

# Facebook BlackBerry SDK v0.3.5

---

Last Updated: 2011-02-08

## Important Note

Due to the class name/package restriction of the current BlackBerry OS, please change the package name from `com.blackberry.*` to `com.yourdomain.*` for all source codes in this SDK, or else your application will crash with other applications which use this SDK too.

## Introduction

This SDK is a sample BlackBerry application which seamlessly integrates with facebook using the facebook Graph API (formerly called "facebook connect"). It is an open-source project under MIT license, and it is completely free to use and/or distribute.

## Features

- Connect to facebook using the facebook Graph API.
- Compatible with facebook OAuth 2.0.
- Simple and intuitive to use.

## Getting Started

### 1. Setting up the environment

First of all, you need to get the source files of this SDK from here:

<https://sourceforge.net/projects/facebook-bb-sdk/>

Secondly, make sure that you have a "facebook application" already set up on facebook (<http://developers.facebook.com/setup/>). Next, you have to create the FacebookContext object with the facebook credentials like this:

```
String NEXT_URL = "http://www.facebook.com/connect/login_success.html";
String APPLICATION_ID = "153555168010272";

FacebookContext fbc = new FacebookContext(NEXT_URL, APPLICATION_ID);
```

## 2. Prompting the user to login (if not yet)

There is a handy class called LoginScreen, which can be used to handle this task nicely (See example below), but of course, feel free to implement your own custom login screen if you wish.

Show the LoginScreen if the user has not logged in yet. Also, register an ActionListener to the LoginScreen so that it will callback your onAction() method upon user login:

```
if ((fbc != null) && fbc.isValidAccessToken()) {
    // User logged in already, do post-login stuff here
    ....
} else {
    // User not logged in yet, show the login screen
    loginScreen = new LoginScreen(fbc);
    loginScreen.addActionListener(this);
    pushScreen(loginScreen);
}
```

## 3. Retrieving the facebook access token.

After the user logged in, the ActionListener will be called back with an "ACTION\_LOGGED\_IN" event. Now you have to handle this event by getting the "access\_token" from the event object like this:

```
public void onAction(Action event) {
    if (event.getSource() == loginScreen) {
        if (event.getAction().equals(LoginScreen.ACTION_LOGGED_IN)) {
            // User just logged in, do post-login stuff here
            fbc.setAccessToken((String) event.getData());
            ....
        }
    }
}
```

#### 4. Requesting extended permissions (optional)

By default, your application can access all general information in a user's profile, including her name, profile picture, gender, and friend list. If your application needs to access other parts of the user's profile that may be private, your application can request extended permissions by the “scope” parameter of the login query string.

Example:

[http://www.facebook.com/dialog/oauth?client\\_id=153555168010272&redirect\\_uri=http://www.facebook.com/connect/login\\_success.html&response\\_type=token&display=touch&scope=publish stream,offline access](http://www.facebook.com/dialog/oauth?client_id=153555168010272&redirect_uri=http://www.facebook.com/connect/login_success.html&response_type=token&display=touch&scope=publish_stream,offline_access)

#### 5. Interacting with facebook using the Graph API.

Examples:

To retrieve the currently logged in user:

```
User user = new FBUser("me", accessToken);
```

To update the user's status:

```
user.setStatus("Hello world!");
```

To retrieve the user's friends list:

```
User[] friends = user.getFriends();
```

To retrieve user's news feed:

```
Post[] posts = user.getStream();
```

or to limit the number of posts to retrieve to 10 items:

```
Post[] posts = user.getStream(10);
```

To post to user's wall:

```
user.publishStream("This is the message", "http://www.your-url.com/", "Link name", "Link caption", "Link description");
```

To post to the user's friends' walls:

```
User[] friends = user.getFriends();

if (friends.length > 0) {
    friends[0].publishStream("This is the message", " http://www.your-url.com/", "Link name", "Link caption", "Link description");
}
```

To retrieve post's comments:

```
Post[] comments = post.getComments();
```

## References

BlackBerry Java Application Development:

<http://na.blackberry.com/eng/developers/javaappdev/>

Facebook BlackBerry SDK:

Project: <https://sourceforge.net/projects/facebook-bb-sdk/>

Subversion: <https://facebook-bb-sdk.svn.sourceforge.net/svnroot/facebook-bb-sdk>

Facebook for Mobile Apps:

<http://developers.facebook.com/docs/guides/mobile/>

Facebook Authentication:

<http://developers.facebook.com/docs/authentication/>

Facebook Extended Permissions:

<http://developers.facebook.com/docs/authentication/permissions>

Facebook Graph API:

<http://developers.facebook.com/docs/api>