HANDS ON WITH THE ORCID API

BOOTCAMP ● VALA2016

Access this workbook online at XXXXXX

Contents

1. About the ORCID APIs

1.1 ORCID API Types & Features

ORCID offers several APIs (Application Programming Interfaces) that allow your systems to connect to the ORCID registry, including reading from and writing to ORCID records. Some API functions are freely available to anyone; others are available to organizations that financially support ORCID with an annual membership subscription.

API Version	Access	Features
Public API	Freely available to anyone	**Authenticate:** Obtain a user's authenticated ORCID iD **Read (Public)**: Search/retrieve public data
Member API	Available to organizations that support ORCID with an annual membership subscription	**Read (Limited):** Search/retrieve "limited-access" data **Add**: Post new items to a record (affiliations, works, etc.) **Update**: Edit or delete items you previously added **Create:** Create new ORCID records (on demand)

1.2 Sandbox Test Environment

In addition to the production Registry and APIs, ORCID also offers a testing environment, the ORCID Sandbox, which we will be using for this boot camp. The Sandbox is open to all users and provides a place to develop and test applications without affecting data in the live ORCID registry.

The Sandbox includes:

- **Sandbox Registry** *(https://sandbox.orcid.org/signin)***:** Simulates the ORCID Registry
- **Sandbox Member API:** Simulates the Member API
- **Sandbox Public API:** Simulates the Public API

The sandbox behaves the same way as the production ORCID Registry with a few exceptions:

- Search & Link tools do not function
- To avoid unintentional spamming, the Sandbox sends emails only to @mailinator.com addresses. Mailinator.com is an inbox testing service that is free and requires no registration. To receive emails from the Sandbox, use an @mailinator.com email address when creating Sandbox record(s)
- Links in the top menu bar (For Researchers, For Organizations, About, etc.) do not function

1.3 ORCID API Technologies

All of the ORCID APIs are based on the same set of technologies:

    - **REST:** ORCID APIs are "RESTful," which means that they use HTTP (hyper-text transfer) calls to transfer information.
- **OAuth:** ORCID APIs use the OAuth 2.0 authentication protocol in order to grant client applications access to users' ORCID records.
    - **XML/JSON:** ORCID APIs support data exchange in either XML or JSON format

1.4 ORCID API Tools

In order to use the ORCID APIs you will need the following software tools:

- Web browser: Firefox (33+), Chrome (38+), Internet Explorer (10+), Safari (6+)

- Internet connection
- Plain text editor: TextEdit (Mac), Notepad++ (Win), or your preferred plain text editor
- Software capable of making HTTP requests:
    - cURL: free, command-line application available for Mac or Windows at http://curl.haxx.se/download.html (pre-installed on most Mac OS versions; accessible within Terminal application)
    - Online tools, e.g. hurl.it or Google OAuth Playground
    - Your own web application, in a language such as Java, Ruby, Python, PHP, etc.

For this boot camp, we will be using the online tool Google OAuth Playground.

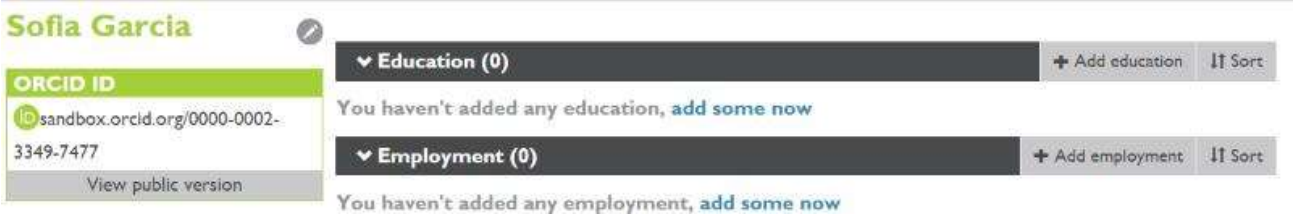
2. Public API: Searching & Retrieving ORCID Records

2.1 Create a Sandbox User Account

In order to try out API calls, such as a reading a record and adding information to it, you will also need to create a test ORCID record in the Sandbox. This can be done through the user interface, much like in the live ORCID registry.

1. Open a web browser and navigate to https://sandbox.orcid.org/signin
2. Click **Register for an ORCID iD**.
3. Complete the form with a name, email, and password

   IMPORTANT! *Don't use a real email address! Instead, make up an address ending in @mailinator.com (use any prefix, e.g. sgarcia@mailinator.com).*
4. Click the **I consent…** checkbox and click **Register**.
5. After completing the registration process, you will be taken to your new Sandbox ORCID record. Make note of the 16-digit ORCID iD for this record – you will need this in order to make API calls later during the boot camp.



6. Add a few pieces of information to your new Sandbox record – biography, education, employment, etc. – so that you have some data to work with in future steps.

2.2 Accessing the Sandbox Public API

Access to the ORCID Public API requires a set of credentials consisting of a Client ID and a Client Secret. Public API credentials are tied to an individual's ORCID record and cannot be transferred to another person. The process for getting Public API credentials is identical in the sandbox and production environments.

1. To access the Public API, you'll first need to verify your email address. Visit http://mailinator.com/ and enter the email that you used to create your Sandbox account.
2. Open the confirmation message from ORCID and click **Verify your email address**.
3. Back in your ORCID account, click **Developer Tools** at the top of the screen.

4. Click the **Register for the free ORCID public API button**.

5. Review and agree to the terms of service when prompted.
6. In the form that appears, enter the following values:

Name	Name of the ORCID application that you are developing Ex: State University ORCID Connector
Website	URL for the homepage of your ORCID application Ex: http://stateuniversity.edu/orcid
Description	Description of the ORCID application that you are developing Ex: An application that allows users to connect their ORCID iD to their State University faculty profile.
Redirect URIs	URIs for use with the OAuth 2.0 protocol; for more information, see About redirect URIs. For this boot camp, use: https://developers.google.com/oauthplayground/ *(add this URI automatically by clicking "+Google OAuth2 Playground" under Test redirect URIs)*

7. Click the Save icon at the bottom of the form to generate your API credentials.

8. To view your API credentials, click **Show Details**.

9. Your API credentials – Client ID and Client Secret – are shown just beneath your Redirect URIs. We'll be using these later in the boot camp, but no need to copying them down – they are always available in the **Developer Tools** section of your account.

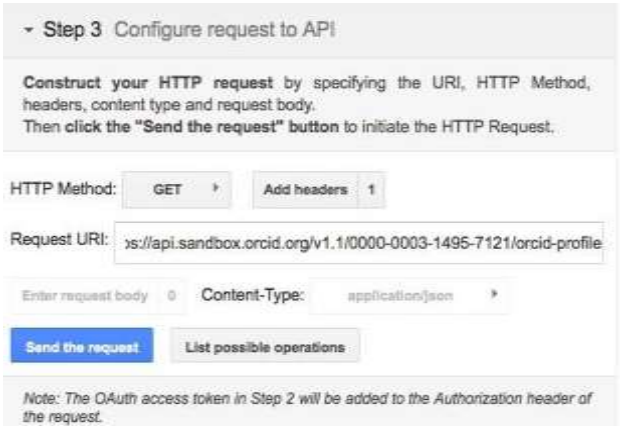
3. Public API: Searching & Retrieving ORCID Records

3.1 Retrieving Public Record Data

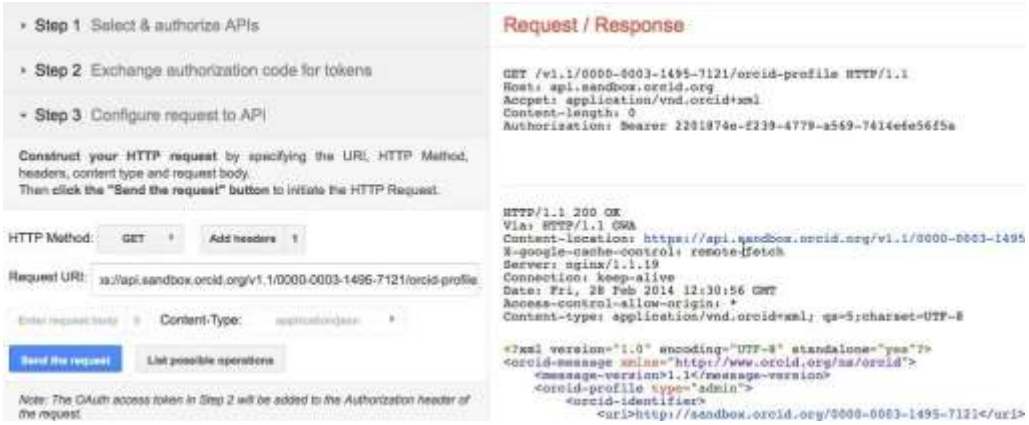
Using the Public API, you can retrieve the XML version of any user's public ORCID record. This is same information that you'll see by viewing a record in the user interface of the ORCID registry, but in machine-readable XML format. In a real-world web application, this process can be used to retrieve and copy information from ORCID records into your own system.

1. Visit https://developers.google.com/oauthplayground/
2. Beneath **Step 3: Configure request to API** on the left side of the screen, set **HTTP** Method to **GET**
3. Click **Add headers** and enter the following values:
    - **Header name:** Accept
    - **Header value:** application/vnd.orcid+xml

4. In the **Request URI** field, enter https://pub.sandbox.orcid.org/v1.2/[orcid-id]/orcid-profile, replacing [orcid-id] with the ORCID iD of the Sandbox record that you created earlier (ex: https://pub.sandbox.orcid.org/v1.2/0000-0002-1223-3173/orcid-profile)



5. Leave the Request Body and Content-Type fields blank, and click **Send the request**.
6. The XML for the ORCID record that you requested will appear in the **Request/Response** section on the right side of the screen.



3.2 Searching Public Record Data

In addition to retrieving an individual's entire ORCID record, the Public API can be used to search for all records whose public data contain particular search terms.

Note that only a portion of the ORCID record is returned for each matching result. To view the entire record, use the Public API to retrieve the record, as in section 3.1.

This tutorial provides a very brief introduction to searching with the API. – For much more information, see http://members.orcid.org/api/tutorial-searching-api-12-and-earlier

3.2.1 Basic Keyword Search

A basic keyword search performs a full-text search of all publicly-visible information in users' ORCID records (Names, ORCID iDs, and any other data with a privacy setting of "Everyone").

1. Visit https://developers.google.com/oauthplayground/
2. Beneath **Step 3: Configure request to API** on the left side of the screen, enter the following:

HTTP Method	GET
Add Headers	Header name: Accept Header value: application/vnd.orcid+xml
Request URI	http://pub.sandbox.orcid.org/v1.2/search/orcid-bio?q=newman *Searches for records with the term "newman" in any field*

3. Leave the Request Body and Content-Type fields blank, and click **Send the request**.
4. XML for matching records appears in the **Request/Response** section on the right.

3.2.2 Boolean Keyword Search

In addition to the basic keyword search, multiple keywords keywords can be included using Boolean terms "AND" or "OR". Exact phrases can be included using quotation marks. Note that "+" signs must be substituted for spaces and that Boolean terms must be uppercase.

1. Beneath **Step 3: Configure request to API** on the left side of the screen, enter the following:

HTTP Method	GET
Add Headers	Header name: Accept Header value: application/vnd.orcid+xml
Request URI	http://pub.sandbox.orcid.org/v1.2/search/orcid-bio?q=michael+AND(hkbu+OR+"Hong+Kong+Baptist+University") *Searches for records with the term "michael" and "hkbu" or "Hong Kong Baptist University" in any field*

2. Leave the Request Body and Content-Type fields blank, and click **Send the Request**.
3. XML for matching records appears in the **Request/Response** section on the right.

3.2.3 Fielded search

For more specific results, search for terms in particular record fields rather than entire ORCID records.

Available search fields include:

- ORCID iD
- Names (Given Names, Family Name, Credit Name, Other Names)
- Email address
- Work Titles
- External Identifiers (DOIs, PMIDs, etc.)
- Creation/Last Modified Dates

For a complete list, see http://members.orcid.org/api/tutorial-searching-api-12-and-earlier

1. Beneath **Step 3: Configure request to API** on the left side of the screen, enter the following:

HTTP Method	GET
Add Headers	Header name: Accept Header value: application/vnd.orcid+xml

Request URI	http://pub.sandbox.orcid.org/v1.2/search/orcid-bio/?q=family-name:Sanchez *Searches for ORCID records with the term "sanchez" in Family (Last) Name*

2. Leave the Request Body and Content-Type fields blank, and click **Send the request**.
3. XML for matching records appears in the **Request/Response** section on the right.

3.2.4. More searches to try!

- **Search for records associated with the exact DOI 10.1087/20120404:**
  http://pub.sandbox.orcid.org/v1.2/search/orcid-bio/?q=digital-object-ids:%2210.1087/20120404%22
- **Search for records modified between May 6, 2015 and today:**
  https://pub.sandbox.orcid.org/search/orcid-bio?q=profile-last-modified-date:%5B2015-05-06T00:00:00Z%20TO%20NOW%5D

4. BONUS: Public API: Getting Authenticated ORCID iDs

When you need to collect a user's ORCID iD in your web application, the best method is to use the ORCID API to get an authenticated ORCID iD.

Rather than relying on users to correctly type their ORCID iDs into a form, the API process prompts users to log into their ORCID account and passes their ORCID iD to your system automatically. This prevents typos and ensures that you get the correct ORCID iD for each user.

4.1 OAuth Introduction

Getting an authenticated ORCID iD through the API depends on an authentication process called OAuth. With OAuth, a user can authorize a third-party system to access their account in another system without sharing their login information.

You've likely encountered OAuth many times before – it's the same technology that allows you to log into other applications using your Google, Facebook or Twitter account.

The OAuth process consists of the following steps:

1. Get an Authorization Code from the user
2. Exchange the Authorization Code for the user's ORCID iD

4.2 Setting up the OAuth Playground

In a real-world situation, API interactions are completed by your system using a programming language such as PHP, Java, or Ruby on Rails. For this boot camp, however, we'll be using a web-based tool, the Google OAuth Playground.

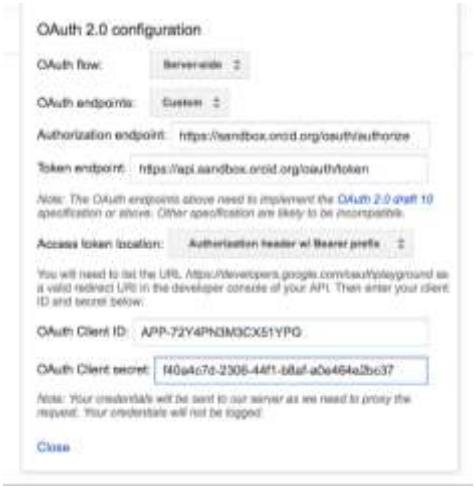
The first thing we need to do is set up the OAuth Playground to work with the ORCID API.

1. Visit https://developers.google.com/oauthplayground
2. Click the gear icon in the upper right corner to open the configuration form.

3. In the configuration form, enter the following:

OAuth flow	Server-side
OAuth endpoints	Custom
Authorization endpoint	https://sandbox.orcid.org/oauth/authorize

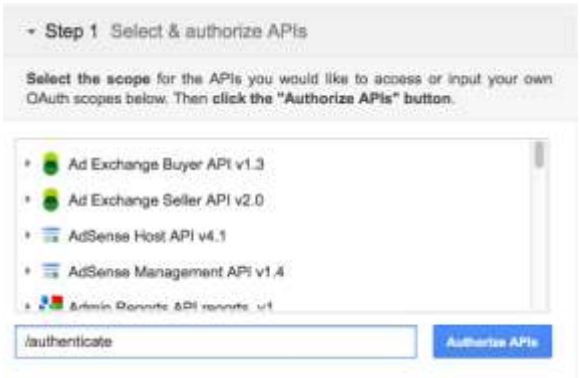
Token endpoint	https://pub.sandbox.orcid.org/oauth/token
Access token location	Authorization header w/Bearer prefix
OAuth Client ID	Your Public API client id (ex: APP-F6TMYF419CVYMSNE)
OAuth Client Secret	Your Public API client secret (ex: f40a4c7d-2306-44f1-b8af-a0e464e2bc37)



4. The configuration screen should look similar to the image at right. After you've entered the settings, click Close in the lower-left corner of the configuration screen.

4.3 Get an Authorization Code

To get an Authorization Code, you'll need to prompt the user to log into their ORCID account and grant permission to your application. In a real-world situation, this is done using an authorization URL that you construct. With the OAuth Playground, however, this step is done by configuring some additional settings and clicking a button.



1. First, we will specify the permission (scope) that we wish to request from the user.

   Do not select any of the options from the list beneath **Step 1: Select & authorize APIs**. Instead, type **/authenticate** in the text box.

   The /authenticate scope allows you to retrieve a user's ORCID iD. This is the only scope available in the Public API. The Member API, however, permits additional scopes, which we'll explore in the next section.

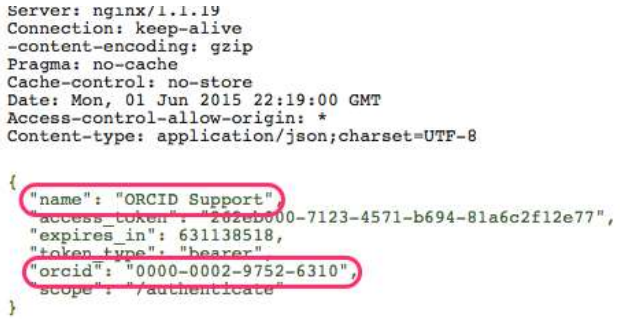
2. Click **Authorize APIs**.
3. An ORCID OAuth login screen will appear, requesting that the user grant the permissions entered in the previous steps. When this screen appears, click Sign In, and sign into your Sandbox account.
4. Click Authorize on the ORCID OAuth login screen and you will be sent back to the OAuth Playground. A 6-character code will appear in the Authorization Code field beneath **Step 2: Exchange authorization codes for tokens**.

4.4 Exchange the Authorization Code for the User's ORCID iD

Once you have an Authorization Code, you can exchange it for the user's ORCID iD. In a real-world situation, this exchange would be done by your system, using a programming language such as PHP, Java, or Ruby on Rails. With the OAuth Playground, however, this step is done by clicking a button.

1. Beneath the **Authorization Code** field, click **Exchange authorization code for tokens**.

2. The user's name and ORCID iD will appear in the **Request/Response** section on the right side of the screen.



5. Member API: Access to amend ORCID records

As discussed in section 1.1, the Public API can only be used to read and search ORCID records, and to get authenticated ORCID iDs. The Member API, however, can be used to add new information to ORCID records, as well as to update information previously added.

5.1 Accessing the Sandbox Member API

As with the Public API, client credentials consisting of a client ID and a client secret are needed in order to access the Member API. Public API credentials cannot be used to access the Member API, so we'll be using a different client ID and secret for the remainder of this boot camp.

Client Credentials for the Member APIs are issued by ORCID. For this boot camp, we have set up Sandbox Client Credentials for you in advance. In the future, you can obtain Client Credentials using the request form at http://orcid.org/content/register-client-application

5.2 Setting up the OAuth Playground

We'll continue to use the Google Developers' OAuth Playground for the next exercises, but a few configuration changes are needed in order to work with the Member API.

1. Visit https://developers.google.com/oauthplayground
2. Click the gear icon in the upper right corner to open the configuration form.

3. Enter the following:
(Fields edited to work with the Member API are highlighted; the rest remain the same.)

OAuth flow	Server-side
OAuth endpoints	Custom
Authorization endpoint	https://sandbox.orcid.org/oauth/authorize
Token endpoint	**https://api.sandbox.orcid.org/oauth/token**
Access token location	Authorization header w/Bearer prefix
OAuth Client ID	**Your Member API client ID** (ex: APP-F6TMYF419CVYMSNE)

OAuth Client Secret	**Your Member API client secret** (ex: f40a4c7d-2306-44f1-b8af-a0e464e2bc37)

4. The configuration screen should look similar to the image at right. After you've entered the settings, click **Close** in the lower left corner of the configuration screen.

6. Member API: Getting permission to edit ORCID records

6.1 Obtaining Access Tokens

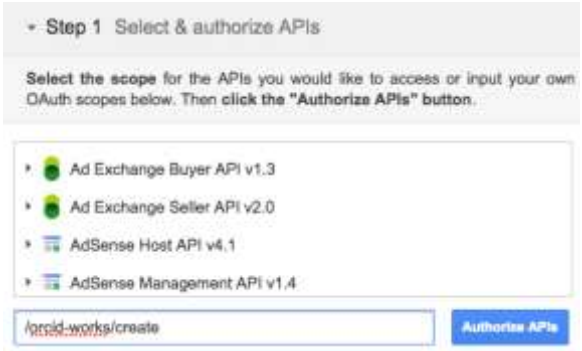
To access an ORCID record via the Member API, you first need to get permission from the owner of the record in the form of an Access Token.

This process of granting permission uses OAuth and is similar to the process used for obtaining an authenticated ORCID iD described in section 4.

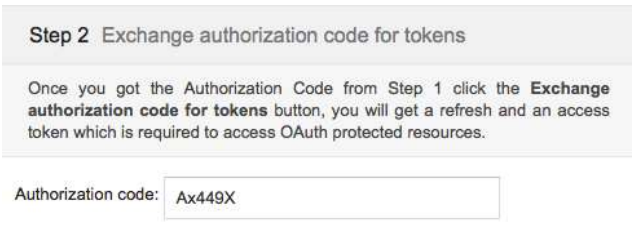
1. Get an **Authorization Code**.
2. Exchange the Authorization Code for an **Access Token**.

6.1.1 Get an Authorization Code

To get an Authorization Code, you'll need to prompt the user to log into his/her ORCID account and grant permission to your application. In a real-world situation, this is done using an authorization URL that you construct. With the OAuth Playground, however, this step is done by configuring some additional settings and clicking a button.



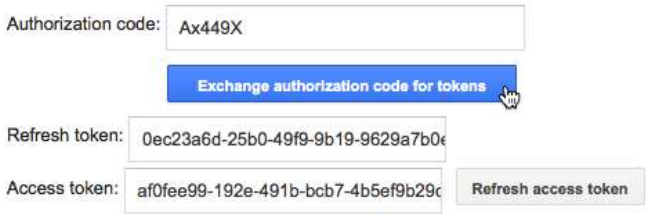
1. Beneath **Step 1: Select & authorize APIs** on the left side of the screen, type **/orcid-works/create** in the text box (do not select any of the options in the box above).
2. Click **Authorize APIs**.
3. An ORCID OAuth login screen will appear. Click **Sign In** and sign into your Sandbox account.
4. Click **Authorize** on the ORCID OAuth login screen and you will be sent back to the OAuth Playground. A 6-character code will appear in the **Authorization Code** field.



6.1.2 Exchange the Authorization Code for an Access Token

Once you have an Authorization Code, you can exchange it for an Access Token, which allows you to read from/write to a user's ORCID record. In a real-world situation, this exchange would be done by your system, using a programming language such as PHP, Java, or Ruby on Rails. With the OAuth Playground, however, this step is done by clicking a button.

1. Beneath the **Authorization Code** field, click **Exchange authorization code for tokens**.
2. The token will appear in the **Access Token** field.
3. Note that you are provided with additional information in the **Request/Response** section on the right side of the screen, such as the name and ORCID iD of the user who granted permission, the lifespan of the token (20 years), and the scope for which the token is valid.



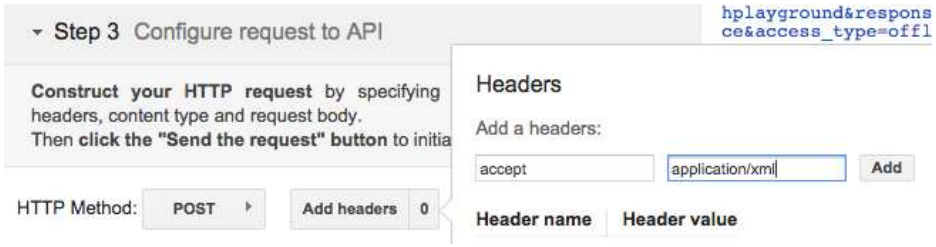

```
Access-Control-Allow-Origin: *
Content-Type: application/json;charset=UTF-8

{
"name": "ORCID Staff",
"access_token": "763b7dcb-adb1-4942-af9d-afe144c3f2f9",
"expires_in": 631138518,
"token_type": "bearer",
"orcid": "0000-0002-1223-3173",
"scope": "/orcid-profile/read-limited"
}
```

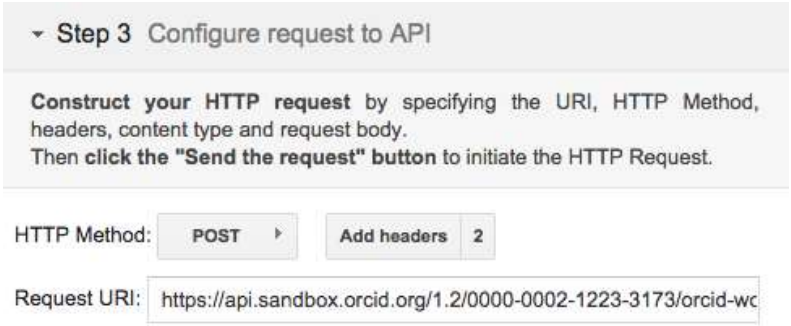
7. Member API: Writing to ORCID records via the API

7.1 Adding a New Work to an ORCID Record

1. Beneath **Step 3: Configure request to API**, set **HTTP Method** to **POST**.
2. Click **Add headers** and enter the following values:
    - **Header name:** accept
    - **Header value:** application/vnd.orcid+xml



3. Click **Add** to add another header and enter the following values:
    - **Header name:** Content-type
    - **Header value:** application/vnd.orcid+xml
4. Click **Add** again, then click **Close**.
5. In the **Request URI** field, enter https://api.sandbox.orcid.org/v1.2/[orcid-id]/orcid-works, replacing [orcid-id] with the ORCID iD of the Sandbox record that you created earlier (ex: https://api.sandbox.orcid.org/v1.2/0000-0002-1223-3173/orcid-works)



6. Click **Enter request body**. Here is where you'll enter the XML for the works you wish to add.
7. Visit http://git.io/vITI9 and copy the XML in the **Sample Work** section.

8. Paste the copied content into the **Request Body** text box and click **Close**.

9. Click **Send the request**.
10. The results will appear in the **Request/Response** section on the right side of the screen. Scroll to the bottom – if you see **HTTP/1.1 201 Created**, the work was successfully posted!

11. Visit the **public view** of your Sandbox record at http://sandbox.orcid.org/[Your sandbox iD] to see the work that you added in the user interface.

7.2 Updating a Work

In the last section, we asked the user for permission to add a new work to their ORCID record using the /orcid-works/create scope. To edit that same work, we need to request permission using the /orcid-works/update scope.

In a real-world situation, create and update permissions can be requested in the same step, resulting in an Access Token that can be used to both add and edit works. When using the OAuth Playground, however, we can only request permission for one scope at a time, so we'll need to generate a new Access Token in order to edit the work that we just added.

1. Beneath **Step 1: Select & authorize APIs**, type **/orcid-works/update** in the text box.
2. Click **Authorize APIs**.
3. An ORCID OAuth login screen will appear. Click **Sign In**, and sign into your Sandbox account.
4. Click **Authorize** on the ORCID OAuth login screen and you will be sent back to the OAuth Playground. A 6-character code will appear in the **Authorization Code** field.

5. Beneath the Authorization Code field, click **Exchange authorization code for tokens**.
6. The token will appear in the **Access Token** field.
7. Beneath **Step 3: Configure request to API**, set **HTTP Method** to **PUT**.
8. Click **Add headers** and enter the following values:
    - **Header name:** Accept
    - **Header value:** application/vnd.orcid+xml
9. Click **Add** to add another header and enter the following values:
    - **Header name:** Content-type
    - **Header value:** application/vnd.orcid+xml
10. Click **Add** again, then click **Close**.
11. In the **Request URI** field, enter https://api.sandbox.orcid.org/v1.2/[orcid-id]/orcid-works, replacing [orcid-id] with the ORCID iD of your Sandbox record (ex: https://api.sandbox.orcid.org/v1.2/0000-0002-1223-3173/orcid-works)
12. Click **Enter request body**. This is where you'll enter the XML for the work that you wish to edit.
13. Visit http://git.io/vITI9 and copy the XML in the **Sample Work Updated** section.
14. Paste the copied content into the **Request Body** field and click **Close**.
15. Click **Send the Request**.
16. The results will appear in the **Request/Response** section on the right side of the screen. If the full XML of the user's record appears, the work was successfully updated!
17. Visit the public view of your Sandbox record at http://sandbox.orcid.org/[Your sandbox iD] to see the changes to the work in the user interface.

8. BONUS: Member API: Creating New Records

Organizations are often interested in creating ORCID records on behalf of their faculty, staff, or students. The ORCID API supports this through a create on demand process that allows users to create a new record at any time, and to grant your system read/write access to their record (via the API). Users who already have an ORCID iD can use the same process to send their existing iD to your system and grant your system read/write access.

See a demo of this process at: https://orcid-create-on-demand.herokuapp.com/

In this section, we'll walk through the basic steps to implement a create on demand process. For more information, see http://members.orcid.org/create-records

8.1 Construct the Authorization URL

The create-on-demand process begins with a special authorization URL that you construct, which includes your ORCID API client credentials and information about any read/write permissions that you'd like to request from the user.

8.1.1 Base URL

Your authorization URL begins with one of the following addresses, depending on the environment that you're using.

Sandbox API	https://sandbox.orcid.org/oauth/authorize?
Production API	https://orcid.org/oauth/authorize?

1. Open a text editor and copy/paste or type the base URL into a new document. For this tutorial, we're using the Sandbox API, so our base URL is:

    ```
    https://sandbox.orcid.org/oauth/authorize?
    ```

8.1.2 Required Parameters

After the base URL, we need to include some additional bits of information – called "parameters" – that identify your client app to the ORCID API and tell the API what permissions you'd like to request from the user. Each parameter is separated by an `&` character.

client_id	Your Member API client ID (ex: APP-F6TMYF419CVYMSNE)
scope	The read/write permission(s) you wish to request from the user, from the **ORCID Scopes** list: http://members.orcid.org/api/orcid-scopes If including multiple permissions, separate them with an HTML-encoded space (`%20`), ex: `/orcid-works/create%20/orcid-works/update` *At minimum, /authenticate needs to be included, which allows you to obtain the user's ORCID iD.*
response_type	`code`
redirect_uri	The page on your site that the user should be sent to when the authorization is complete, ex: http://yoursite/somepage.html *This URL must be registered as part of your API credential registration.* For this boot camp, use: https://developers.google.com/oauthplayground/

1. In your text editor, add the required parameters to your base URL, as in the example below. Make sure to replace the Client ID with your assigned Client ID!

```
https://sandbox.orcid.org/oauth/authorize?
client_id=APP-F6TMYF419CVYMSNE&
scope=/orcid-works/create%20/orcid-works/update&
response_type=code&
redirect_uri=https://developers.google.com/oauthplayground
```

8.1.3 Optional Parameters

In addition to the required parameters, you can add any of the optional parameters below to customize the ORCID authorization screen and pre-fill the registration form with information from your system.

state	A random value used as a security tactic to prevent Cross-Site Request Forgery (CSRF). *When ORCID sends the user back to your redirect_uri, the state value will be included in the response.* More about using the state parameter to prevent CSRF
family_names	The user's family (last) name, used to pre-fill the registration form.
given_names	The user's given (first) name, used to pre-fill the registration form.
email	The researcher's email address, used to pre-fill the sign-in or registration form.
lang	The language to display the authorization page in, as a 2-letter ISO 639-1 code
show_login	Forces the login form to display by default, set to `true` or `false`

1. In your text editor, add some optional parameters to pre-fill the registration form. Your final authorization URL should like something like:

IMPORTANT! *Make sure to use a @mailinator.com email address. Use a different address from the example – addresses must be unique.*

```
https://sandbox.orcid.org/oauth/authorize?
client_id=APP-F6TMYF419CVYMSNE&
scope=/orcid-works/create%20/orcid-works/update&
response_type=code&
redirect_uri=https://developers.google.com/oauthplayground&
family_names=Orcidson&
given_names=Orc&
email=orcidson@mailinator.com
```

8.2 Build a Redirect Page/Application

After clicking your authorization URL and completing the ORCID registration or sign-in process, users are sent to the page that you specified in the redirect_uri parameter.

When a user is sent to this page, the ORCID API attaches an Authorization Code to the end of the URL for the page. As we saw in the previous sections of this tutorial, an Authorization Code can be exchanged for a user's ORCID iD and (optionally) an Access Token.

In this tutorial, we've been relying on the OAuth Playground to perform the authorization code exchange. In a live application, this is done by your system via your specified redirect page.

This step requires some programming expertise, so you may need to enlist the help of a web developer. A variety of languages can be used – code can be executed right in the redirect page, or the redirect page can connect to an application living on your server.

Regardless of the language, your redirect page (or any server application it connects to) needs to:

1. Capture the Authorization Code.
2. Exchange the Authorization Code for an Access Token.
3. Store the Access Token and ORCID iD for use in future use (e.g. in a database).

See an example in PHP at: https://github.com/lizkrznarich/orcid-demo-app

For more examples contributed by ORCID members, see: http://members.orcid.org/api/code-examples

Note: *If you simply want to promote ORCID creation, but don't need to collect users' ORCID iDs or Access Tokens, you can create a static HTML redirect page that does not perform an authorization code exchange.*

8.3 Prompt Users to Create or Connect their ORCID iD

Once you've created an authorization URL and a redirect page, the next step is to prompt users to click the authorization URL. To do this, you can:

1. Send each user an email with the authorization URL.

   URLs can easily be customized with family_name, given_name, and email parameters using the mailmerge feature available in many email programs.

2. Create a page on your site with the authorization URL included as a button or a link.

For this tutorial, simply copy the link and paste it into your browser's address bar and press Return/Enter.

1. If you've signed out of your Sandbox account since the prior exercises, you should see the ORCID registration form with the name and email address pre-filled – to finish creating your new sandbox account:
   - Enter a **password** (twice)
   - Click the checkbox beside **I consent to the privacy policy…**
   - Click **Authorize**
2. After clicking Authorize, you'll be sent to the redirect URI (Google's OAuth Playground), and you'll see the 6-character authorization code in your browser's address bar. A confirmation email has also been sent to the address that you specified in the registration form.

3. Leave the OAuth Playground window open for the next steps, but don't exchange the Authorization Code yet.

8.4 Obtain Access Tokens and ORCID iDs

In a live application, this step would be completed via your redirect page, as discussed above. For this tutorial, we'll continue to use the OAuth Playground.

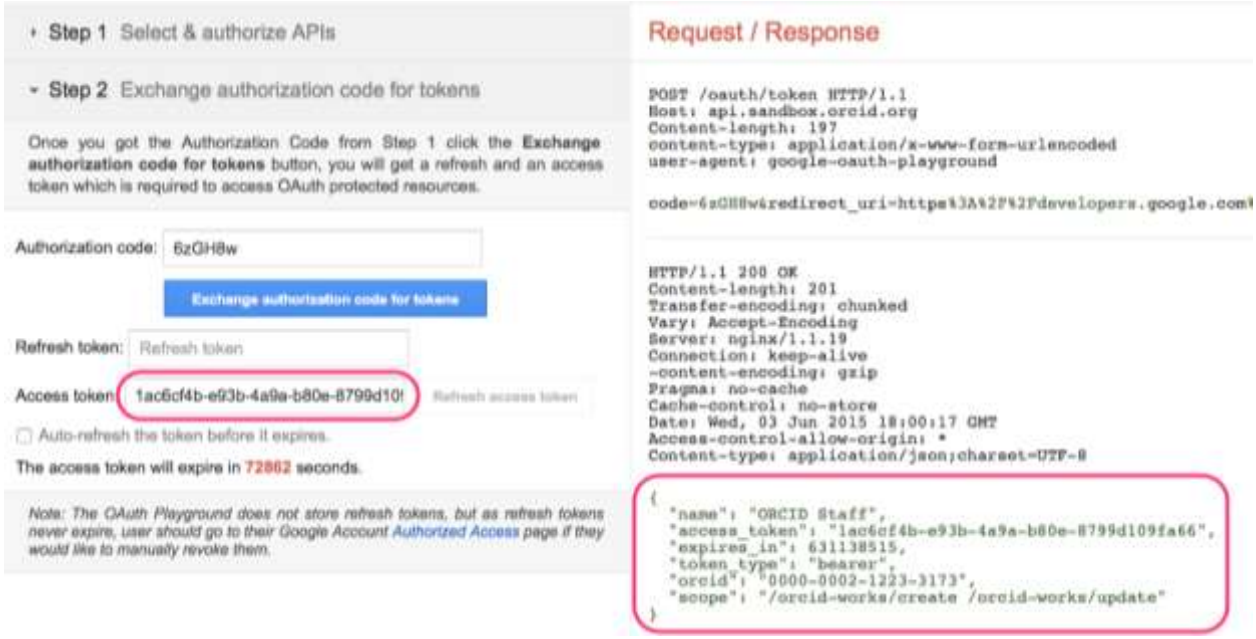
Before we continue with the Authorization Code exchange, we first need to make sure that the OAuth Playground is still configured correctly.

1. Click the gear icon in the upper right corner to open the configuration form.

2. In the configuration form, verify that the following settings (which are the same as those used in sections 4.6 and 4.7) remain. Re-enter any that are not set.

OAuth flow	Server-side
OAuth endpoints	Custom
Authorization endpoint	https://sandbox.orcid.org/oauth/authorize
Token endpoint	https://api.sandbox.orcid.org/oauth/token
Access token location	Authorization header w/Bearer prefix
OAuth Client ID	Your Member API client ID (ex: APP-F6TMYF419CVYMSNE)
OAuth Client Secret	Your Member API client secret (ex: f40a4c7d-2306-44f1-b8af-a0e464e2bc37)

3. Click **Close** in the lower left corner of the configuration screen.
4. Beneath the **Authorization Code** field, click **Exchange authorization code for tokens**.
5. The token will appear in the Access Token field and in the **Request/Response** section on the right side of the screen. The Request/Response section also includes the user's name and ORCID iD, and the scope(s) for which token is valid. In your live app, these are the pieces of information that your system should store.



8.5 Read from/Write to Users' ORCID Records

Once you have the user's ORCID iD and Access Token, it can be used to make API calls that read from/write to the user's ORCID record per the permissions requested, as we did earlier in this tutorial.

Access Tokens are valid for 20 years by default. They can be used immediately or at any point in the future.

For example, you can:

- Read information from the user's ORCID record and copy it into your system
- Add information from your system to the user's ORCID record – affiliations, works, etc.
- Update the user's ORCID record when new information is available in your system
- Update your system when new information is added to the user's ORCID record

9. ORCID API Resources

- See example implementations and workflow guides http://members.orcid.org
- Read technical documentation http://members.orcid.org/api
- Join the ORCID API Users Group https://groups.google.com/group/orcid-api-users
- Sign up for a Technical Webinar http://members.orcid.org/event-list
- Email ORCID Support support@orcid.org