



GameManager

继承关系：继承自NetworkBehaviour

重要属性：

```
public static GameManager Instance;  
public Dictionary<ulong, PlayerInfoData> playerInfoDataDict { get; private set; }  
public UnityEvent OnStartGame;
```

OnNetworkSpawn

1. 调用基类OnNetworkSpawn
2. 向NetworkManager.SceneManager.OnLoadEventCompleted中添加一个回调方法，该方法在所有玩家加载完成事件后进行回调

OnLoadEventCompleted

1. 判断当前场景是否为GameScene，如果是则执行OnStartGame回调方法。在PlayerInit脚本中会将一个方法添加到OnStartGame里

Start

1. 为单例模式Instance赋值
2. 使用 `DontDestroyOnLoad` 方法保证在切换场景时使用该方法的物体会被保留
3. 切换至StartScene

加载场景

1. 注意在网络上加载场景跟SceneManager.LoadScene是有区别的，这里使用NetworkManager.SceneManager.LoadScene加载对应场景

开始游戏

1. 记录传入的playerInfoDataDict
2. <在所有客户端上更新玩家信息>

在所有客户端上更新玩家信息

1. 遍历所有的playerInfo，更新玩家信息<**UpdatePlayerInfoClientRpc**>

UpdatePlayerInfoClientRpc [ClientRpc]

注意：Network中做主机的玩家会同时具有Server和Client

1. 如果当前是Server则返回，避免重复更新玩家数据
2. 查看当前Client中GameManager palyerInfoDataDict中是否包含传入的玩家Id，如果没有则添加到字典中，如果有则更新字典中的value