1) Open   Your Create and empty Project and Frist Import Playfab SdK
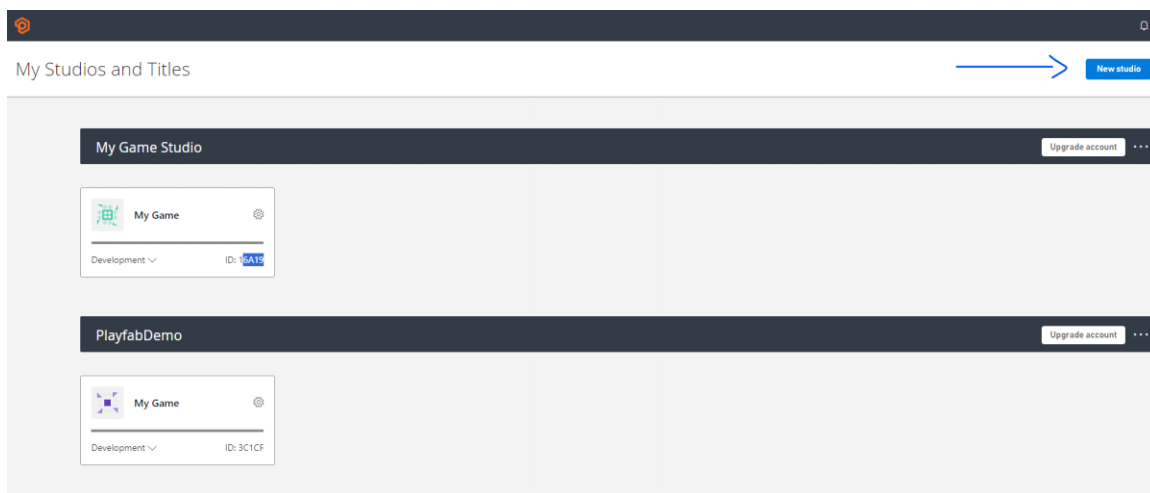
https://learn.microsoft.com/en-us/gaming/playfab/sdks/unity3d/installing-unity3d-sdk
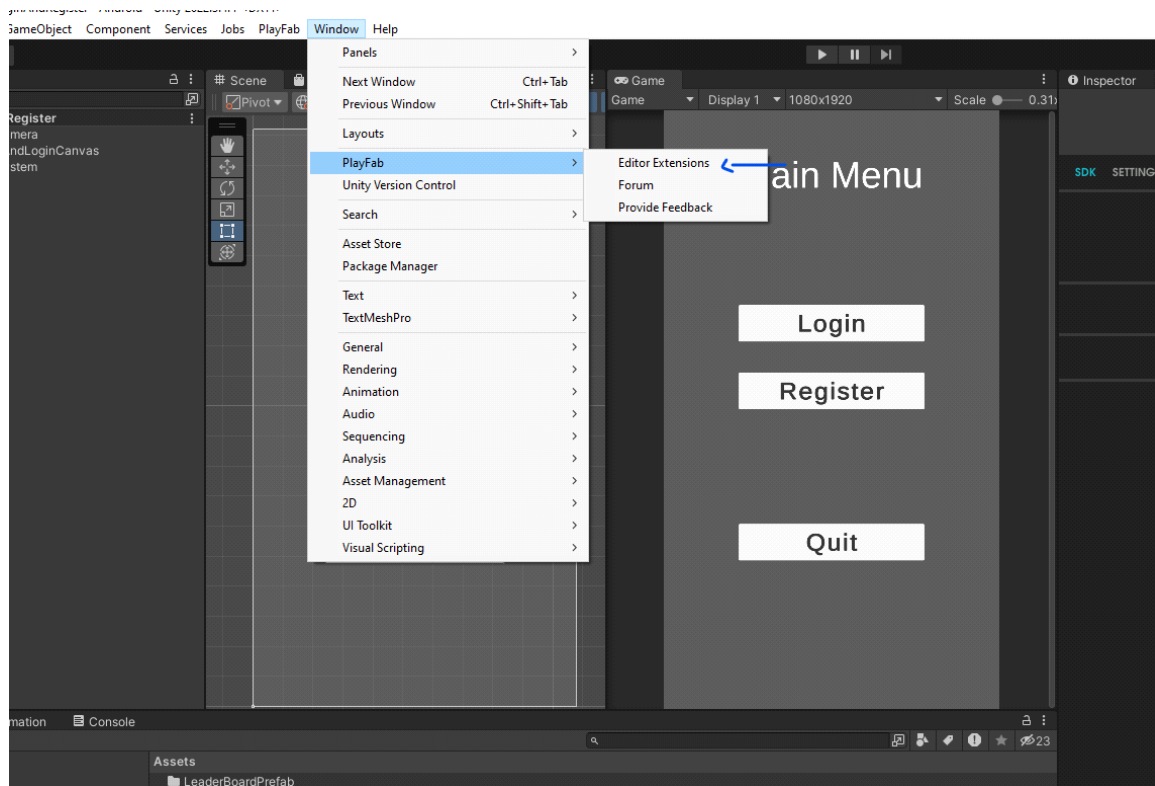
2) json.net Packege import Your Project

3) Newtonsofrt Packege import Your Project

4) Open Chrome and go to Playfab Dashboard Login with your email id

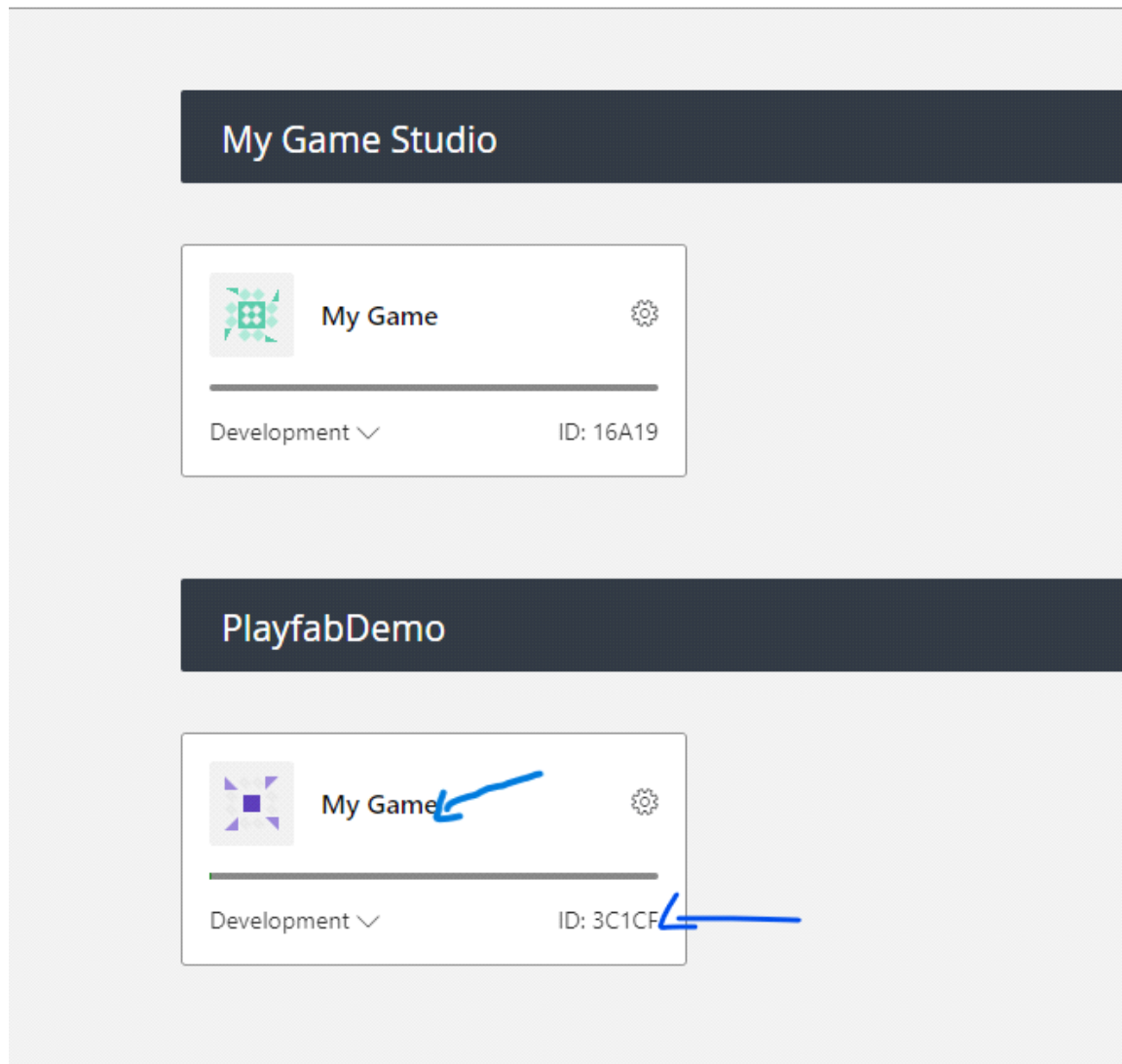5) After loging Create a New studio in playfab deshboard

6) go to your projrct and open window tab and find playfab ans playfab in side Palfab Extensisons.
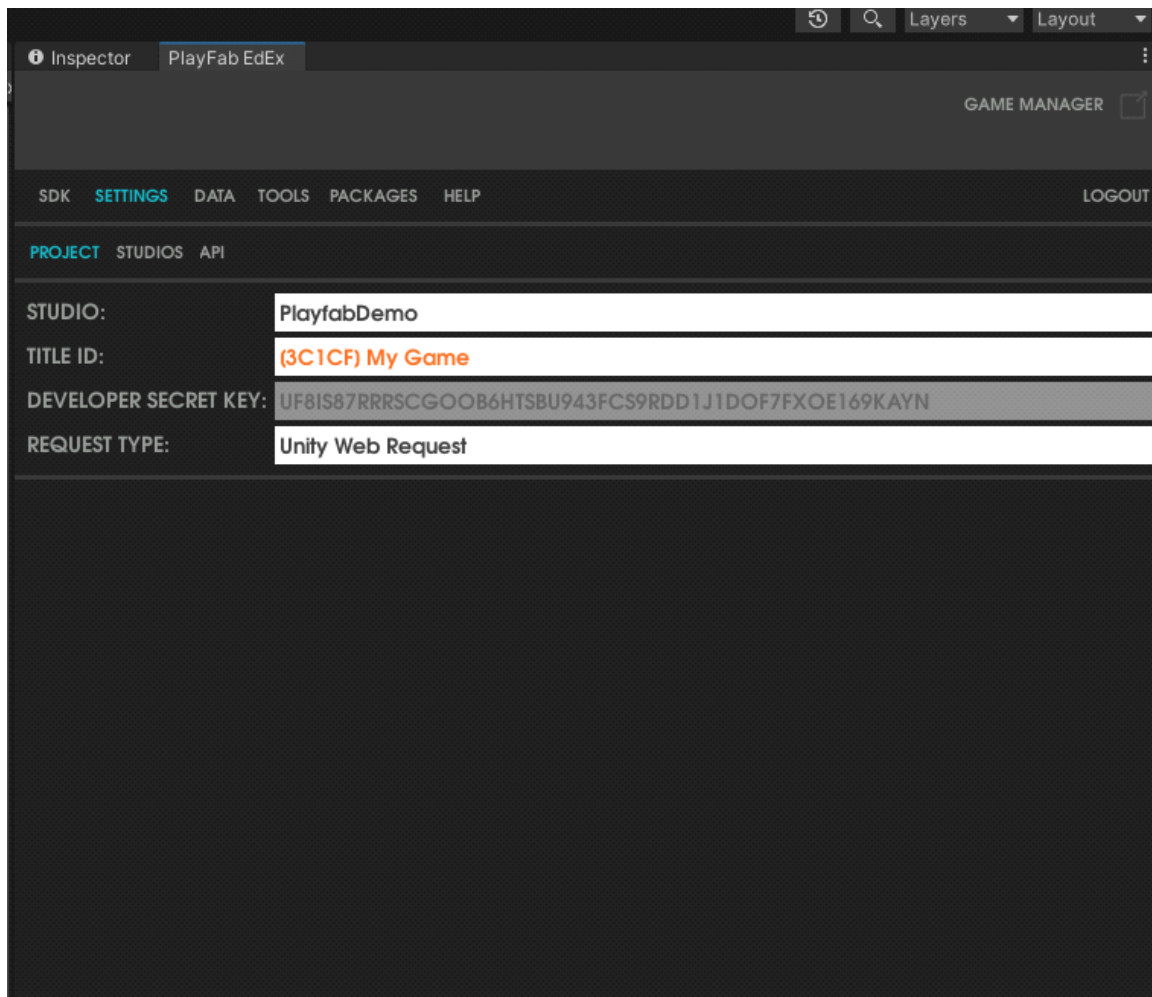
7) Open Playfab Extensisons tab inside open setting in side fillup   data.

        1) studio na

        2) Titel Id

        3) Request type in select = unity web request

My Studios and Titles

My Game Studio

My Game ⚙

Development ⌄   ID: 16A19

PlayfabDemo

My Game ⚙

Development ⌄   ID: 3C1CF

This Studio Name And Tital Id Enter this playfab Extensisons.

8) Add Cloud Script in Playfab Deshboard in side click to your studio and find Automation in side Revisions.

Development ∨ 49/100K

Title Overview

BUILD
- Players
- Multiplayer
- Groups
- Automation
- Add-ons

ENGAGE
- Economy
- Leaderboards
- Content

ANALYZE
- Dashboards
- Data
- Experiments

Cloud Script   Rewarded Ads   Rules   Scheduled Tasks

Revisions

Cloud Script > Revisions (Legacy)

Functions   Overview (Legacy)   Revisions (Legacy)

Revision 11 (live) ∨

```
1    handlers.setupWinLossData = function(){
2        var WinLossDictionary ={
3            "Wins" : 0,
4            "Losses" :0
5        };
6
7        var playerDataToUpdate = JSON.stringify(WinLossDictionary);
8
9        var updateDataRequest = server.UpdateUserData({
10           PlayFabId : currentPlayerId,
11           Data:{
12               "PlayerData" : playerDataToUpdate
13           }
14       });
15
16       return {success: true, message: "Player Win.Loss Data Ready"};
17   }
18
19   handlers.updateLevelOneRecordTimeStat = function(args, context){
20
21       var timeInSeconds = args.timeInSeconds;
22
23           var updateWinRecord = server.UpdatePlayerStatistics({
24           PlayFabId: currentPlayerId,
25           Statistics: [{
26               StatisticName: "Level 1 Record Times",
27               Value: timeInSeconds
28           }]
29
30       });
31
32       return {success: true, message: "Level One Time Record was Updated"};
33   }
34
35   handlers.updateLevelTwoRecordTimeStat = function(args, context){
36
37       var timeInSeconds = args.timeInSeconds;
```