

# AppMetr Unity Plugin

## Интеграция плагина

1. Импортируете плагин **\*AppmetrUnityPlugin.unitypackage\*** в проект (двойным кликом по файлу или через меню *\*Assets/Import Package/Custom Package...\**).
2. Для прослушивания команд с сервера добавляете на сцену объект с именем *AppMetrCommandListener*, на объект добавляете скрипт *AppMetrCommandListener*. Подписываетесь на события *AppMetrCommandListener.OnCommand*.
3. Перед вызовом команд плагина, необходимо в любом месте кода вызвать *AppMetr.Setup(token)*.

## Пример кода

```
using UnityEngine;
using System.Collections;

public class AppMetrManager : MonoBehaviour {

    void Awake() {
        AppMetrCommandListener.OnCommand += HandleAppMetrCommand;
    }

    void OnDisable() {
        AppMetrCommandListener.OnCommand -= HandleAppMetrOnCommand;
    }

    void Start() {
        AppMetr.Setup("00000000-0000-0000-0000-000000000000");
    }

    public void HandleAppMetrCommand(string command) {
        Debug.Log("AppMetrManager: HandleAppMetrCommand\n" + command);
    }
}
```

# AppMetr API

## Класс AppMetr

Это основной класс, предоставляющий доступ к функциям библиотеки.

### Методы:

#### - Setup

Инициализирует библиотеку

Параметры:

*token* - токен приложения

#### - TrackSession

Регистрирует запуск приложения

Параметры:

*properties* - дополнительные параметры (сведения о пользователе, например, дата регистрации, имя, возраст)

#### - TrackLevel

Регистрирует достижения нового уровня игроком

Параметры:

*level* - достигнутый уровень

*properties* - дополнительные параметры

#### - TrackEvent

Регистрирует игровое событие

Параметры:

*eventName* - название события

*properties* - дополнительные параметры

#### - TrackPayment

Регистрирует совершение платежа

Параметры:

*payment* - сведения о выполненной покупке

*properties* - дополнительные параметры

- AttachProperties

Добавление параметров

Параметры:

*properties* - параметры

- TrackExperimentStart

Регистрирует начало эксперимента

Параметры:

*experiment* - имя эксперимента

*groupId* - группа

- TrackExperimentEnd

Регистрирует конец эксперимента

Параметры:

*experiment* - имя эксперимента

- Identify

Идентификация пользователя

Параметры:

*userId* - идентификатор пользователя

- Flush

Форсированная отправка событий на сервер

## Класс AppMetrCommandListener

Этот класс предоставляет доступ к событию получения команды от сервера.

Пример использования:

```
AppMetrCommandListener.OnCommand += HandleOnCommand;  
public void HandleOnCommand(string command) {  
    Debug.Log("HandleOnCommand:\n" + command);  
}
```

Пример использования плагина можно посмотреть в  
***\*Plugins/Appmetr/Sample/AppMetrSample.cs\****.

