**AB包**

**什么是AB包？**

AB包全名AssetBundle， 通过将资源分布在不同的AB包中可以最大程度地减少 运行时的内存压力，并且可以有选择地加载内容, 动态下载和加载。

AB包与Resources 的比较 AB包

1： 资源可以分布在多个包中 2： 储存的位置自定义灵活

3： 压缩的方式灵活

4： 支持后期进行动态更新

**打AB包的时候需要注意什么？**

1：获取文件夹下所有的资源的时候

Resources

所有资源达打成一个打包

必须放在Resources文件夹下 全部压缩成二进制

打包后资源只读无法动态更改

有可能会是

（D:/UnityWork/ab/Assets/Resources\\Capsule.prefab ）这样是获取不到依赖项的

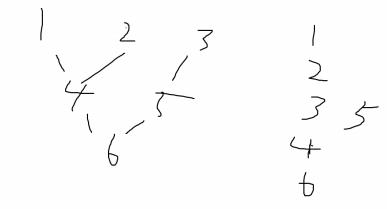
需要更改成（Assets/Resources/Capsule.prefab（ "Assets" + Path.GetFullPath(list[i]).Replace(Path.GetFullPath(Application.dataPath), "").Replace('\\', '/');））

2：不是所有的后缀名文件都可以打包 下面这些就不可以打包 要把这些给删

除掉，然后进行打包



**依赖项**



把依赖项打一起（ 注意：有的材质球可能会被重复利用 把重复利用的打成一 个包），或者 一个依赖项一个包（注意：有的模型上面会出现名称一样但是后 缀名不一样 如：



) 这样的话还是打一个包会报错 可以把名称一样后缀名不一样的打成一个包 或者 找美术改一下名称

AB包加载

从AB包加载一个物体的时候 要确保它的依赖项也被加载 如：只加载了一个 Cube没加载依赖项的话会造成材质丢失

卸载AB包

加载的资源应该在不再需要时及时卸载 ，以释放内存。使用 AssetBundle.Unload方法可以卸载AB包及其加载的资产。

通过引用计数来判断他是否被人依赖 ，如果引用计数为零的情况下说明该对象没有被 依赖 所有就卸载掉

AssetBundle.Unload（false） :只卸载包 AssetBundle.Unload（true） :卸载包和Obj

AssetBundle.Unload（true） : 除了卸载压缩文件数据外 ，还会销毁所有从该 Bundle中加载的对象 ，这可能会导致场景中引用这些对象的游戏对象出现缺 失。 所有加载物体和依赖项的时候会写一个引用计数 当 引用计数为0的时候 调用 AssetBundle.Unload（true）

1 ：你刚开始第一次就进入项目的时候 ，你需要把 ， S目录给拷到P目录 ， S目录就是一 般放什么

S目录放所有的资源 刚进入游戏 直接把S目录Copy到P

S目录放版本文件 ，版本文件对比 去服务器下载资源 2:循环依赖

A-->B B-->C C-->A