

# FACEBOOK

View online: <https://www.construct.net/en/make-games/manuals/construct-3/plugin-reference/facebook>

The **Facebook** object allows you to integrate your game with Facebook. It can be used in any web-hosted game, not just in Facebook games. For example, you could have a game running on your own server. The Facebook object still allows you to integrate with user's Facebook accounts even when the user is playing the game on your website. However, it is still necessary to submit the app to Facebook so you have an App ID. Apps can be submitted to Facebook via [Facebook Developers](#).

*The Facebook plugin is **not compatible with Instant Games**. Use the [Instant Games](#) plugin instead.*

The Facebook object makes asynchronous requests. This means the game keeps running while operations which may take several seconds (like submitting a score) complete in the background. Therefore, the Facebook object works around the principle of an action which starts a task (like *Publish score*), which then triggers a condition when it has completed (like *On score submitted*). These may be a few seconds apart and the game continues to run in between, rather than freezing the game while it waits for completion. As with most asynchronous requests, actions may complete in a different order to that which they were made.

## Readiness

The Facebook object takes a moment to load on startup. Do not attempt to use the Facebook object before *On ready* has triggered or *Is ready* is true, otherwise the actions will be ignored.

You must enter the **App ID** for the Facebook object to successfully load, even in testing. If you don't enter it, the object will never become ready. This is because Facebook prevent you using the API unless you have a real App ID to give it.

## Logging in

The user must successfully log in with their Facebook account before they can use any other features of the Facebook object, such as posting to the user's wall. This is done with the *Log in* action, and since login can take a few seconds you must wait until *On user logged in* triggers before using any other features.

## Facebook Playable Ads

This plugin also provides the *On CTA click* action for Facebook Playable Ads. This action can be used independently of the other features, i.e. without having to log in first, or provide an app ID or any other properties.

## Facebook Properties

---

### App ID

The App ID given for your app in the Facebook Developers section.

---

### App Secret

This is only necessary if you plan to use the Scores API. Otherwise do not enter the app secret. The app secret can be found in the Facebook Developers section for your app and is currently necessary to enable the Scores API.

## Facebook conditions

---

### Is ready

True if the Facebook object is ready to be used. When false, the Facebook object is still loading the necessary scripts and all actions will be ignored.

---

### Is user logged in

True if the user is currently logged in with their Facebook account.

---

### On name available

Triggered shortly after *On user logged in*, when the expressions to get the user's name are available.

---

### On ready

Triggered when the Facebook object is ready to be used. Before this triggers, the Facebook object is still loading the necessary scripts and all actions will be ignored.

---

### On user logged in

Triggered when the user has successfully logged in. It also triggers on startup if the user has previously logged in, and they have returned to the app with a remembered login. This also triggers at the start of each layout throughout the game while the user is logged in.

---

### On user logged out

Triggered if the user logs out during the game.

---

### On hi-score

Triggers once for each hi-score after the *Request hi-score board* action. The *HiscoreName*, *HiscoreRank* and *Score* expressions contain the current hi-score information.

---

### On score submitted

Triggers when the *Publish score* action has successfully completed.

---

### On user top score available

Triggers when the *Request user top score* action has successfully completed. The *Score* expression has the user's top score.

---

## Facebook actions

---

### Log in

Open a popup window requesting the user to log in. This can only be called in a user-input event, such as a key press, button push, mouse click or touch screen tap. The user may cancel at the log in screen so logging in is not guaranteed to be successful. You must provide a comma-separated list of permissions to request. Some features of the Facebook plugin depend on the user approving certain permissions. Do not request permissions that your application does not need. You can find documentation on the available permissions in the [Facebook developer's guide](#).

---

### Log out

Log the current user out from Facebook.

---

### On CTA click

**For Facebook Playable Ads.** In a Playable Ad, use this action when the user interacts with the call-to-action (CTA) in the ad. This action can be used independently of the rest of the plugin's features.

---

### Prompt to share link

Open a dialog prompting the current user to share a URL of your choosing on their wall. This does not require any permissions, but the user is free to cancel the action.

---

### Prompt to share this app

Open a dialog prompting the current user to share the URL to the current Facebook app on their wall. This does not require any permissions, but the user is free to cancel the action.

---

### Prompt wall post

Open a dialog prompting the current user to make a wall post. No content or links is added - it is an empty text box for the user to type anything they like. This does not require any permissions, but the user is free to cancel the action.

---

### **Publish link**

Automatically publish a URL to the user's wall without any prompt. Descriptions and a thumbnail image can optionally be provided. This requires that the user has logged in with the *Publish to stream* permission.

---

### **Publish wall post**

Automatically publish a message to the user's wall without any prompt. This requires that the user has logged in with the *Publish to stream* permission.

---

### **Publish score**

Publish a numerical score to the hi-score board for the game. This requires that the user has logged in with the *Publish scores* permission. When the action completes, it triggers *On score submitted*.

---

### **Request hi-score board**

Request the game's hi-scores. This requires that the user has logged in with the *Publish scores* permission. This action will repeatedly trigger *On hi-score* once per score on the hi-score board. It is up to you to find a way to display these scores; appending each hi-score to a text object is a simple way to display them.

---

### **Request user top score**

Request the current logged in user's top score for this game. This requires that the user has logged in with the *Publish scores* permission. When the action completes, it triggers *On user top score available*.

---

## **Facebook expressions**

---

### **FirstName**

The current user's first name. This is only set after *On name available* triggers.

---

### **FullName**

The current user's full name. This is only set after *On name available* triggers.

---

### **LastName**

The current user's full name. This is only set after *On name available* triggers.

---

### **UserIDStr**

The current user's ID, which can be used to distinguish different users with the same name. This is only set if the user is logged in. Note: although this is a number, it is returned as a string since IDs can be larger numbers than Javascript can express.

---

### **HscoreName**

The current hi-score board entry name. This is only set in an *On hi-score* event.

---

### **HscoreRank**

The current hi-score board entry rank, from 1 down to the number of hi-score entries requested. This is only set in an *On hi-score* event.

---

### **HscoreUserIDStr**

Return the user ID of the current hi-score board entry. This is unique even if two people on the hi-score board have the same name. Note: although this is a number, it is returned as a string since IDs can be larger numbers than Javascript can express.

---

### **Score**

Either the current hi-score in an *On hi-score* event, or the current user's score in *On user top score available*.