

# AppsFlyer接入文档

## 1. 接口

添加 `quicksdk` namespace:

```
using quicksdk;
```

使用以下方法获取 SDK 实例。所有 SDK 接口都通过该实例调用：

```
QuickSDK.getInstance();
```

## 1. AppsFlyer初始化

方法:

```
public void AppsFlyerInit(string devkey, string appid, bool issandbox = false)
```

参数:

- `DEV_KEY`: 从 [AppsFlyer HQ](#) 获取
- `APP_ID`: AppsFlyer HQ 上的应用程序 ID (不包括 `nativepc-` 前缀)
- `bool issandbox`: 是否激活沙箱模式。默认为“false”。此选项用于调试。在沙盒模式下, AppsFlyer 面板不会显示数据。

示例:

```
QuickSDK.getInstance().AppsFlyerInit("xxxx", "xxxx", false);
```

所有后续的 API 调用都必须在初始化后进行。

## 2. Start

此方法向 AppsFlyer 发送首次打开/会话请求。

方法:

```
public void AppsFlyerStart(bool skipFirst = false)
```

参数:

`bool skipFirst`: 决定是否跳过首次打开事件并发送会话事件。默认值为false。如果为 true, 则跳过首次打开事件并发送会话事件

示例:

```
QuickSDK.getInstance().AppsFlyerStart();
```

### 3. Stop

此方法可停止 SDK 运行和与 AppsFlyer 服务器通信。在实现用户选择进入/退出时使用。

方法:

```
public void AppsFlyerStop()
```

示例:

```
// Starting the SDK
QuickSDK.getInstance().AppsFlyerStart();
// ...
// Stopping the SDK, preventing further communication with AppsFlyer
QuickSDK.getInstance().AppsFlyerStop();
```

### 4. LogEvent

此方法接收事件名称和 JSON 对象，并向 AppsFlyer 发送应用内事件。

方法:

```
public void AppsFlyerLogEvent(String event_name,
    Dictionary<string, object> event_parameters,
    Dictionary<string, object> event_custom_parameters = null
)
```

参数:

- `string event_name`: 事件名称.
- `Dictionary<string, object> event_parameters`: 事件参数, 其中包含 [predefined event parameters](#).
- `Dictionary<string, object> event_custom_parameters`: 自定义事件参数, 可为空

示例:

```
// set event name
string event_name = "af_purchase";
// set event values
Dictionary<string, object> event_parameters = new Dictionary<string, object>();
event_parameters.Add("af_currency", "USD");
event_parameters.Add("af_revenue", 12.12);
// send logEvent request
afm.LogEvent(event_name, event_parameters);

// send logEvent request with custom params
Dictionary<string, object> event_custom_parameters = new Dictionary<string, object>();
event_custom_parameters.Add("goodsName", "新人邀约购物日");
QuickSDK.GetInstance().AppsFlyerLogEvent(event_name, event_parameters,
event_custom_parameters);
```

## 5. IsInstallOlderThanDate

方法:

```
public void AppsFlyerIsInstallOlderThanDate(string datestring)
```

参数:

- string datestring: 日期字符串, 格式为 "yyyy-mm-ddThh:mm:ss+hh:mm"。

Usage:

```
// the creation date in this example is "2023-03-23T08:30:00+00:00"
bool newerDate = afm.IsInstallOlderThanDate("2023-06-13T10:00:00+00:00");
bool olderDate = afm.IsInstallOlderThanDate("2023-02-11T10:00:00+00:00");

// will return true
Debug.Log("newerDate:" + (newerDate ? "true" : "false"));
// will return false
Debug.Log("olderDate:" + (olderDate ? "true" : "false"));

// example usage with skipFirst -
// skipping if the install date is NOT older than the given date
bool IsInstallOlderThanDate =
QuickSDK.GetInstance().AppsFlyerIsInstallOlderThanDate("2023-02-11T10:00:00+00:00");
QuickSDK.GetInstance().AppsFlyerStart(!IsInstallOlderThanDate);
```

## 6. SetCustomerId

此方法设置一个客户 ID, 使您能够将您的唯一 ID 与 AppsFlyer 唯一 ID 和其他设备 ID 进行交叉引用。注意: 您只能在调用 `start()` 之前使用此方法。

客户 ID 可在原始数据报告和通过 API 发送的回传中使用。

#### 方法:

```
public void AppsFlyerSetCustomerUserId(string cuid)
```

#### 参数:

- `string cuid`: Custom user id.

#### Usage:

```
QuickSDK.getInstance().AppsFlyerInit("xxxx", "xxxx", false);
QuickSDK.getInstance().AppsFlyerSetCustomerUserId("15667737-366d-4994-ac8b-653fe6b2be4a");
QuickSDK.getInstance().AppsFlyerStart();
```

## 7. SetSharingFilterForPartners

通过此方法，您可以配置 SDK 应将哪些合作伙伴排除在数据共享之外。使用此方法排除的合作伙伴将不会通过回传、应用程序接口、原始数据报告或任何其他方式接收数据。

#### 方法:

```
public void AppsFlyerSetSharingFilterForPartners(List<string> sharingFilter)
```

#### 参数:

- `List<string> sharingFilter`: a list of partners to filter. For 示例: `new List<string>() {"partner1_int", "partner2_int"};`

#### Usage:

```
QuickSDK.getInstance().AppsFlyerInit("xxxx", "xxxx", false);
// set the sharing filter
var sharingFilter = new List<string>() {"partner1_int", "partner2_int"};
QuickSDK.getInstance().AppsFlyerSetSharingFilterForPartners(sharingFilter);
QuickSDK.getInstance().AppsFlyerStart();
```

## 8. GetAppsFlyerUID

获取 AppsFlyer 的唯一设备 ID。安装应用程序时，SDK 会生成 AppsFlyer 的唯一设备 ID。启动 SDK 时，该 ID 将被记录为首次安装应用程序的 ID。

#### 方法:

```
public string GetAppsFlyerUID()
```

#### 示例:

```
QuickSDK.getInstance().GetAppsFlyerUID();
```

