Re-creating CAMPER

[Overview 2](#_Toc525550891)

[Basic steps for re-creating CAMPER 2](#_Toc525550892)

[Security settings 3](#_Toc525550893)

[User tokens 3](#_Toc525550894)

[Adding and updating code pages 4](#_Toc525550895)

[CAMPER COMMON 4](#_Toc525550896)

[List of CAMPER COMMON pages: 7](#_Toc525550897)

[CAMPER 8](#_Toc525550898)

[Customizing the dashboard for CAMPER 9](#_Toc525550899)

[Defining variables for CAMPER 10](#_Toc525550900)

[Creating Tables 11](#_Toc525550901)

[Updating field formulas 11](#_Toc525550902)

[CAMPER COMMON tables overview 13](#_Toc525550903)

[CAMPER tables overview 13](#_Toc525550904)

[Customizing table report views 15](#_Toc525550905)

[Uploading data 15](#_Toc525550906)

[Working around the import tag error in CAMPER 16](#_Toc525550907)

[Importing to CAMPER Record ID-linked tables without breaking the links 16](#_Toc525550908)

[Record ID-Linked CAMPER table descriptions 18](#_Toc525550909)

[Importing to other CAMPER tables 19](#_Toc525550910)

[Other CAMPER table descriptions 19](#_Toc525550911)

[Importing data to CAMPER Common 21](#_Toc525550912)

[User roles and groups 22](#_Toc525550913)

[Groups 22](#_Toc525550914)

[Roles 22](#_Toc525550915)

[Role summary 23](#_Toc525550916)

# Overview

CAMPER and CAMPER COMMON are Quick Base apps that were used to create and manage the JUICE website’s UI text and learning content. If you have configured the JUICE site to use Amazon S3 to hold the files that define the JUICE’s UI text and content, you may re-create CAMPER and CAMPER COMMON for use as your content management system.

This documentation assumes you have created an instance of the JUICE site using S3, have reviewed the Basic JUICE documentation that is available in the JUICE GitHub documentation directory, and are familiar with the JUICE site structure.   
  
It also assumes that you have read the CAMPER design and structure documentation, and have basic familiarity with Quick Base.

Once CAMPER and CAMPER COMMON are set up, there is additional documentation in the GitHub documentation directory on how to work in CAMPER to modify and create new content components for the JUICE site, as well as create alternate custom programs.

# Basic steps for re-creating CAMPER

The basic steps for re-creating the JUICE content management system (CAMPER) in Quick Base are as follows. Specific instructions for each step are included below.

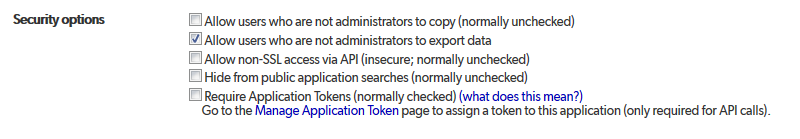
1. Get the CAMPER and CAMPER COMMON code pages from the GitHub JUICE repository directory called **quickbase\_pages\_GENERIC. You also will need these files from the Managing and customizing JUICE with Quick Base directory:**
   1. **CAMPER COMMON Data**
   2. **CAMPER Data**
   3. **Table descriptions**
2. Create 2 new apps: CAMPER COMMON and CAMPER.
3. Modify the app security settings and configure a user token for CAMPER.
4. Set up CAMPER COMMON
   1. Add and update code pages
   2. Create tables and their relationships. Give access to CAMPER in the tags table tag field.
   3. Upload data
   4. Customize table report views as desired
5. Set up CAMPER  
   1. Add and update its code page
   2. Create a custom dashboard and add required links
   3. Define variables
   4. Create tables and their relationships, linking to CAMPER COMMON where needed
   5. Upload data
   6. Customize table report views as desired
   7. Update the code page **program\_import\_module.html**
6. Add required user groups, and add user roles as desired.

# Security settings

The JUICE site authoring templates update file names in CAMPER and in alternate JUICE programs when a file is saved. To allow access to CAMPER by the JUICE site, CAMPER uses **user tokens**.

CAMPER COMMON and CAMPER do not use Application tokens. If Application tokens are required, many of the CAMPER functions will not work.

After you create CAMPER COMMON and CAMPER, go to **Home | Settings | App Properties | Security Options**, and uncheck the Require Application Tokens checkbox. The settings should look like this:

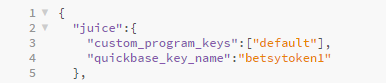


# User tokens

CAMPER uses **user tokens** to allow the JUICE authoring templates to update the file name records in CAMPER. If the user token is not set, you will not be able to save changes to your files.

For the user tokens to work properly, do the following:

1. Your Quick Base account manager must create a user token and assign CAMPER to it.
2. In Amazon S3, find and download the file /programs/program\_id\_mapping.json. Update the first group to replace the value “betsytoken1” with the user token **name**:



1. Upload the file to S3, and force the site to reload as follows:  
   1. Access the JUICE site as Alistair Admin using /juice/oauth\_test. See the documentation on *Accessing JUICE* for more details on how to log in.
   2. After JUICE loads, update the URL with this: /juice/reload
   3. A response of “true” indicates that the site has reloaded the program id mapping values.
2. In the JUICE site code config file **default.json**, update the **qb\_keys** values to replace betsytoken1 with your own user token name, and add the token value, which looks something like this:

c3rjkl\_wert\_sw2y3sdeksjelskf8sheycksh8jus

# Adding and updating code pages

CAMPER’s functionality relies on a number of code pages. The majority of code pages are located in the CAMPER COMMON application, and are accessible via the Pages screen under Settings. They are identified by the .html or .js extensions. The core CAMPER application does contain one additional code page, named **common\_dbspecific.js**.   
  
When creating a new instance of CAMPER and CAMPER COMMON, some modifications must be made to these files, as described below.

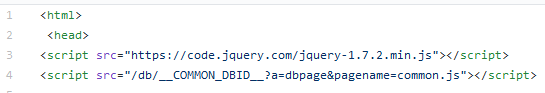
## CAMPER COMMON

CAMPER COMMON requires 22 code pages that are used by CAMPER. If you chose to create apps for alternate JUICE programs, those apps use the CAMPER COMMON code pages also.

To add the code pages to CAMPER COMMON, you must do the following:

1. Download the code pages from the JUICE **quickbase\_pages\_GENERIC** directory in GitHub.
2. Create each page in CAMPER COMMON, then copy and paste the text from the GitHub pages into it. Note: If you can figure out how to import the pages into Quick Base, do that instead!
3. On EACH PAGE, replace the text \_\_COMMON\_DBID\_\_ with your own instance of CAMPER COMMON’s database id, which is displayed in the URL:





Example:

If the URL of you CAMPER COMMON instance home page is

[*https://myaccount.quickbase.com/db/****bmf7isxyz***](https://myaccount.quickbase.com/db/bmf7isxyz) *,*

the script declaration line

*<script src="/db/\_\_COMMON\_DBID\_\_?a=dbpage&pagename=common.js"></script> ,*

must be substituted by

*<script src="/db/* ***bmf7isxyz*** *?a=dbpage&pagename=common.js"></script>*

1. On the page **common.js***,* replace the string \_\_COMMON\_TAGS\_DBID\_\_ with the table id of the Tags table in CAMPER COMMON.

Example:

If the URL of the Tags table home page in your CAMPER COMMON instance is

[https://myaccount.quickbase.com/db/ bmh25cxyz?a=td](https://myaccount.quickbase.com/db/%20bmh25cxyz?a=td) *,*

the line

*var QBU\_dbid\_common\_tags = "\_\_COMMON\_TAGS\_DBID\_\_";*

must be substituted by

*var QBU\_dbid\_common\_tags = “****bmh25cxyz*** *";*

1. The code page **publish\_search.html** contains these strings:

dev.\_\_YOUR\_SERVER.COM\_\_

test.\_\_YOUR\_SERVER.COM\_\_

www.\_\_YOUR\_SERVER.COM\_\_

connect.\_\_YOUR\_SERVER.COM\_\_

Replace these with the appropriate names for your server configuration. If you are only using one or two servers, remove the others from the list.

|  |
| --- |
|  |

1. The code page **viewjson.html** contains the strings \_\_S3\_DEV\_BUCKET\_\_, \_\_S3\_QA\_BUCKET\_\_, and \_\_S3\_PROD\_BUCKET\_\_.   
     
   Replace these strings with the name of the Amazon S3 buckets containing the dev, qa, and productions files, respectively (as shown on your S3 Console home page).
2. After you have created all the tables and fields in CAMPER, update the **tableProps** variable on the code page **program\_import\_module.html**. This variable contains information about the tables being imported and their fields. Each line in this variable starts with a table name, and then lists one or two fields by field id:

var QBU\_tableProps = {

    "\_dbid\_modules" : {moduleIdField: 11},

    "\_dbid\_modulette\_groups" : {moduleIdField:15, relatedItemField:9, relatedItemFieldName:"related\_module"},

    "\_dbid\_modulettes" : {moduleIdField:25, relatedItemField:15, relatedItemFieldName:"related\_modulette\_group"},

    "\_dbid\_tabs": {moduleIdField:18, relatedItemField:7, relatedItemFieldName:"related\_modulette"},

    "\_dbid\_tracks": {moduleIdField:18, relatedItemField:7, relatedItemFieldName:"related\_tab"},

    "\_dbid\_widget\_instances": {moduleIdField:23, relatedItemField:11, relatedItemFieldName:"related\_track"},

    "\_dbid\_files": {moduleIdField:36, relatedItemField:7, relatedItemFieldName:"related\_widget\_instance"},

    "\_dbid\_modulette\_rubrics": {relatedItemField:7, relatedItemFieldName:"rubric\_id"},

    "\_dbid\_modulette\_sub\_competencies": {moduleIdField:12, relatedItemField:6, relatedItemFieldName:"related\_modulette"}

}

Replace the field ids in the tableProps variable with the correct field ids for your CAMPER application:

* For the modules table entry, the moduleIdField value should be the field id of the field **id** in the modules table. For example, if the field id of **id** is 9, the value should be 9.
* For all the other tables, the moduleIdField value should be the field id of the field **related\_module\_id**
* The relatedItemField value should be the field id of the field with the name relatedItemFieldName. For example, for the widget\_instances table entry, the relatedItemField value should be the field id of the field **related\_track**.

### List of CAMPER COMMON pages:

note: The GitHub directory includes a page called **common\_dbspecific.js**. This page is for CAMPER, not CAMPER COMMON.

add\_tag.html

ancillary\_file\_upload.html

authoring\_page\_edit.html

authoring\_page\_new\_file.html

common.js

make\_program\_copy.html

preview\_dashboard\_v2.html

preview\_module.html

preview\_modulette.html

preview\_widget\_filelist.html

program\_import\_module.html

program\_management.html

publish\_search.html

review\_feedback.html

s3\_audio\_mapping.html

s3\_audio\_upload.html

s3\_copy\_ancillary\_files.html

s3\_copy\_dash\_progcust.html

s3\_copy\_image\_files\_v2.html

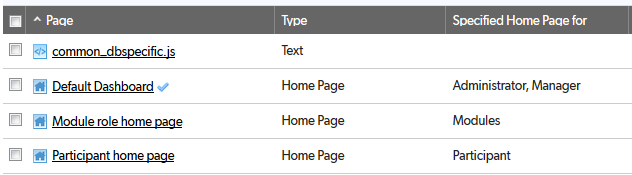
s3\_copy\_structures.html

s3\_copyfile.html

viewjson.html

## CAMPER

CAMPER has just one code page called **common\_dbspecific.js**, which defines the JavaScript variables used in other code pages. You must provide the correct values for all of these variables. The text for the page is provided below, or you can get it from the GitHub directory.   
  
Create the new page, copy and paste the text provided into it, then update the text as specified in the page text. See below for detailed examples.



**Page contents:**

// \*\*\*\*\*\*UPDATE THIS to your current app's Quick Base application database id in CAMPER and in program copies!!!!!

var QBU\_dbid = "\_\_DBID\_\_"; //e.g., "bjtmkaxyz"

// \*\*\*\*\*\*

var QBU\_dbid\_master = "\_\_CORE\_DBID\_\_"; // CAMPER application database id, e.g., "bjtmkaxyz". Do not update in program copies

var QBU\_db\_baseURL = "https://MYACCOUNT.quickbase.com/db/"; // SUBSTITUTE with the url for your QB account

var QBU\_juiceDomain\_dev = "http://[your dev web app url]/"; // SUBSTITUTE with the url for your JUICE server (development instance)

var QBU\_juiceDomain = QBU\_juiceDomain\_dev;

var QBU\_juiceDomain\_s = "https://[your dev web app url]/"; // SUBSTITUTE with the url for your JUICE server (development instance)--HTTPS version

var QBU\_filesDomain = "//[your s3 account]/"; // SUBSTITUTE with the url for your s3 account

**Substitutions:**

*var QBU\_dbid = "\_\_DBID\_\_"; //e.g., "bjtmkaxyz"*

Substitute \_\_DBID\_\_ with the database id of the QB application. For instance, if the URL of your CAMPER instance home page is *https://myaccount.quickbase.com/db/* ***bjtmkaxyz ,*** the variable value will be “bjtmkaxyz”. **Please note that this variable must be updated manually after making a program copy** (see documentation on how to create a custom program for more information).

*var QBU\_dbid\_master = "\_\_CORE\_DBID\_\_";*

Substitute \_\_CORE\_DBID\_\_ with the database id of your “master” QB application (the one you use to make program copies). Identical to \_\_DBID\_\_ when you create your original CAMPER instance .

*var QBU\_db\_baseURL = "https://MYACCOUNT.quickbase.com/db/";*

Substitute the url for your QB account.

*var QBU\_juiceDomain\_dev = "http:/dev.juiceyourskills.com/";*

Susbtitute the url for your JUICE server (development instance if you have more than one instance)

*var QBU\_juiceDomain\_s = "https://dev.juiceyourskills.com/";*

Susbtitute the url for your JUICE server (development instance if you have more than one instance)—note the use of “https” protocol!

*var QBU\_filesDomain = "//s3-us-west-2.amazonaws.com/";*

Substitute the url for your s3 account where the JUICE files are stored (if using a AMAZON S3 repository).

# Customizing the dashboard for CAMPER

To support JUICE previewing, publishing, and custom program functions, you need to add links to your CAMPER dashboard that load code pages from CAMPER COMMON.

Use a **text widget** to add the four links listed below. We recommend that you set them to open in a new window. In each link, make the following substitutions:

\_QUICKBASE\_URL\_ is the URL of your Quick Base instance

\_COMMON\_DBID\_  is the id for your instance of CAMPER COMMON

\_CAMPER\_ID\_ is the id for your instance of CAMPER.

Preview Dashboard

https://\_QUICKBASE\_URL\_/db/\_COMMON\_DBID\_?a=dbpage&pagename=preview\_dashboard\_v2.html&tableid=\_CAMPER\_ID\_

Publish Search

https://\_QUICKBASE\_URL\_/db/\_COMMON\_DBID\_?a=dbpage&pagename=publish\_search.html&tableid=\_CAMPER\_ID\_

Publish Dashboard to your dev server instance

https://\_QUICKBASE\_URL\_/db/\_COMMON\_DBID\_?a=dbpage&pagename=preview\_dashboard\_v2.html&tableid=\_CAMPER\_ID\_&publish=1

Publish Dashboard to your qa and prod server instance (if any)

https://\_QUICKBASE\_URL\_/db/\_COMMON\_DBID\_?a=dbpage&pagename=s3\_copy\_dash\_progcust.html&tableid=\_CAMPER\_ID\_&mode=dash

Use a **web page widget** to add this link:

https://\_QUICKBASE\_URL\_/db/\_COMMON\_DBID\_?a=dbpage&pagename=program\_management.html

Make these substitutions:

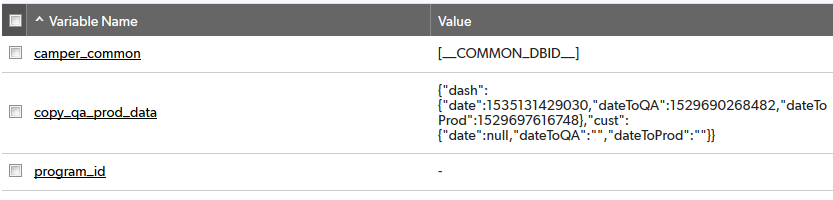
\_QUICKBASE\_URL\_ is the URL of your Quick Base instance

\_COMMON\_DBID\_  is the id for your instance of CAMPER COMMON.

If the URLs and your instance of CAMPER COMMON are correctly configured, each link in the text widget will load a page in a new tab.    The Program Management widget will show a Make Copy button.

# Defining variables for CAMPER

CAMPER uses three predefined variables. Set these up on the Variables page as follows. Replace \_COMMON\_DBID\_  with the id for your instance of CAMPER COMMON.



camper\_common

Use the database id for your instance of CAMPER COMMON

copy\_qa\_prod\_data

{"dash":{"date":1535131429030,"dateToQA":1529690268482,"dateToProd":1529697616748},"cust":{"date":null,"dateToQA":"","dateToProd":""}}

program\_id

-

(this reallyis just a hyphen!)

# Creating Tables

The basic steps for creating tables are as follows:

1. Review the appropriate section below to see what tables are required for CAMPER COMMON or CAMPER.
2. Locate the Table Descriptions directory in the documentation. This contains a Word doc for each table that includes a complete description of the table and its relationships.
3. Create the table and its relationships. Note that the tags field in the modules, modulettes, and ancillary files table must be hooked up to the tags table’s tag field in your instance of CAMPER COMMON.
4. Update manually created field formulas to include the correct field ids.
5. Create report views as desired.

## Updating field formulas

There are two types of field formulas in the CAMPER tables:

* Added automatically by the relationship building tool. These are in the fields called “Add\_xxxxx.”
* Manually created by the JUICE team.

The auto-built formulas do not need any updating. The formulas manually created by the JUICE team contain field ids that reference the fields in the original instance of CAMPER. They must be updated to use the correct field ids from your instance of CAMPER, or replaced with a field selected from the field dropdown.

Example:

This is from the Edit field in the Files table

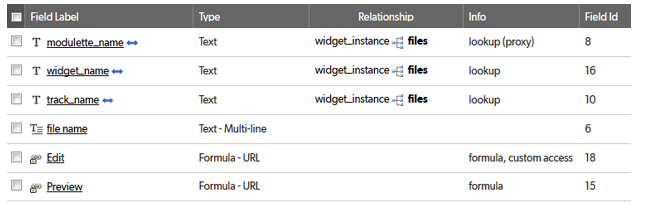
URLRoot() & "db/" & [camper\_common] & "?a=dbpage&pagename=authoring\_page\_edit.html&name="&[\_fid\_16]&"&fileRef="&[\_fid\_6]&"&recID="&[\_fid\_3]&"&tableid="& Dbid()

This formula contains field ids that reference the fields in the original instance of CAMPER. They must be updated to use the correct field ids or fields from your instance of CAMPER.

You can find the fields that the ids are referencing by looking at the list in the table description document.

In the example above, the first two field ids reference fields 16 and 6 in this table. These are the **widget\_name** and **field\_name** fields.

Find these fields in your own CAMPER instance’s field list for the files table and update the field ids in the formula to match, or find these field names in the field dropdown and select it.



## CAMPER COMMON tables overview

| **Table** | **What it defines** | **Required?** | **Relationships** |
| --- | --- | --- | --- |
| tags | List of tags used in the CAMPER applications | yes | Used to populate the tag field multi-selection lists in CAMPER modules, modulettes, and ancillary\_files tables |
| competencies | List of skill competencies.   Saved in user activity data and can be used for data analysis. | yes | sub\_competencies |
| sub\_competencies | List of subcompetencies. Each competency has an associated set of subcompetencies.  Saved in user activity data and can be used for data analysis. | yes | competencies  Also a lookup for the CAMPER modulette-sub\_competencies table |

## CAMPER tables overview

| **Table** | **What it defines** | **Required?** | **Relationships** |
| --- | --- | --- | --- |
| module | Basic attributes of JUICE modules. | yes | widgets, module\_instances, modulette\_groups  tag field looks up values from CAMPER COMMON. |
| modulette\_groups | Attributes of the mini-lesson groupings that appear on the module page | yes | modulettes, modules |
| modulettes | Attributes of JUICE mini-lessons | yes | modulette\_groups, tabs, modulette-sub\_competencies  tag field looks up values from CAMPER COMMON. |
| tabs | Tabs that are available in each mini-lesson | yes | modulettes, tracks |
| tracks | Tracks that are available on each tab | yes | tabs, widget\_instances |
| widget\_instances | Widget associated with each track | yes | widgets, tracks, files |
| files | File that should be played by each widget instance | yes | widget\_instances |
| widgets | List of available widgets | yes | modules, widget\_instances |
| ancillary files | Titles and tags for external files linked to from the JUICE mini-lessons | yes | tag field looks up values from CAMPER COMMON. |
| modulette\_sub\_competencies | Mapping of mini-lessons to a sub-competency list developed by the JUICE team (list resides in CAMPER Common). |  | modulettes, CAMPER COMMON:sub\_competencies |
| projects | List of projects (assignments) and learning goals. Data provided is customized for the *College for America* curriculum*.* |  | module\_instances |
| module\_instances | Mapping of modules to projects. Data provided is customized for the *College for America* curriculum*.* |  | modules, projects |
| rubrics | List of rubrics with their associated module id. Used to generate search results. Data provided is customized for the *College for America* curriculum*.* |  | modulette\_rubrics |
| modulette\_rubrics | Additional mapping of modulettes to rubrics. Used to generate search results. Data provided is customized for the *College for America* curriculum*.* |  | rubrics |
| audio\_maps Settings | Audio map that overrides “auto play” mode for narrations.  Currently not used except for testing. |  |  |

# Customizing table report views

The document *Suggested table report and form layouts for CAMPER* located in the Table descriptions directory provides screenshots of the layouts that the JUICE team felt were most helpful for supporting their content management workflow.

# Uploading data

After your tables are ready, you can upload data to them.

It’s very important to be careful when uploading data to CAMPER COMMON and CAMPER. If you do not follow the steps below, you can break the links between all of the tables.

The basic steps for uploading data are as follows:

1. Read the entire Uploading data section to familiarize yourself with the issues you can encounter.
2. Download the data from the CAMPER COMMON and JUICE Data directories in GitHub. Most tables will have 2 files—one for the initial upload, and one for the “real” upload.
3. Follow the instructions for each table below.

## Working around the import tag error in CAMPER

In CAMPER, the module, modulette, and ancillary files tables include a multi-select tag field that holds the list of tags that are used in JUICE Search. The tag list resides in the CAMPER COMMON app and has 184 items. In Quick Base, multi-select fields are limited to 100 items, but keeping the tag list in a different app works around that limitation.

Unfortunately, if you have already created CAMPER COMMON and connected the tag field to it, the tag field will have a “back-up list” that includes all the values that are in the CAMPER COMMON tags table. When you import, the import checks the length of the of the back-up list, and will throw an error since there are more than 100 items.

Work around this as follows:

1. Before you import to the module, modulette, or ancillary files table, go to the tag field for each table and change the input type to “From list.” To get to the tag field, go to the “Reports & Charts” for the table and click the “View Tags” report. Hover your cursor over the Tags field and click the menu arrow that appears, then click “Field Properties . . .” from the drop-down. You will see radio buttons for input type. Choose “From list”.
2. Select and delete all the items that are on the list.
3. Save your changes.
4. Do your imports.
5. Clear the tag list again if you are importing to the modulettes table.

The modules and ancillary\_files imports do not need the tag list cleared. The module import only adds a few tags on the first import and does not push you over the limit. The ancillary files table only requires one import.

1. Return to the tags field and re-set the Input type to “From another field.” Choose your CAMPER COMMON app, and the tags table tag field.
2. Save your changes

## Importing to CAMPER Record ID-linked tables without breaking the links

In CAMPER, the tables related to defining the hierarchy of the JUICE site content contain links to other CAMPER tables. The links for these tables are based on the Quick Base Record ID, which is the tables’ unique key field. The linked information is used to define the hierarchy of the JUICE content.

Example:

A file record contains links to the widget instance it is associated with. The widget instance contains links to the track, the track to the tab, the tab to the modulette, the modulette to the modulette group, and the modulette group to the module.

When you use the regular Quick Base import to create new data, the Record IDs are regenerated, which will break the links between the various linked tables. It is possible for you to recreate the links manually for some of the tables, but not all of them have enough identifying information for you to do this.

Instead, to preserve the Record IDs, use this **special import process**:

1. Create dummy records with the Record IDs that the data requires by importing the file called [table]\_data\_UPLOAD 1. *Example:* modules\_data\_UPLOAD 1
2. Choose to import into a table from a file
3. Choose the correct table
4. Choose the UPLOAD 1 data file provided
5. Click Import from File
6. On the Preview screen, set **Record ID** to “Do not import.”
7. Click Import (with Update)

This file is designed to generate dummy records that have Record IDs for each set of data in the real data file. It generates more records than are in the real data file because there are gaps in the Record IDs in the real data file. For example, the real data file may only have 13 records, but it includes a Record ID of 15. The UPLOAD 1 file will generate 15 records.

1. After you have completed the UPLOAD 1 uploads, update the dummy records with the real data by importing the real data file, called [table]\_data. Example: modules\_data

**VERY IMPORTANT!** This time, if you are not importing to the widgets table, keep the **Record ID** set to “To existing field.”

This will update all the records with Record IDs that exist in the real data file with the correct values.

1. Get rid of the extra rows that were created by UPLOAD 1 and not updated in the real data upload. Do this as follows:  
   1. Review the table descriptions below to find out which field has the text value “extra row”
   2. Add the field to your Quick Base table if necessary; in most cases it will be there
   3. Use the Quick Base search to find all the records with “extra row”
   4. Click the checkbox at the top of the checkbox column to select all of the rows, and click Delete to remove them.

**CAUTION!** Type the entire phrase “extra row” to find fields to delete. If you just type “extra,” you may some fields that should be retained!

### Record ID-Linked CAMPER table descriptions

Note: The tables marked “Required” are necessary to manage the core JUICE content.

| **Required or optional?** | **CAMPER table** | **What it defines** | **Field with text value “extra row”** | **Requires tag workaround** |
| --- | --- | --- | --- | --- |
| Required | module | Basic attributes of JUICE modules. | title | Yes—before first import |
| Required | modulette\_groups | Attributes of the mini-lesson groupings that appear on the module page | title | no |
| Required | modulettes | Attributes of JUICE mini-lessons | title | Yes—before BOTH imports |
| Required | tabs | Tabs that are available in each mini-lesson | tab\_header | no |
| Required | tracks | Tracks that are available on each tab | title | no |
| Required | widget\_instances | Widget associated with each track | files\_list | no |
| Required | files | File that should be played by each widget instance | file\_name | no |
| Optional | modulette\_sub\_competencies | Mapping of mini-lessons to a sub-competency list developed by the JUICE team (list resides in CAMPER Common). | Related modulette  \*\*This is set to 500, which will result in a blank modulette name in the table. In the “cleaning” step, remove the rows with blank modulette names and IDs. | no |
| Optional | projects  If you choose to use this table, use the module\_instances table as well. | List of projects (assignments) and learning goals. Data provided is customized for the *College for America* curriculum*.* | goal | no |

## Importing to other CAMPER tables

The rest of the CAMPER tables either stand on their own, or are linked by a field other than the Record ID. You don’t need to import dummy records for these tables, and can use the **standard import**:

1. Choose to import into a table from a file
2. Choose the correct table
3. Choose the data file provided
4. Click Import from File
5. On the Preview screen, set Record ID to “Do not import.” For the widget data upload, also set the **name** field to “Do not import”
6. Click Import (with Update)

### Other CAMPER table descriptions

Note: The table marked “Required” is necessary to manage the core JUICE content.

| **Required or optional?** | **Table** | **What it defines** | **Link type** | **Requires tag workaround** |
| --- | --- | --- | --- | --- |
| Required | ancillary files | Titles and tags for external files linked to from the JUICE mini-lessons | Stand alone | Yes—before import |
| Required | widgets  Set both name and record id to “do not import” | List of available widgets | Linked, but not on Record ID | no |
| Optional | audio\_maps Settings | Audio map that overrides “auto play” mode for narrations.  Currently not used except for testing. | Stand alone | no |
| Optional | rubrics | List of rubrics with their associated module id. Used to generate search results. Data provided is customized for the *College for America* curriculum*.* | Linked, but not on Record ID | no |
| Optional | modulette\_rubrics | Additional mapping of modulettes to rubrics. Used to generate search results. Data provided is customized for the *College for America* curriculum*.* | Linked, but not on Record ID | no |
| Optional | module\_instances  If you choose to use this table, use the projects table as well. | Mapping of modules to projects. Data provided is customized for the *College for America* curriculum*.* | Linked, but the Record IDs do not need to be preserved. | no |

## Importing data to CAMPER Common

In CAMPER Common, two of the tables contain links to other CAMPER Common tables and required the special import process described above. The other table is standalone and can use the standard import described above.

| **CAMPER Common table** | **What it defines** | **Type of import** |
| --- | --- | --- |
| tags | List of tags used in the CAMPER applications | Standard |
| competencies | List of skill competencies.   Saved in user activity data and can be used for data analysis. | Special  Does not require any extra rows to be removed after the first import. |
| sub\_competencies | List of subcompetencies. Each competency has an associated set of subcompetencies.  Saved in user activity data and can be used for data analysis. | Special  “extra row” field is sub\_competency |

# 

# User roles and groups

## Groups

CAMPER and CAMPER COMMON require the following groups:

| **Group** | **What it does** | **Who needs it** | **Apps accessible to this group** |
| --- | --- | --- | --- |
| CAMPER\_COMMON\_Access | Gives access to the shared code pages that are stored in CAMPER COMMON. | Everyone who will be editing, previewing, uploading, or publishing content from the CAMPER app or any alternate program app. | CAMPER COMMON  Role: Participant |
| JUICE\_Module\_View\* | Gives access to the CAMPER modules table for users in alternate JUICE program apps. | Users in alternate program apps who should be able to import modules from Core JUICE and are not members of the core JUICE program.  See the **Working in CAMPER** documentation on *Creating a new program for Admins* for more information. | CAMPER  Role: Modules |

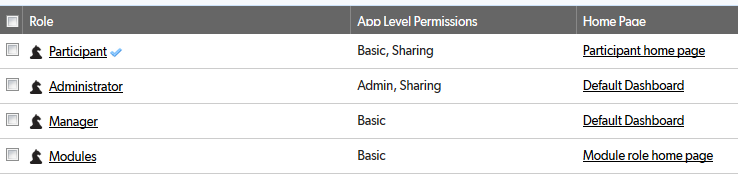
\*The JUICE\_Module\_View group is intended to allow members in other programs to see a list of modules on their program’s **Import Modules** page. It is not intended to give them access to Core JUICE CAMPER. We created a special “Modules” role for the group that had all tables removed from the table toolbar and a special note on the Dashboard telling members that they should not be in Core JUICE CAMPER.

Note that if you do not plan to create any alternate JUICE programs, or if all of your alternate programs will be managed by someone who has manager or admin rights to CAMPER, you do not need the JUICE\_Module\_View group.

## Roles

The JUICE team found it useful to establish these 4 roles. You can create whatever roles you feel meet the needs of your team. Here is a description of the JUICE team roles for reference.

The “Manager” role was developed to support alternate programs managed by someone outside the core team:



**Admins** were able to go anywhere and do anything across all programs.

**Managers** had the same rights as admins, except that they could not:

* Invite or manage users
* Customize tables or forms
* Create reports
* Change mini-lesson content file names from the authoring template or various reports.
* Change ancillary file names from edit file name or from reports.
* Update the competency or subcompetency lists in CAMPER COMMON
* Assign user tokens for the site to be able to talk to Quick Base.
* Assign users to groups.

**Participants** (Authors) could only access the participant view of the mini-lesson content file table in their own program. This view allowed them to Edit and Preview files, and set statuses.

The role of **Module** was used only for the JUICE\_Module\_View group. If a group member accessed CAMPER, they saw a view of the table toolbar with no tables showing, and a special version of the Dashboard warning them that they should not be there.

## Role summary

| **Action** | **Admin** | **Manager** | **Participant** |
| --- | --- | --- | --- |
| Access all tables in all programs and add or edit records | Yes | Own program | Own program  Files table only  Edit only |
| Customize forms, add reports, modify fields | Yes | No | No |
| Invite new users | Yes | No | No |
| Assign admin status to users | Yes | No | No |
| Create new roles | Yes | No | No |
| Edit locked file in files table  Note: The “author” Edit button is currently set to disallow admin access and is not visible to admins unless this permission is changed | Yes | Yes | No |
| Upload module and Try it images  (functionality to come) | Yes | Yes | No |
| Change file table names via authoring template (can overwrite CORE JUICE files easily) | No | No | No |
| Change file table names via reports (requires access to S3) | Yes | No | No |
| Change ancillary file names via reports (requires access to S3) | Yes | No | No |
| Change ancillary file names via edit ancillary file  (requires access to S3) | Yes | No | No |
| Access CAMPER COMMON | Yes | No | No |
| Add subcompetencies and competencies to CAMPER COMMON | Yes | No | No |
| Create user tokens and assign to programs | Yes  Billing manager only | No | No |
| Add users to JUICE\_Team\_Participant or JUICE\_Program\_Manager groups  to give access to CAMPER Common | Yes  Billing manager only | No | No |