical offset, range 0 to 1(screen bottom to top).

Font:

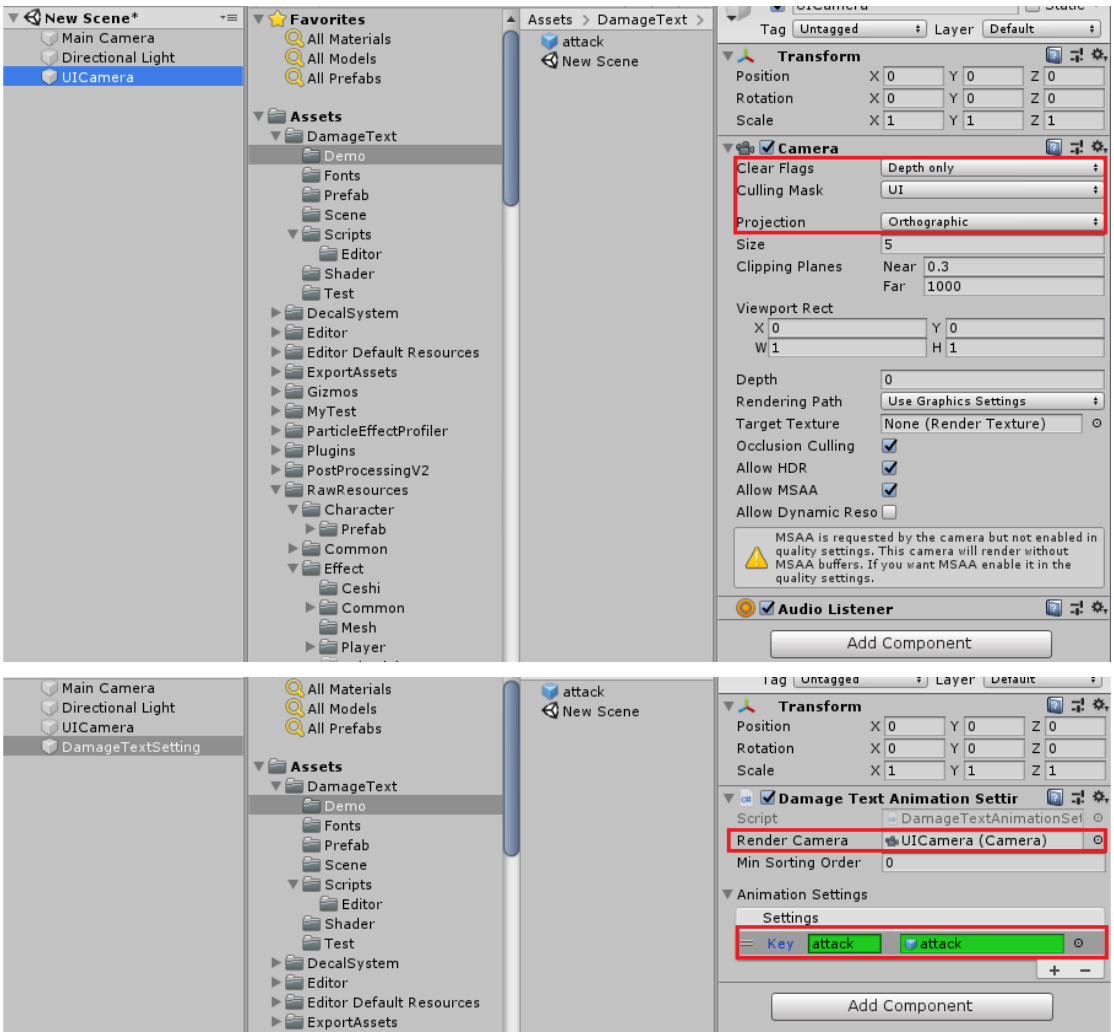
font file(**can not be null!!!**)

Curves:

animation curves include offset, alpha, scale.support multi group curves(random choice)

Step 3.

Set render camera and animation key(DamageTextAnimationSetting.cs)



RenderCamera:

Orthographic camera with culling mask UI.

Min Sorting Order:

Customized min sorting order, see Section Render Order for detail

AnimationSettings

Customized key words and animation prefabs in step2.

Step 4.

Call API to show your content

```
{
    DamageTextManager.Instance.ShowDamageText("attack", content.ToString(), tf);
}
```

Params:

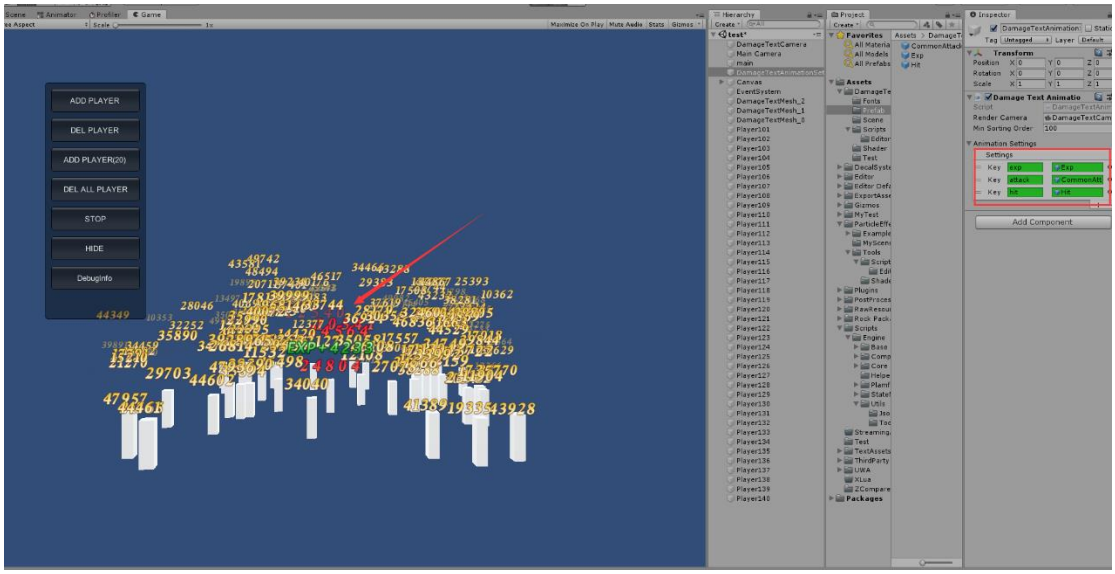
key:Customized key in step 3.

content:damage text string content.

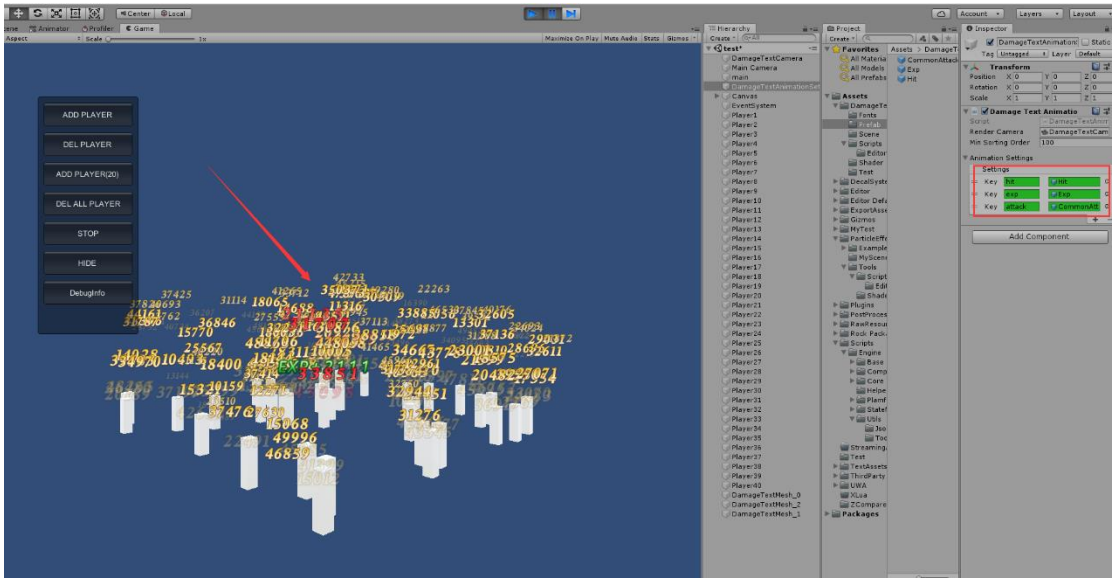
transform:object transform(if screen space, this can be null)

Render Order:

The order in the Animation Settings determines the rendering order for the different type of damage text.

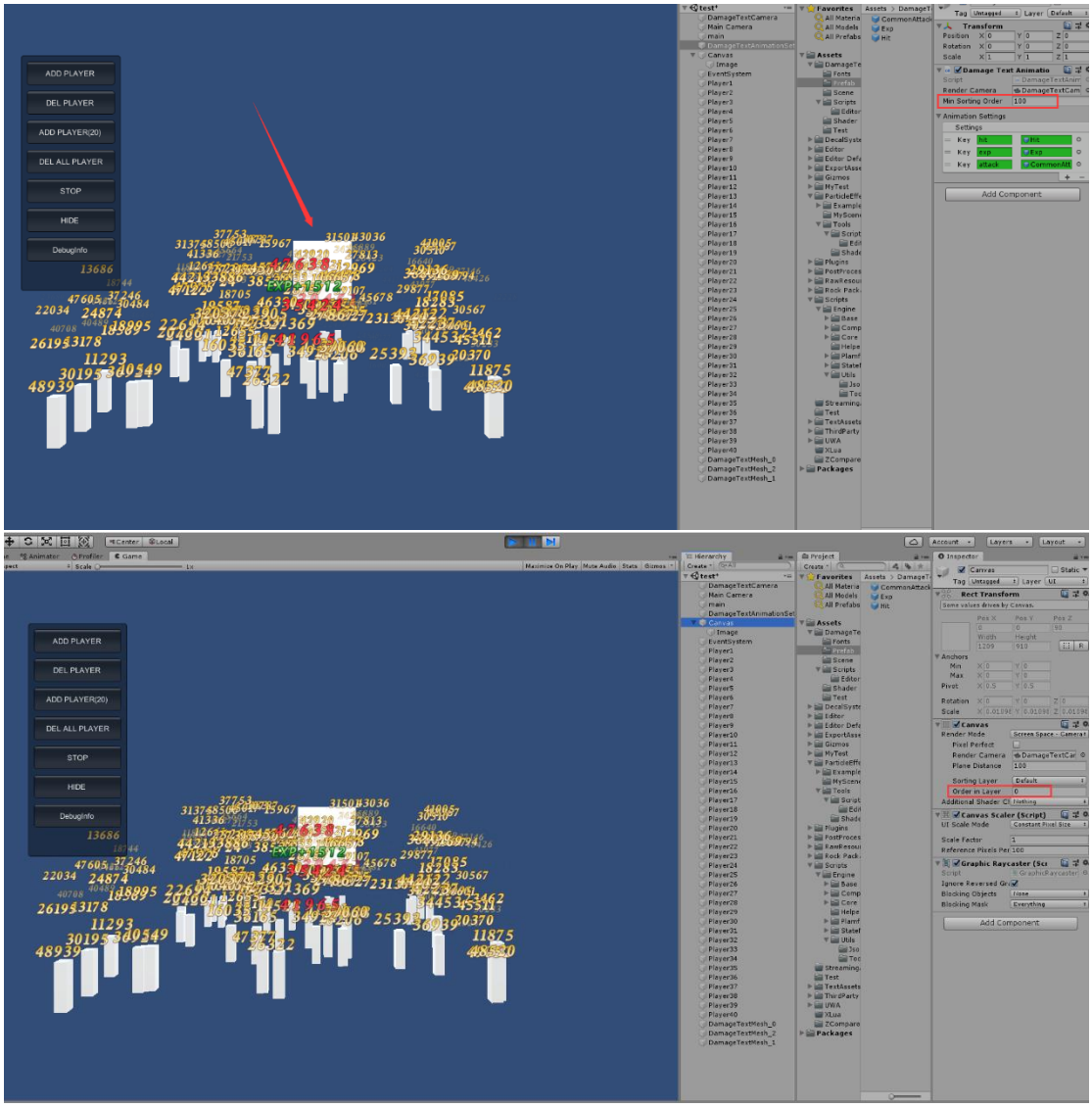


exp > attack > hit

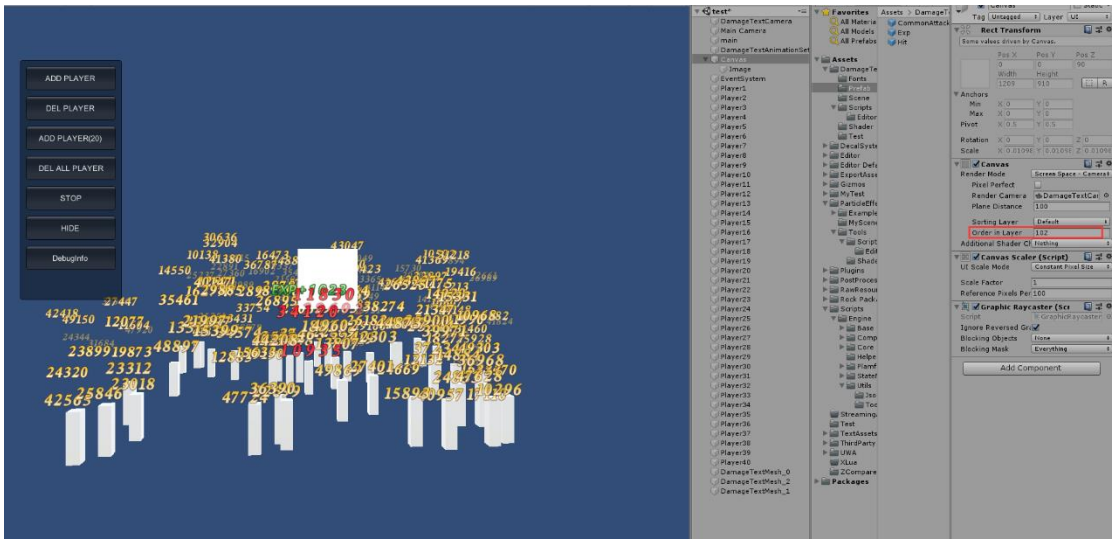


hit > exp > attack

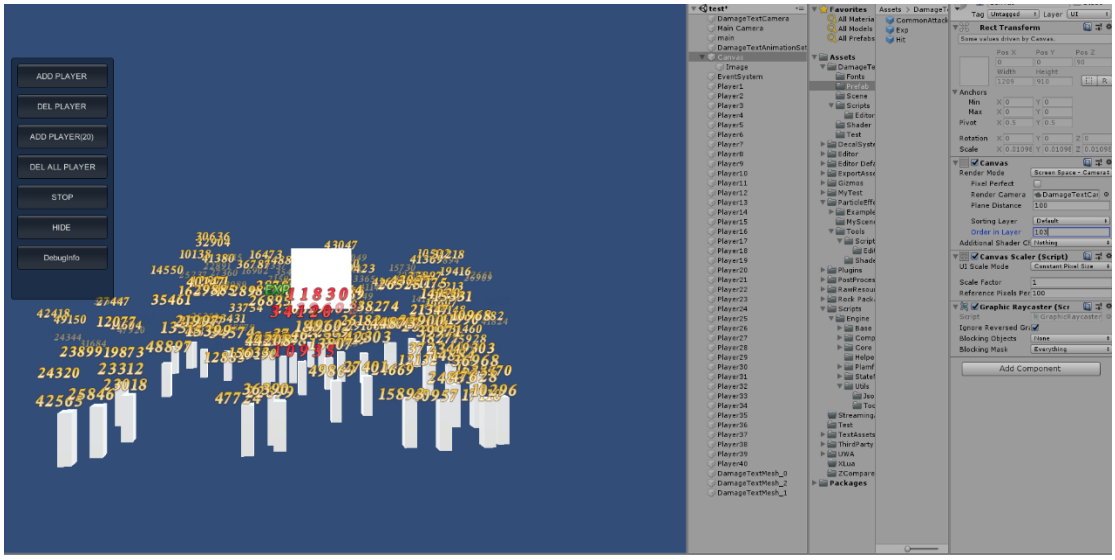
The Min Sortting Order determines the rendering order between the damage text and the UI.



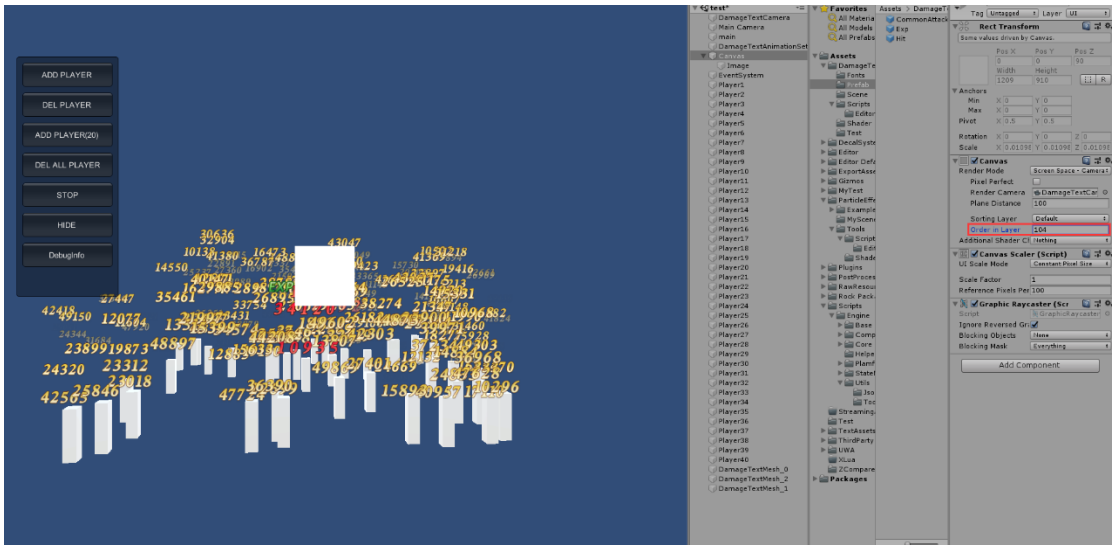
hit > exp > attack > ui texture



hit > exp > ui texture > attack



hit > ui texture > exp > attack



ui texture > hit > exp > attack