

SOCAN

music. people. **connected.**
musique. monde. **connectés.**

API Code Samples Acknowledgement API Code Sample

Created: August 01, 2016
Revised: April 10, 2017
Version 1.0
Author: SOCAN

API Code Samples

Acknowledgement API

To submit an Acknowledgement request, you must first have an acknowledgement/submission code. You will receive this code in the response of submission (i.e. A Works API Request). The API Specification document at <https://developer.socan.ca/documentation> will provide further details on all the requirements to submit a request.

Before you begin, you need to consider the following:

- The code sample describes how to create web requests using RestSharp visual studio extension (run 'Install-Package RestSharp' in NuGet PM to install).
- Ensure that Service Point Manager is set to return true for all certificates and its security protocol property is set to TLS12.

Below are the steps required to submit a Works API web request:

1. Set up the web request to use the URL corresponding to the Acknowledge API call and ensure that it is using SSL (For this example, we have set up our code to return true for all certificates), TLS1.2, and a POST Method. The API Key has to be included in the query string as a parameter.

```
string url = "https://api.socan.ca/sandbox/getACK";  
string apiKey = "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";  
string webMethod = "?apiKey=" + apiKey;  
  
...  
  
ServicePointManager.ServerCertificateValidationCallback = delegate { return true; };  
ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls12;
```

2. Attach required JSON to the web request body.

```
string acknowledgeJSON = "{\"API\":\"1001000\",\"MEMBER_NO\":\"9999999\",\"CODE\":\"cw160089z99\"}";  
...  
uploadRequest.AddParameter("application/json", acknowledgeJSON, ParameterType.RequestBody);
```

3. Attach OAuth Token to Request (For more information, refer to "OAuthAPICodesSamples.pdf" document)

```
string oAuthToken = GetOAuthToken();  
string token = "Bearer " + oAuthToken;  
request.AddDefaultHeader("Authorization", token);
```

4. Submit Request and Process response.

```
var request = new RestClient(url);  
var uploadRequest = new RestRequest(webMethod, Method.POST);  
...  
var response = request.Execute(uploadRequest);
```

