

在之前的学习中，我们已经添加了一些基本的音效。在本课的内容中，我们将添加了更多的音效。

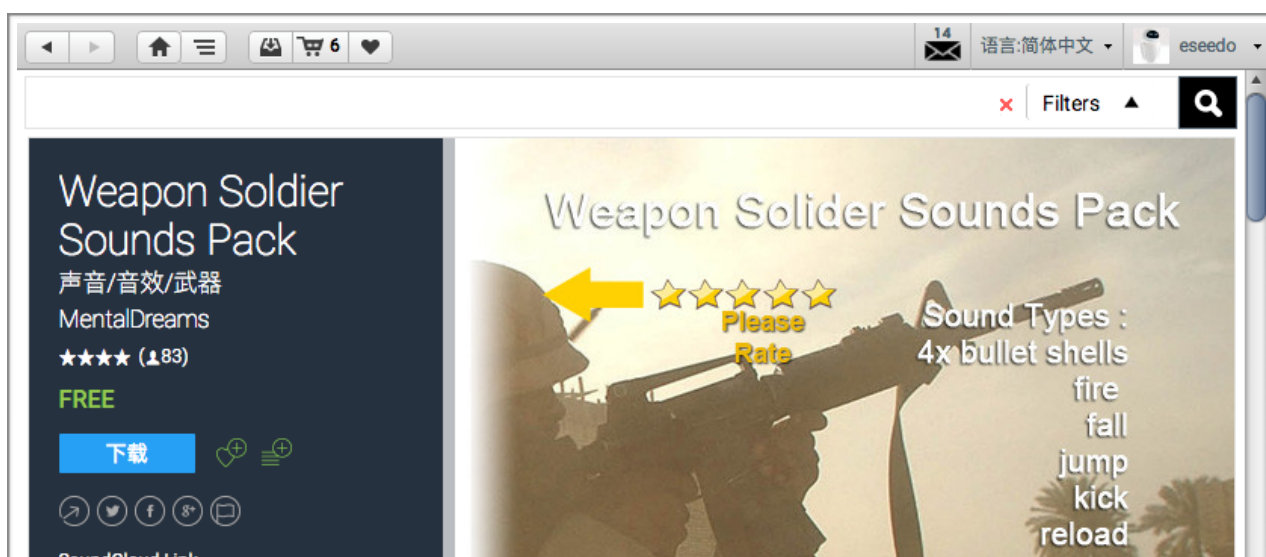
当然，我们还是从Unity的Asset Store中获取免费资源。

打开Unity，切换到Asset Store视图，然后搜索blood sound，然后找到如下图所示的这个音效资源。



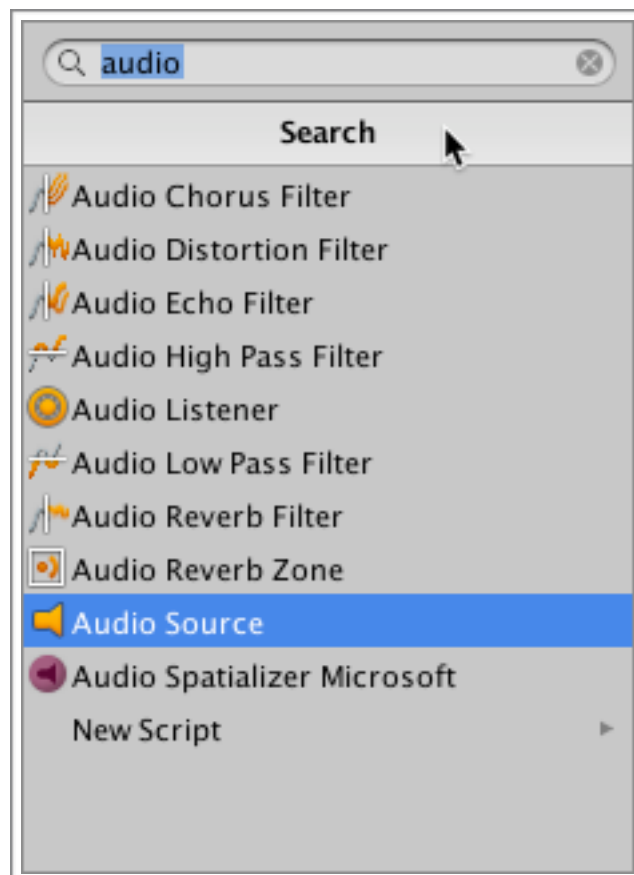
将其下载并导入到项目中，并在Project视图中将导入的BloodSFX资源拖到Sounds文件夹中。

接下来在Asset Store中搜索fire sound，选择FREE ONLY，然后找到如下图所示的音效资源：

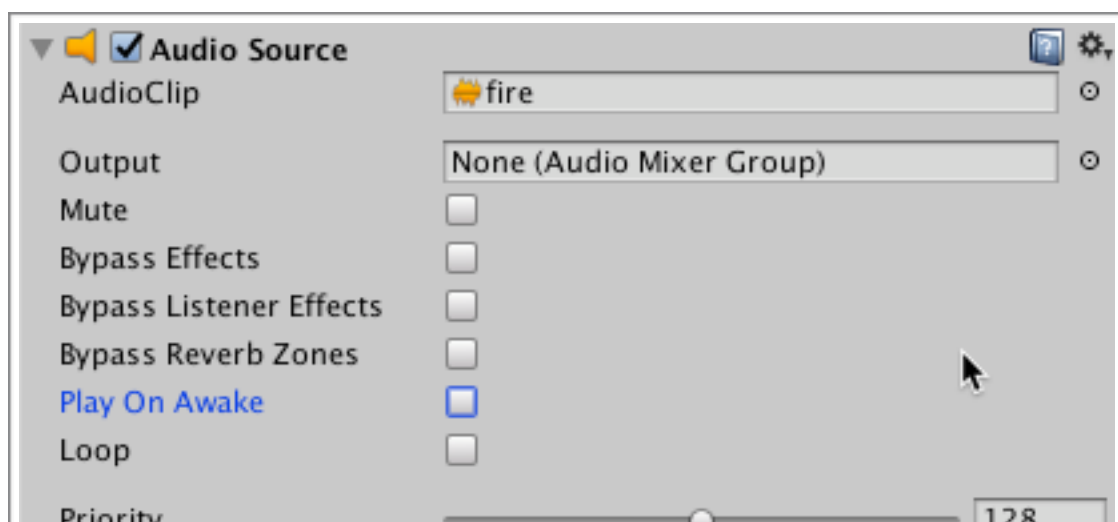


将其下载并导入到项目中，并在Project视图中将对应的资源包MentalDreamsAssets拖动到Sounds中。

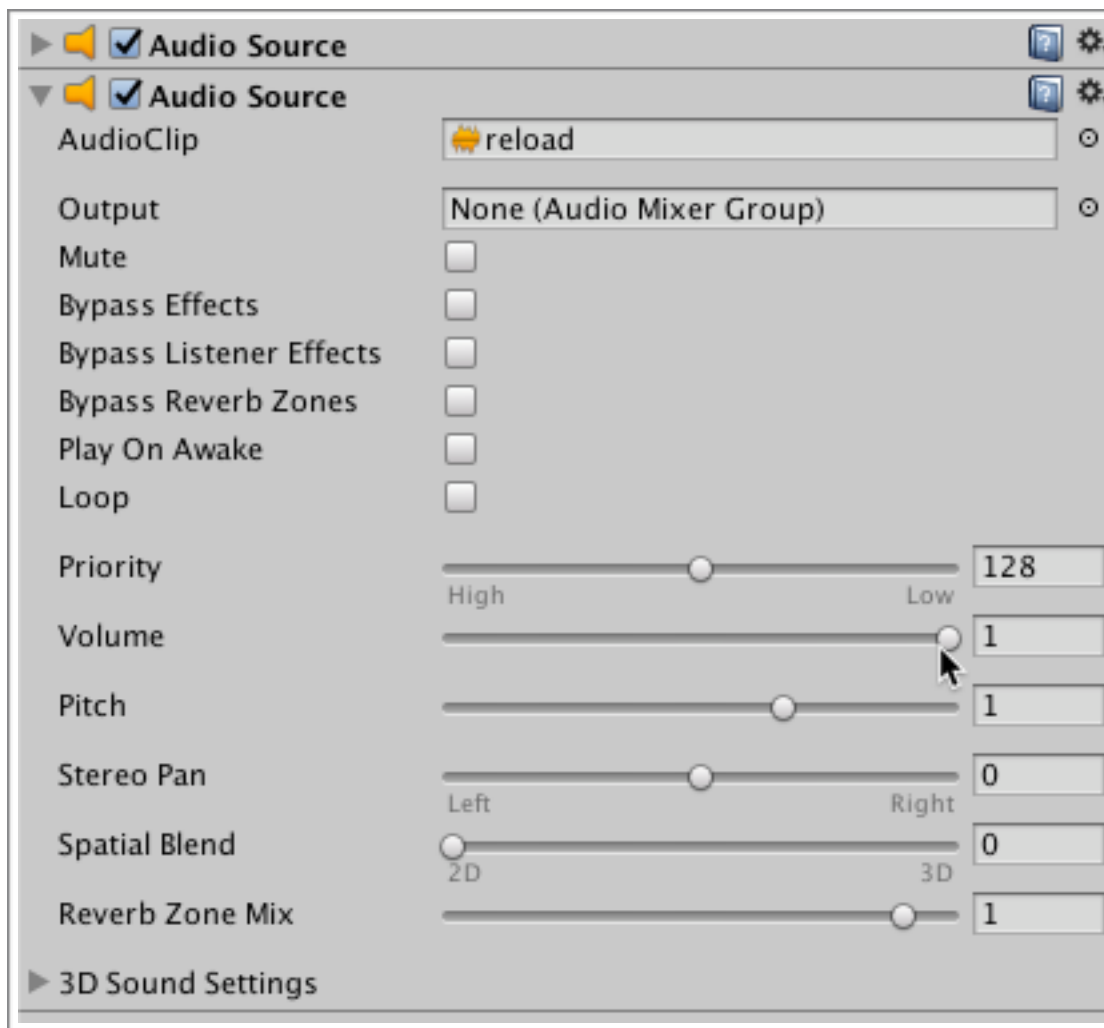
在Hierarchy视图中选择CameraParent下的Main Camera的子对象weapon1，然后在Inspector视图中点击Add Component，选择Audio Source。



然后在Audio Source组件的AudioClip属性处，选择Project视图中的Assets-Sounds-MentalDreamAssets-SoliderSoundsPack中的fire音效，此外取消勾选Play On Awake，如图所示。



设置完成后再次点击Add Component，并添加一个新的Audio Source组件。将AudioClip属性设置为reload音效，同样要取消勾选Play On Awake，如图所示。



接下来在Project视图的_Scripts文件夹中找到并打开ShootEnemy.cs文件，并更改代码如下：

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
//import namespace
using UnityEngine.UI;

public class ShootEnemy : MonoBehaviour {

    //创建到Button对象的引用
    public Button shootBtn;
```

```

//创建到主摄像机的引用
public Camera fpsCam;

//设置敌人每次受到伤害的数值
public float damage = 10f;

//敌人受伤的粒子特效
public GameObject bloodEffect;

//攻击的粒子特效

public GameObject shootingEffect;

//添加的攻击力度
public int forceAdd = 300;

//1.定义两个音源对象
AudioSource shootSound;
AudioSource reloadSound;

// Use this for initialization
void Start () {

    //Debug.Log ("Activated!");
    //添加按钮的响应事件
    shootBtn.onClick.AddListener (OnShoot);

    //2.获取音源组件
    AudioSource[] audios = GetComponents<AudioSource>();

    //3.设置音源
    shootSound = audios [0];
    reloadSound = audios [1];
}

public void OnShoot(){

    //4.播放音效
    shootSound.Play();

    Debug.Log ("shooting!");
}

```

```

        //定义一个RaycastHit类型变量，用于保存检测信息
        RaycastHit hit;

        //判断是否检测到命中敌人
        if (Physics.Raycast (fpsCam.transform.position,
        fpsCam.transform.forward, out hit)) {

            //获取所受攻击的敌人
            Enemy target =
            hit.transform.GetComponent<Enemy>();

            //destroy enemy

            if (target != null) {

                //instantiate blood effect

                target.TakeDamage (damage);
                //创建敌人受伤的粒子特效

                GameObject bloodBurst = Instantiate
                (bloodEffect, hit.point, Quaternion.LookRotation (hit.normal));

                //0.2秒后销毁粒子特效

                Destroy (bloodBurst, 0.2f);
            } else {
                //load shooting effect

                //如果没有击中敌人，则创建攻击时的粒子特效

                GameObject shootingGo = Instantiate
                (shootingEffect, hit.point, Quaternion.LookRotation
                (hit.normal));

                //0.2秒后销毁粒子特效
                Destroy (shootingGo,0.2f);
            }

            //攻击敌人时添加一个额外的冲击力

```

```

        if (hit.rigidbody != null) {
            hit.rigidbody.AddForce (-hit.normal *
forceAdd);
        }

        //输出所命中的对象名称
        Debug.Log (hit.transform.name);

    }

}
}

```

还是按照注释行的数字编号来解释一下：

- 1.定义了两个音源对象，分别用在攻击敌人和重新装弹上
- 2.使用数组来获取音源组件，注意这里用的是GetComponents，而不是GetComponent，因而获取的是一个数组，而非单一组件对象。
- 3.分别设置两个音源对象
- 4.在射击时播放对应的音效。

好了，攻击敌人的音效已经添加了，接下来我们还将给敌人本身添加点音效。

在Unity编辑器的Project视图中找到_Prefabs中的zombieEnemy预设体，在Inspector视图中点击Add Component，并添加一个新的Audio Source组件。

将AudioClip属性设置为Project视图中Assets/Sounds/BloodSFX/Splat中的bloodfx1音效（或者其它你个人喜好的），同时注意取消勾选Play On Awake。

然后打开Enemy.cs这个文件，并更改其中的代码如下：

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Enemy : MonoBehaviour {

    //设置敌人的生命值

```

```

public float health = 30f;

//1.定义敌人受伤的音效
AudioSource bloodSound;

// Use this for initialization
void Start () {

    //2.获取音源
    AudioSource[] audios = GetComponents<AudioSource>();

    //3.设置音效
    bloodSound = audios[1];
}

//敌人受到伤害后的处理
public void TakeDamage(float damage){

    //4.播放音效
    bloodSound.Play();

    //敌人生命值减少特定的数值
    health -= damage;

    //输出敌人生命值
    print (health);

    //当敌人生命值变为0的时候，就死亡
    if (health <= 0) {

        //Enemy Die

        Die ();
    }
}

//敌人死亡
void Die(){

    //5.在1秒钟后销毁敌人对象
    Destroy (gameObject, 1f);
}

```

```
}
```

这里所添加的几行代码跟刚才的完全类似，其作用如下：

1. 定义敌人受伤的音效
2. 获取音源组件的数组
3. 设置敌人受伤的音效
4. 播放音效。

回到Unity编辑器，点击工具栏上的Play按钮，就可以预览游戏效果了。

可以看到从视觉上没有什么变化，只是增加了跟敌人相关的特定音效。