

CRIPT OPERATION MANUAL



This page is intentionally left blank

Table of Contents

Introduction	4
What is CRIPT?.....	4
Why do you need CRIPT?.....	4
Overview of the CRIPT platform.....	4
Prerequisites.....	4
Contact Technical Support.....	5
Initial Setup	5
Registration with CRIPT.....	5
Login to CRIPT.....	6
Create a Project.....	6
Add Collaborators to a Project.....	6
Edit project.....	7
Delete a Project.....	7
Create a Collection.....	8
Edit a Collection.....	8
Delete a Collection.....	8
Help Menu.....	8
Share a Thought.....	9
Excel Uploader	10
Using the Excel uploader.....	10
Required Downloads.....	10
Excel Uploader.....	10
Excel Template.....	10
Column setup.....	10
Structure of Excel Sheets.....	10
Individual Excel Sheets.....	10
Python SDK	10
Quickstart.....	10
Sample Tasks.....	10
CRIPT Blog	10

Introduction

What is CRIPT?

The Community Resource for Innovation in Polymer Technology (CRIPT) provides a platform for researchers working in polymer science and engineering to capture and share data. CRIPT is hosted by a team at the Massachusetts Institute of Technology (MIT) along with collaborators in academia, industry, and government, with support from the Convergence Accelerator Program of the National Science Foundation (NSF).



Community Resource for Innovation in Polymer Technology

Prerequisites

Before attempting to access the CRIPT platform, users must verify their identity. Verification consists of obtaining a unique identification number from a third-party organization, Open Researcher and Contributor ID (ORCID). This ensures the user's identity while protecting their work.

ORCID

Open Researcher and Contributor ID (ORCID) is a global, not-for-profit organization. ORCID provides a persistent digital identifier (ORCID iD) that distinguishes one user from all other researchers.

To obtain an ORCID iD, visit the official ORCID website at www.orcid.org.

EDUCATIONAL/GOVERNMENTAL ORGANIZATIONS

Some unique identification credentials provided by universities and governmental agencies can be used to access the CRIPT platform. These must be obtained from the user's institution.

Contact Technical Support

This manual was created to instruct users on the proper usage of the CRIPT platform. Additional support may also be found using the following resources:

- The CRIPT blog can be found on the CRIPT platform home page at blog.criptapp.org. It contains helpful usage tips.
- “Share a Thought” links are featured on many CRIPT platform pages. Clicking on these links allow users to comment and voice their concerns regarding CRIPT platform usage.
- Email requests for help can be sent to CRIPT@mit.edu.

Chapter 1

Initial Setup

Registration with CRIPT

Before registering with CRIPT, the user will require an ORCID identification number (ID). For more information, see “Prerequisites” on page 1-4 in the Introduction.

1. Navigate to the CRIPT website, <https://criptapp.org>.
2. Select Login/Register (1) on the top right corner of the page.
3. Use drop-down (2) to select your institution. If your institution is not listed, the consent form can be downloaded from the link (3). The link to the form can be downloaded from the hyperlink, **Institutional consent form**.
4. Complete the form and email back to cript@mit.edu.
5. Select Connect ORCID iD (4) to launch the ORCID sign-in page.
6. Enter the user's Email ID or 16-digit ORCID iD (5) and password (6), and select SIGN IN (7).
7. If lacking an ORCID iD, select Register now (8).
8. Enter user information and select NEXT (9).
9. Verify that there is not an ORCID iD associated with user name by using the scroll bar (10) to scan through names.
10. Once confirmed, select NONE OF THESE ARE ME, CONTINUE WITH REGISTRATION (11).
11. Create ORCID iD by entering a Password (12), Confirm password (13) by entering it again.
12. An optional checkbox (14) can be selected for quarterly notifications.
13. Select NEXT (15) to continue.
14. Set visibility settings (16) and Terms of Use (17).
15. Select the reCAPTCHA box (18) and then select REGISTER (19).
16. The globus site will launch. Select Continue (20).
17. Enter Email address (21) and Organization (22).
18. Select which purpose the account will be used for (23).
19. Select Terms of Service (24) to accept Privacy Policy and then select Continue (25).
20. The ORCID Authorize access page will open, displaying the user's ORCID credentials. Selecting Authorize access (26) will launch the CRIPT home page.

Login to CRIPT

The CRIPT home page allows the user to create new projects, search for existing projects, and find helpful tips on the CRIPT platform.

CR IPT HOME PAGE

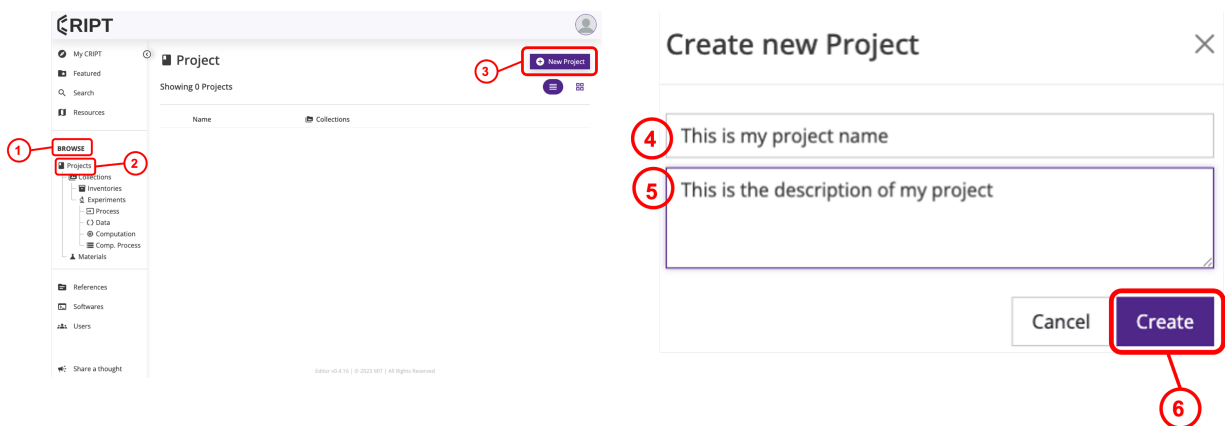
The CR IPT home page allows the user to navigate through CR IPT projects, create new projects, and find helpful tips on the CR IPT platform.

Create a Project

The first step in working with user data in CR IPT is to create a project. A project is an organizational tool that will contain a collection of thematically related work that is typically a collaboration between a number of individuals and can lead to a series of scientific publications. The project will hold a set of collections, which are publishable groupings of research (i.e. papers) as well as a set of materials that are used between research performed within the project. By default, the user project will not be public when first created. Therefore, it can only be viewed and modified by collaborators that are added to the project.

1. Under the Browse drop-down (1), select Projects (2).
2. Select New project (3).
3. Enter a valid project Name (4).
4. Enter a Description (5) for the project.
5. Select Create (6).

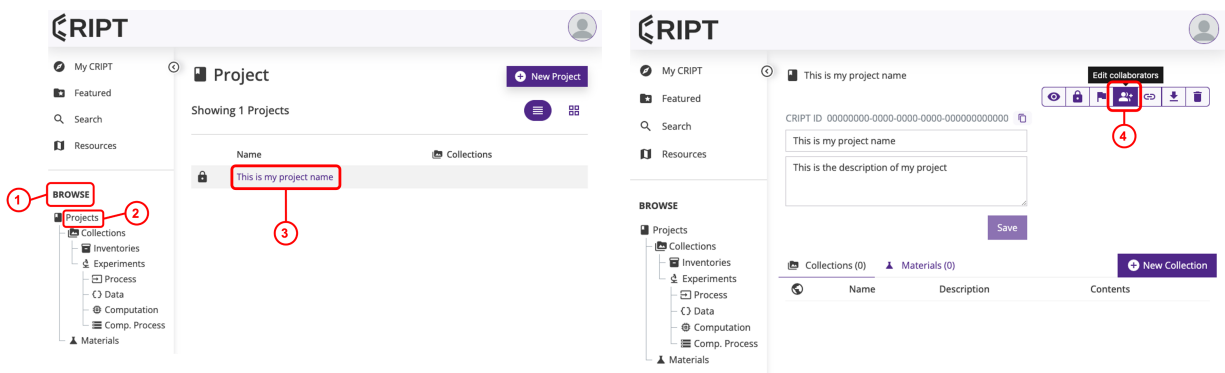
NOTE: New projects are private by default and can only be viewed and modified by collaborators that are associated with the project.



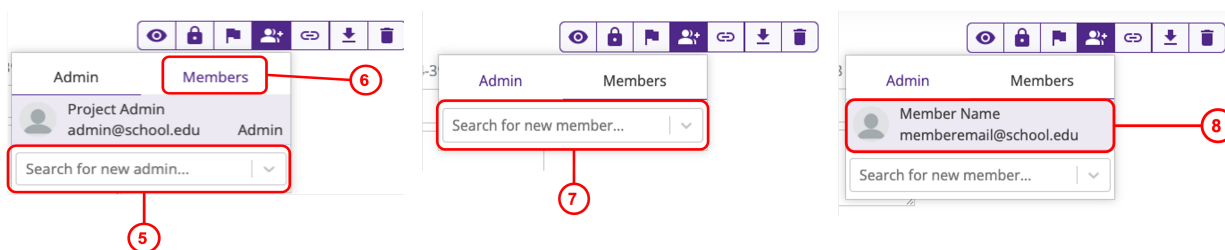
Add Collaborators to a Project

Once a project is created, specify which users can contribute to that project. Add all collaborators, including principal investigators. If a collaborator is not added, they will not be able to view or modify any data in the project until it is public.

1. Under Browse drop-down (1), select Projects (2).
2. Select project name (3)
3. Select Edit Collaborators icon (4) in top right corner

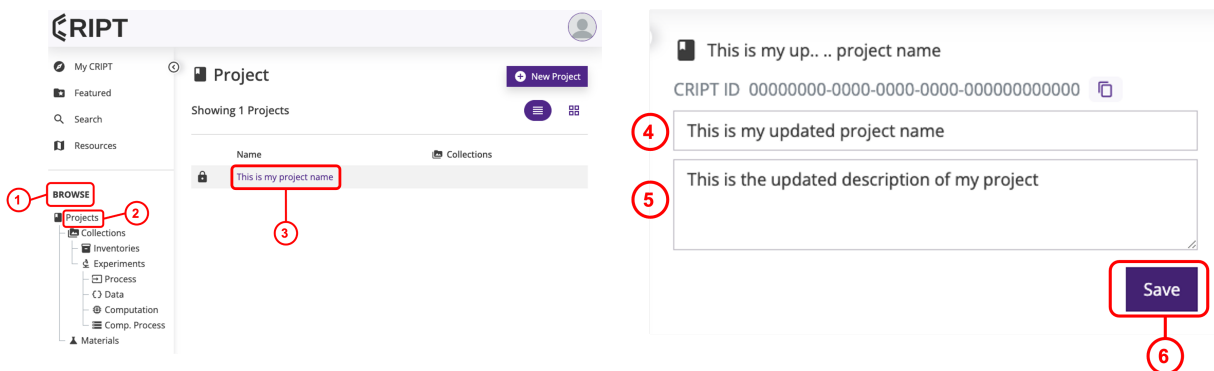


4. To add additional administrators to the project, select “Search for a new admin...” (5) and locate the user name in the drop-down list
5. To add new members, select members tab (6)
6. Select “Search for a new member...” (7) input or locate and select the user name in the drop-down list to add them to the project.
7. Once the collaborator is successfully added to a project, their name will appear under the members tab (8).



Edit project

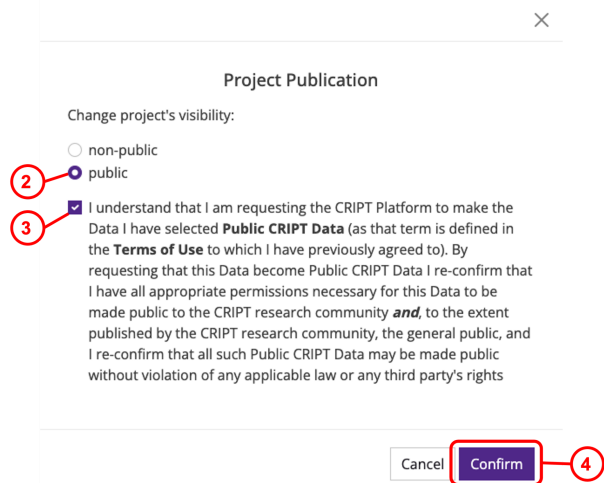
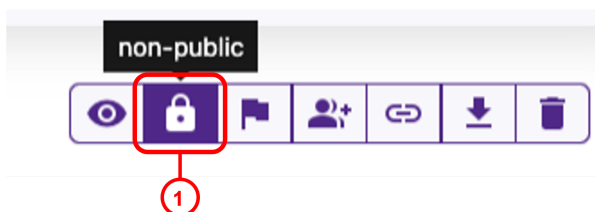
1. Navigate to project: select Browse drop-down (1) and select Projects (2).
2. Select the project to be edited (3).
3. Users can edit the manual Name (4), project description (5),
4. Select Save (6) to save changes.



NOTE: New projects are private by default and can only be viewed and modified by collaborators that are associated with the project.

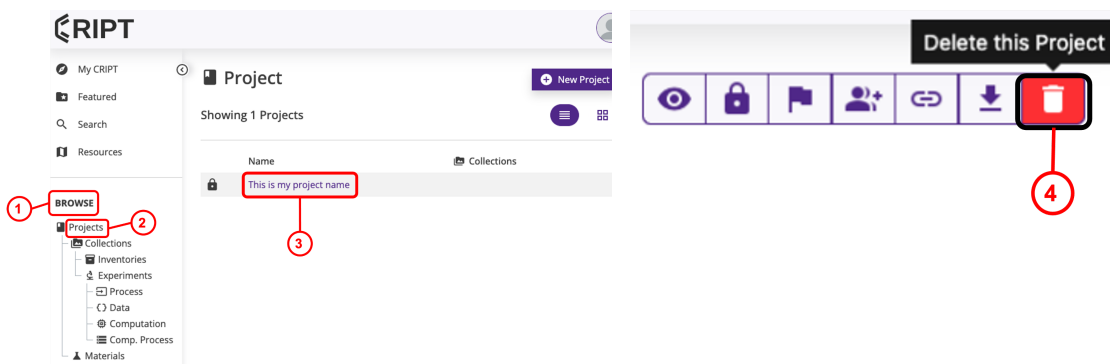
Non-public change

1. Navigate project. For more information, see “Edit project” on page 1-14.
2. Select non-public lock button (1) in top right corner menu
3. Select public (2)
4. Read statement and select box (3) to acknowledge terms of use
5. Select confirm (4)

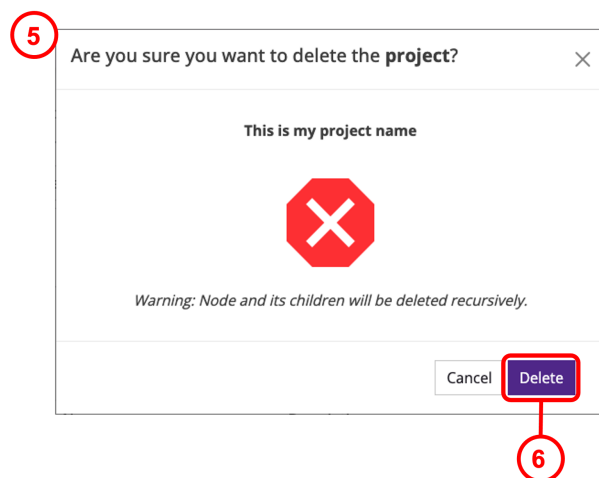


Delete a Project

1. Under Browse drop-down (1) and select Projects (2).
2. Select the project to be deleted (3).
3. In the top right corner, select trash can icon (4).



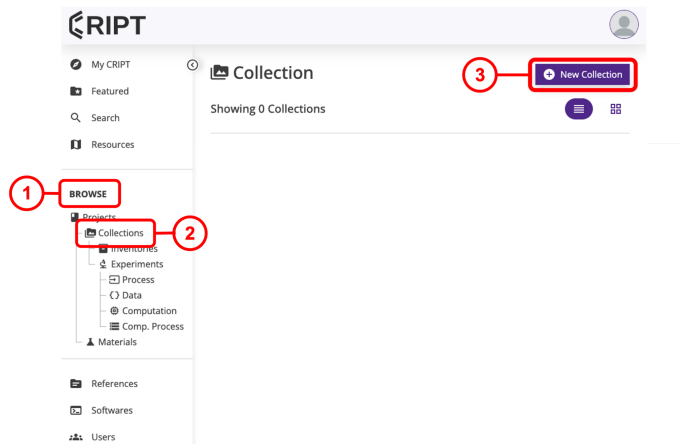
4. A pop up window will appear to confirm deletion (5).
5. Select delete (6).



Create a Collection

Each project within CRIPT holds both Materials and Collections. Collections are groupings of data that represent publishable units of research (i.e. one paper). Collections are the organizational level used to make data viewable to the public. All collections are by default non-public; the general public cannot view the users collection until it is made public at the project level. For more information, see “Non-public change” on page 1-16.

1. Under Browse drop-down (1), select Collections (2).
2. Select New Collection (3).

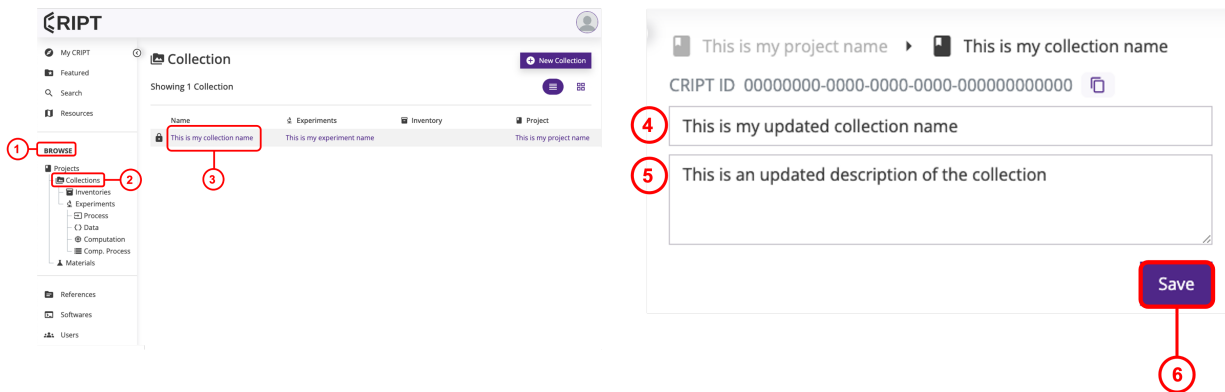


3. Select project (4) from dropdown menu to create new collection under
4. Enter Name (5) for New Collection.
5. Enter Description (6) for New Collection and select Create (7) to save.

Field	Value
Select Project	This is my project name
Name	This is my collection name
Notes	This is a description of the collection

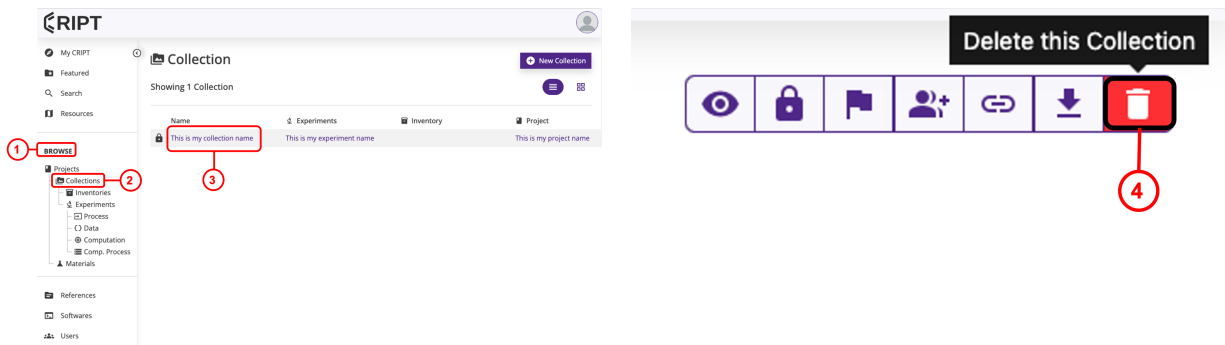
Edit a Collection

1. Under Browse drop-down (1), select Collections (2).
2. Select collection to be edited (3).
3. Users can edit the collection name (4) and collection description (5).
4. Select Save (6) to save changes.

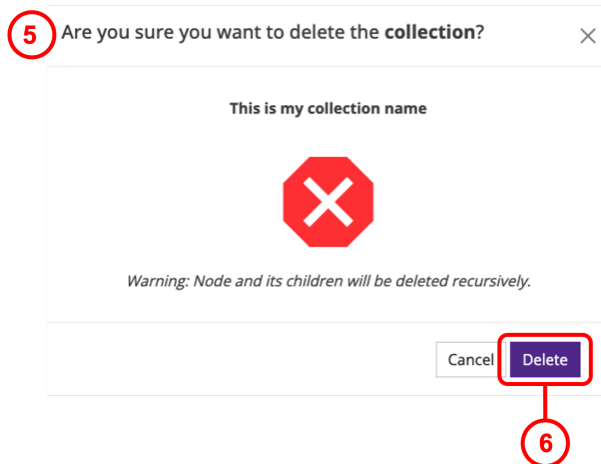


Delete a Collection

1. Under Browse drop-down (1) and select Collections (2).
2. Select the Collection to be deleted (3).
3. In the top right corner, select trash can icon (4).



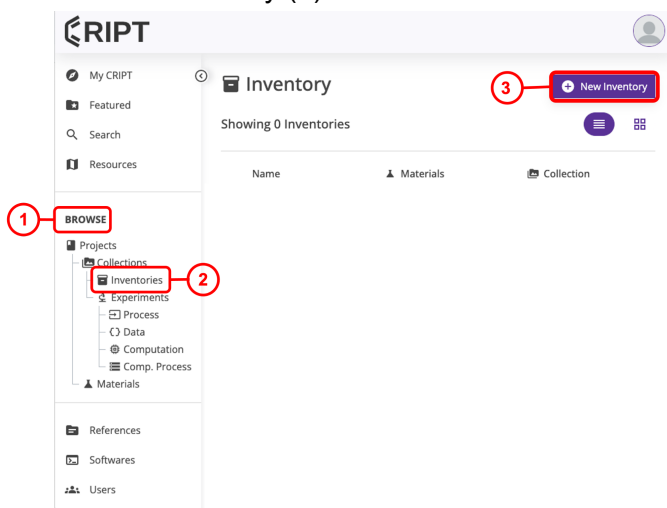
4. A pop up window will appear to confirm deletion (5)
5. Click delete (6)



Create an Inventory

Each collection can have multiple inventories. An inventory represents groups of materials, such as a class of monomers.

1. Under the Browse drop-down (1), select Inventories (2).
2. Select New Inventory (3).



3. Select collection (4) from dropdown menu to create new inventory under
4. Enter a valid inventory name (5).
5. Enter a Description (6) for the inventory.
6. Select Create (7).

Create new Inventory ×

4 Select Collection

5 Name

6 Notes

Cancel Create

7

The form is titled 'Create new Inventory' with a close button. It contains three input fields: 'Select Collection' (a dropdown menu), 'Name', and 'Notes'. Below the fields are two buttons: 'Cancel' and 'Create'. The 'Create' button is highlighted with a red circle and the number '7'.

Edit an Inventory

1. Under Browse drop-down (1), select Inventory (2).

2. Select inventory to be edited (3).
3. Users can edit the inventory name (4) and inventory description (5).
4. Select Save (6) to save changes.

Experiments

Each collection can include experiments which serve to organize procedures and processes. Created experiments contain sections for Process, Data, Computation, and Computational Process.

Note: when defining a new experiment, it is recommended to first start with the materials, followed by process, product material, and finishing with characterization data.

Processes

Processes contain the procedural details of the experiments carried out. After a new process is created, details regarding the experiment, such as conditions and equipment can be added to the process.

+ Process

🔗 process name
🗑️

Save

🔗 Prerequisite Processes (0)
▼

👤 Ingredients (0)
▼

🔧 Conditions (0)
▼

📄 Properties (0)
▼

🔧 Equipments (0)
▼

📄 Keywords (0)
▼

👤 Products (0)
▼

🗑️ Waste (0)
▼

Data

Data can be attached to either a material or process, and contains the raw and processed data files produced by either the materials or processes. Data type can be identified using the dropdown menu, and a data file can be uploaded.

+ Data

{ } myExperimentNMR
🗑️

Save

📁 Files (0)
^

Name	Size
Choose File	No file chosen



+


👤 Related Materials (0)
▼


Computation



Computation includes the information related to the creation or transformation of data. The steps to calculate material properties derived from experimental data.



+ ComputationProcess



 myCompProcess 



Notes 



extension 

 Input & Output Data 

 Properties (0) 

 Ingredients (0) 

 Conditions (0) 

 Software Configurations (0) 

Computation Process

Computational process is used for simulations of reactions or physical transformations on virtual materials.

Material

Projects contain both collections and materials. Materials contain identifiers and property data of a chemical, mixture or substance. When creating a new experiment, it is recommended to start with the materials.

Creating a New Material

This is my up... project name ▶ ▲ MyMaterial

CRIPID ID 0000000-0000-0000-0000-000000000000

MyMaterial

Notes

Save

Components

Search a Material +

Identifiers

Search an Identifier +

Properties (0) ^

Search a Property +

Keywords

Search a Keyword +

Draw Molecule

Components: the constituents comprising the new material. If selecting a component that was not created within the current project, a window will appear prompting the user to copy the material information into the current project. Scroll to the bottom of the window and select “yes” to confirm copying of the material information to the current project.

Action required

The material Pentyl 2,2-bis(4-chlorophenyl)acetate is not in your project.

Would you like to copy it?

Identifiers: Identifiers such as chemical names or SMILES strings can be added.

Properties: Material properties such as density and molecular mass can be added.

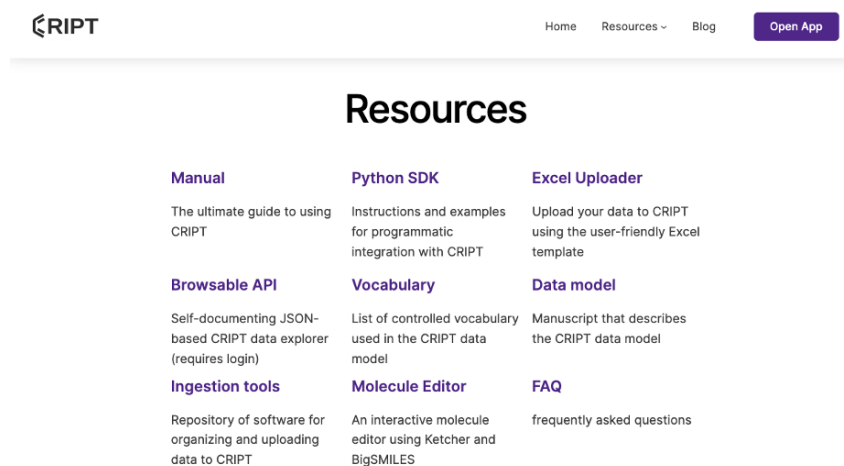
Draw molecule: Molecular structure of the molecule can be drawn by selecting the “Draw” button.

Keywords: general identifiers, such as “elastomer” and “network” that the material can be categorized under.

Note: Each material can only be associated with the project that it was created within. A user cannot reference a material that is created under another project, and must copy the material information to the project that it is to be associated with.

Resources Menu

The Resources page provides many different resources to assist users in maximizing the utility of the CRIPT platform. Most features of the resources menu can be accessed from the CRIPT Home Page or the CRIPT Landing Page.



Manual	Python SDK	Excel Uploader
The ultimate guide to using CRIPT	Instructions and examples for programmatic integration with CRIPT	Upload your data to CRIPT using the user-friendly Excel template
Browsable API	Vocabulary	Data model
Self-documenting JSON-based CRIPT data explorer (requires login)	List of controlled vocabulary used in the CRIPT data model	Manuscript that describes the CRIPT data model
Ingestion tools	Molecule Editor	FAQ
Repository of software for organizing and uploading data to CRIPT	An interactive molecule editor using Ketcher and BigSMILES	frequently asked questions

CRIPT Manual

Comprehensive guide to using the CRIPT application interface.

Browsable API

Self-documenting JSON-based CRIPT data explorer (requires login).

Ingestion tools

Repository of software for organizing and uploading data to CRIPT.

Python SDK

Instructions and examples for programmatic integration with CRIPT via the CRIPT python package `cript.py`. For more information, see Python SDK.

Vocabulary

List of controlled vocabulary used in the CRIPT data model.

Molecule Editor

An interactive molecule editor using Ketcher and BigSMILES

Excel Uploader

Upload data to CRIPT in tabular form using the Excel template. For more information, see "Excel Uploader" on page 2-2.

Data model

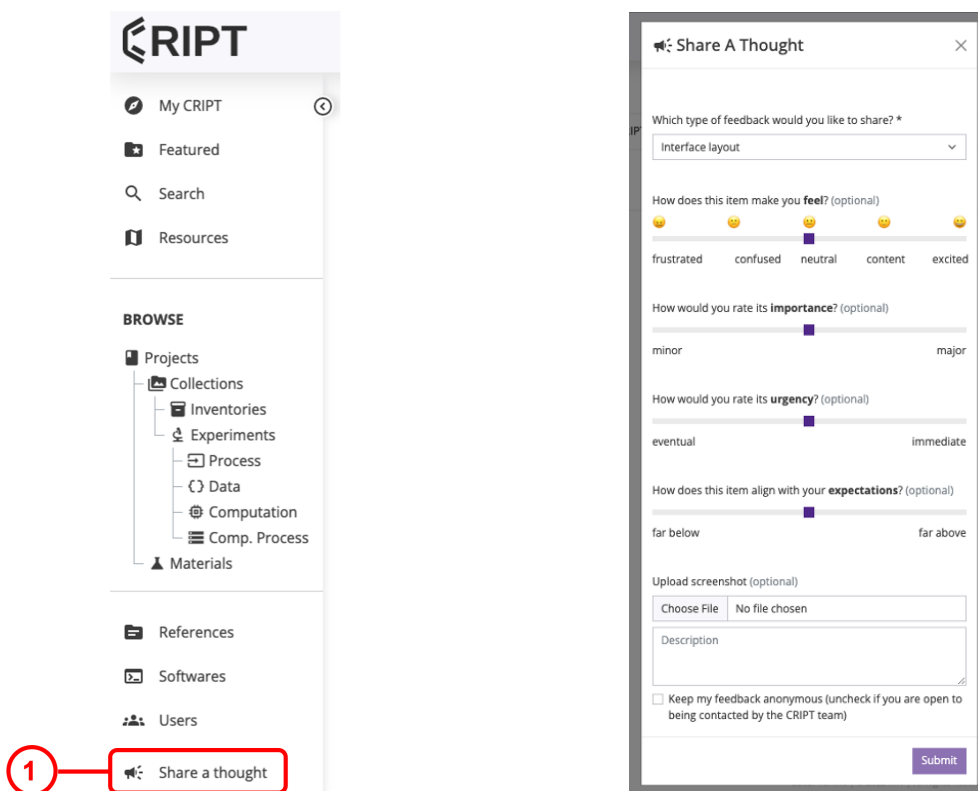
Manuscript that describes the CRIPT data model.

Frequently asked questions (FAQs)

Answers to common questions about the CRIPT platform.

Share a Thought

1. The Share a thought button (1) allows the user to provide user feedback on the CRIPT platform. The feedback collected is used to refine the CRIPT platform experience to better serve the users. Any feedback is greatly appreciated by the CRIPT team. The Share a thought button can be found in the lower left of most pages while navigating through the CRIPT web app.
2. Select the type of feedback from the drop-down.
3. Optional: Select and drag the square slider to select how you feel, rate the importance, and urgency.
4. Optional: you may upload a screenshot and describe your feedback in more detail.
5. Optional: you can choose to submit your feedback anonymously, or leave it unchecked if you consent to being contacted to discuss your suggestions with the CRIPT team.
6. Select Submit to send feedback.



Chapter 2

Excel Uploader

Using the Excel Uploader

Required Downloads

Uploading data to CRIPT using the Excel Uploader requires the following:

- Download the Excel Template
- Download the CRIPT Excel Uploader executable
- Locate and record the user specific API token.

Excel template

1. Navigate to CRIPT Home > Help > Excel Uploader > CRIPT Excel Template, or use the following hyperlink "[CRIPT Excel Template](#)".
2. Download file and save to a known working location.
3. For information on using the template, see "Excel Template" on page 2-12.

CRIPT Excel Uploader Executable

1. Navigate to CRIPT Home > Help > Excel Uploader > CRIPT Excel Uploader executable or use the following hyperlink "[CRIPT Excel Uploader executable](#)".
2. Download file and save to a known working location.
3. For information on using the Excel Uploader, see "Excel Uploader" on page 2-2.

Excel Uploader

1. Locate the save the CRIPT Excel Uploader file and run the file.

NOTE: All fields must be filled out before upload can begin.

HOST

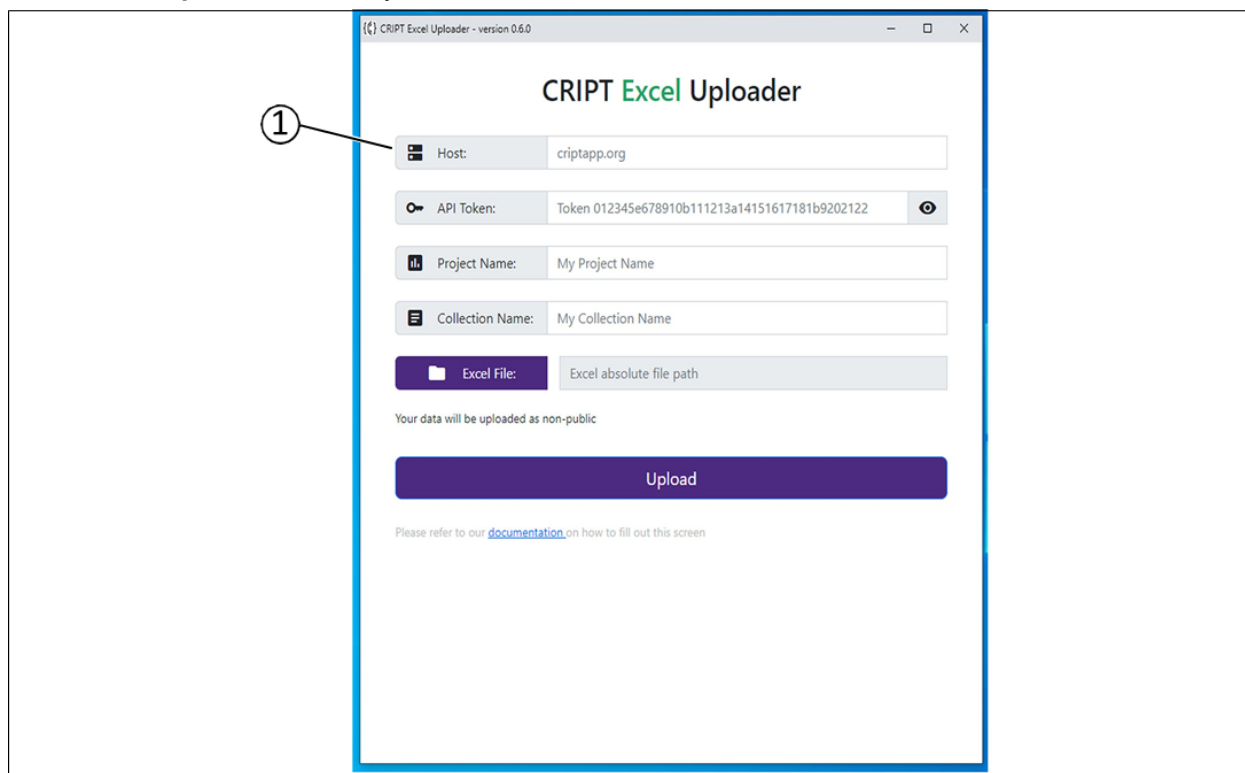
Host (1) indicates the CRIPT instance that you want to upload your data to, whether that is CRIPT or a private instance. For most users, the Host will be **criptapp.org**.

NOTE: This field will need to be entered.

Example: Host: criptapp.org

However, if any user wants to connect to their own private instance of CRIPT, they can easily do that by changing the host to whatever URL they are using.

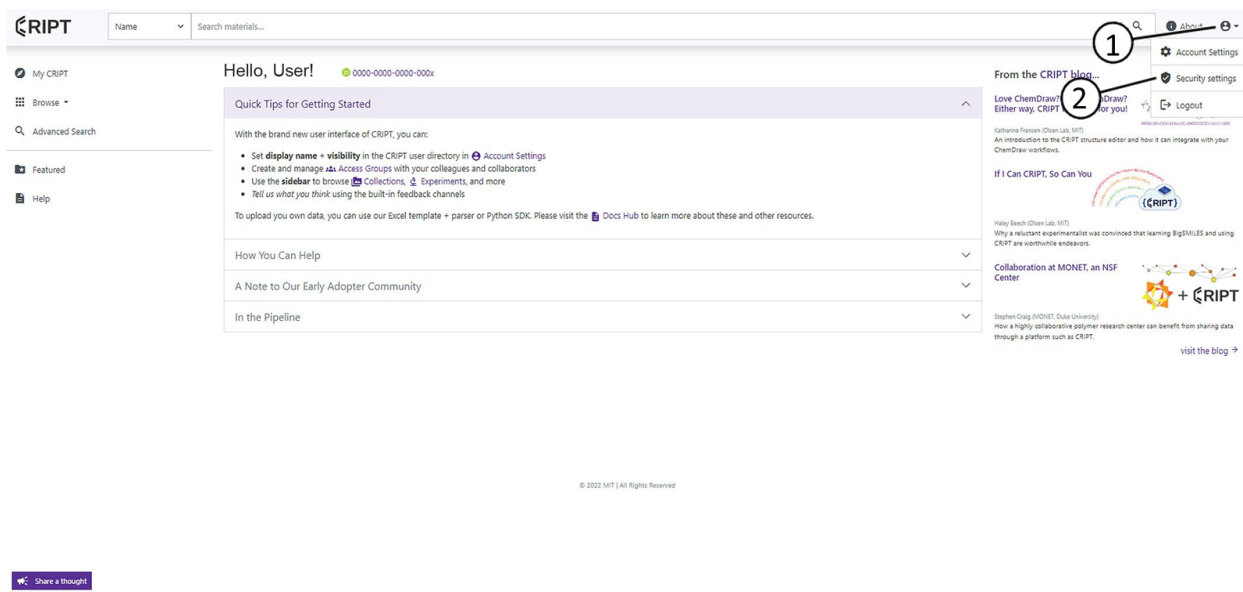
Example: Host: myPrivateWebsite.com



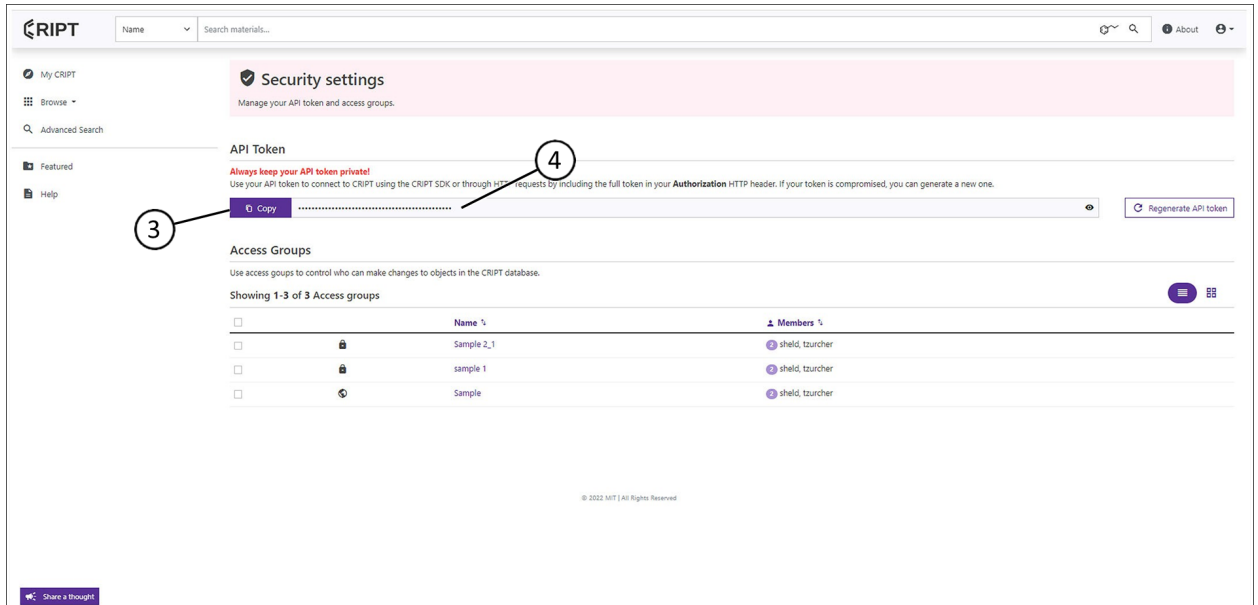
API TOKEN

The API Token is needed to authenticate the user before saving data.

1. On the CRIPT home page, select the user icon (1) and Security settings (2)



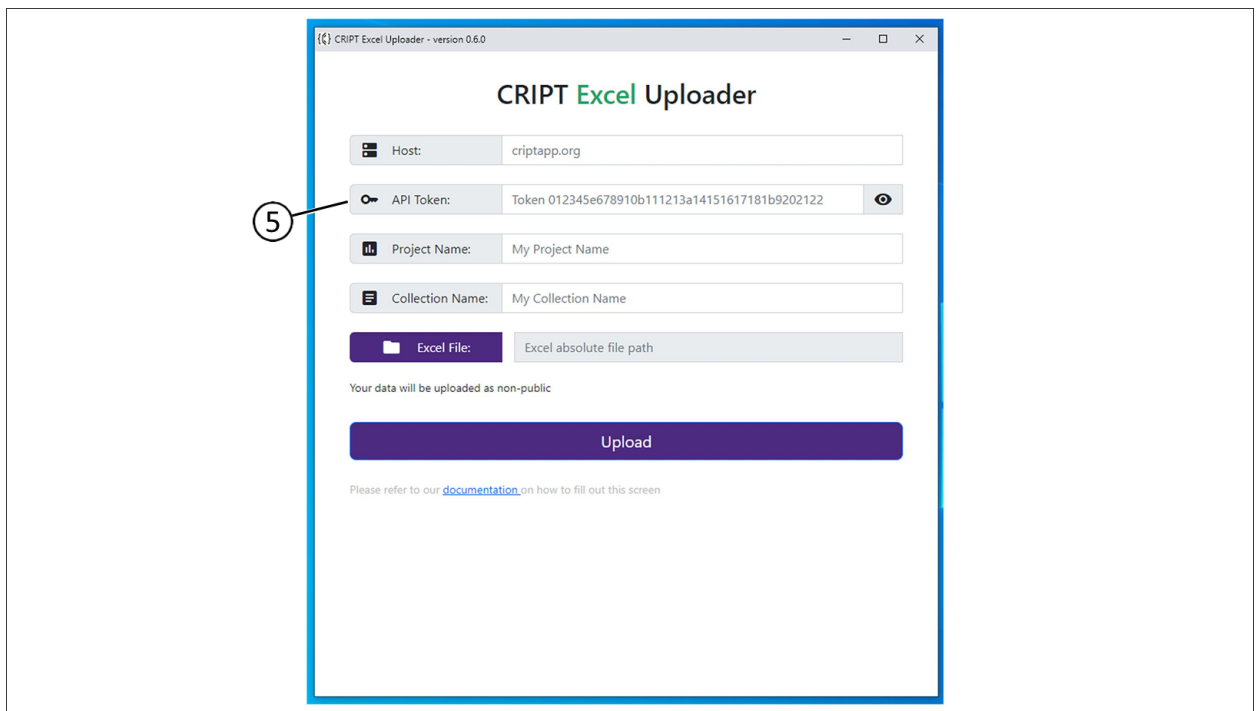
2. Select the Copy button (3) next to the API Token (4) to copy it to the clipboard.



3. Paste API Token into the API Token field (5).

NOTE: The word “Token” in front of the random characters is part of the token as well.

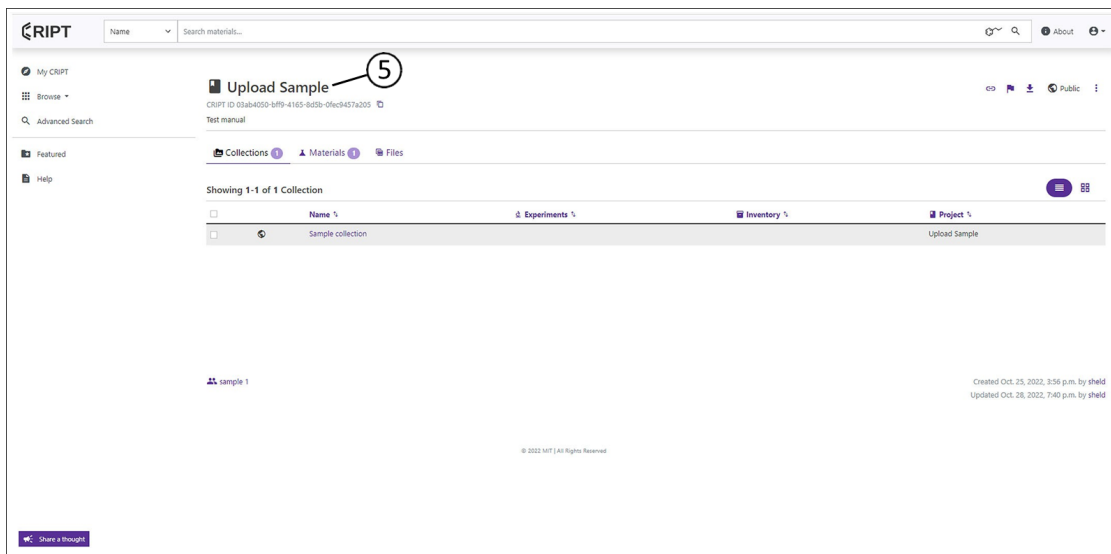
Example: API Token: Token 4abc478b25e30766652f76103b978349c4c4b214



PROJECT

A Project can be thought of as a bunch of folders with each containing experiments that contribute to a single project. A Project is needed for the software to know which project these experiments are for, and each Collection belongs to a Project. For more information, see “Create a Project” on page 1-10.

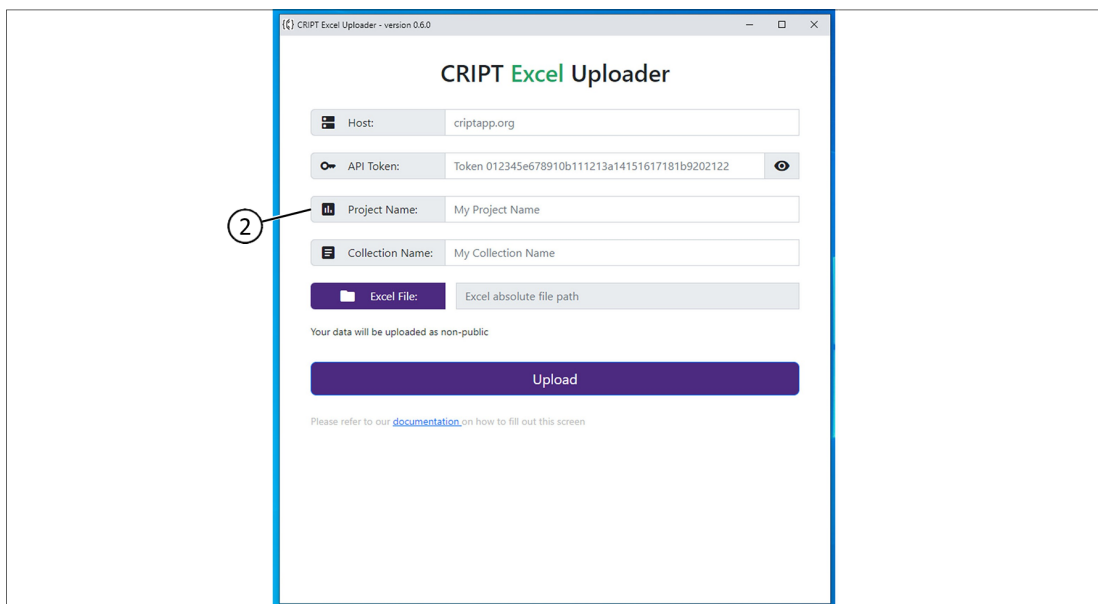
1. Navigate to the Projects tab within CRIPT.
2. Locate and select the project associated with the data to be uploaded.
3. Copy the Project name (1).



4. Paste Project name into the Project Name field (2).

Example:

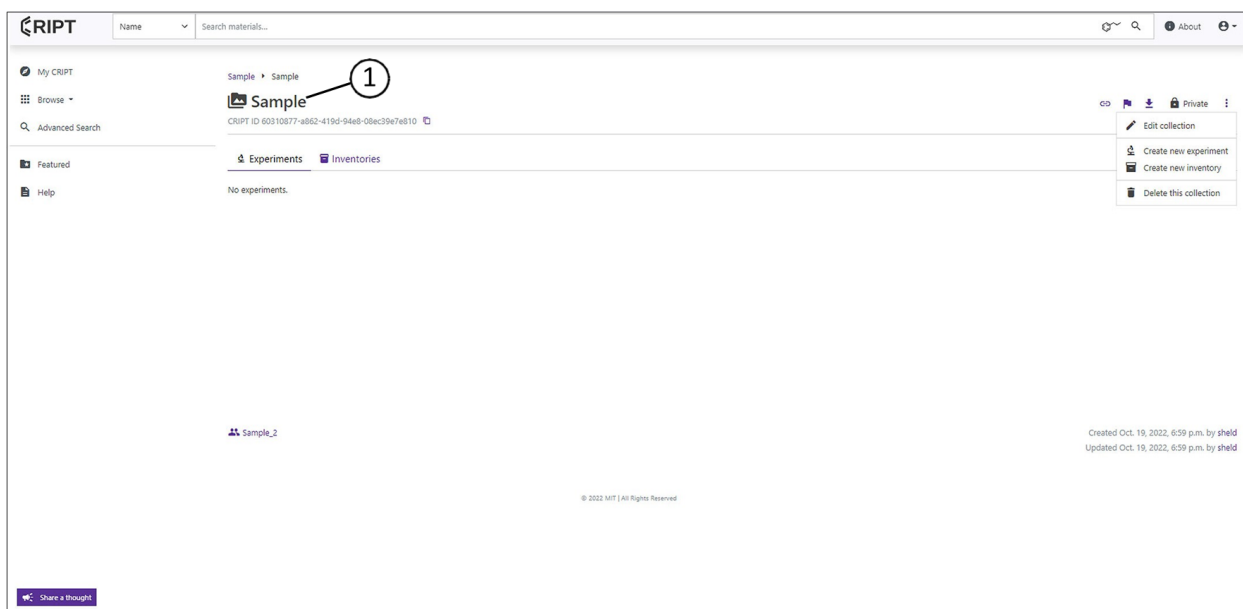
Project Name: Upload Sample



COLLECTION

A Collection can be thought of as a binder filled with experiments. The entire Excel file will become a collection within the CRIPT Platform. For more information, see “Create a Collection” on page 1-20.

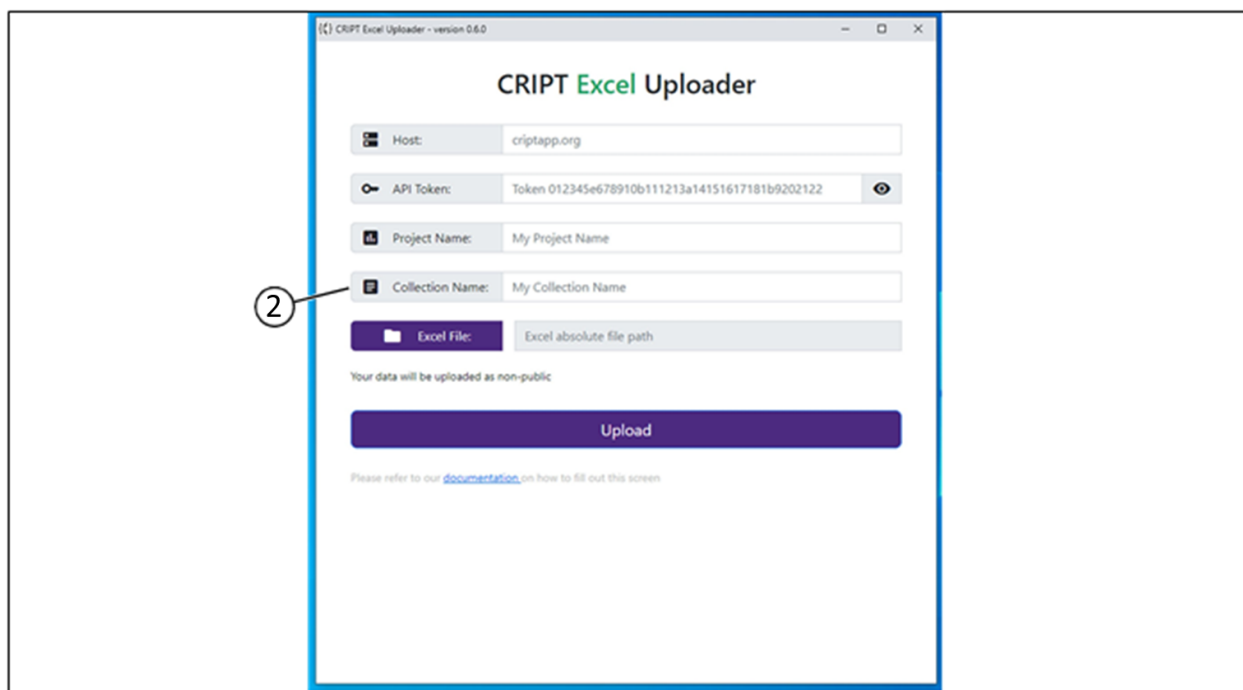
1. Navigate to the Collections tab within CRIPT.
2. Locate and select the collection associated with the data to be uploaded.
3. Copy the Collection name (1).



4. Paste Collection name into the Collection Name field (2).

Example:

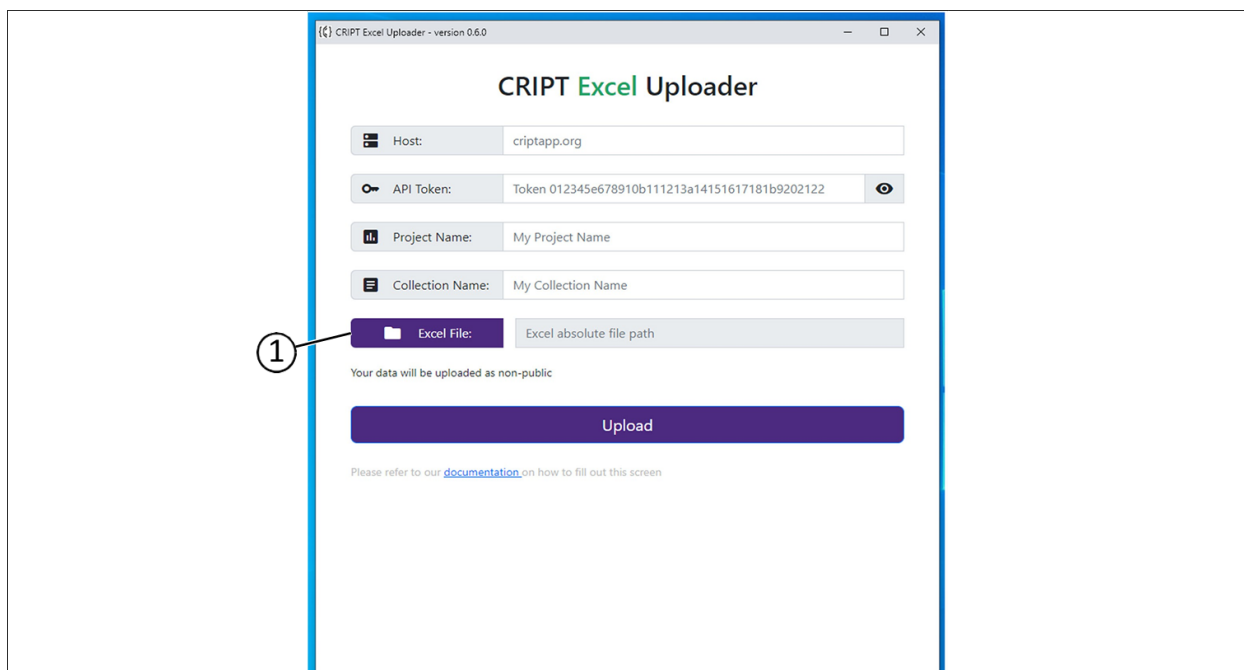
Collection Name: Sample



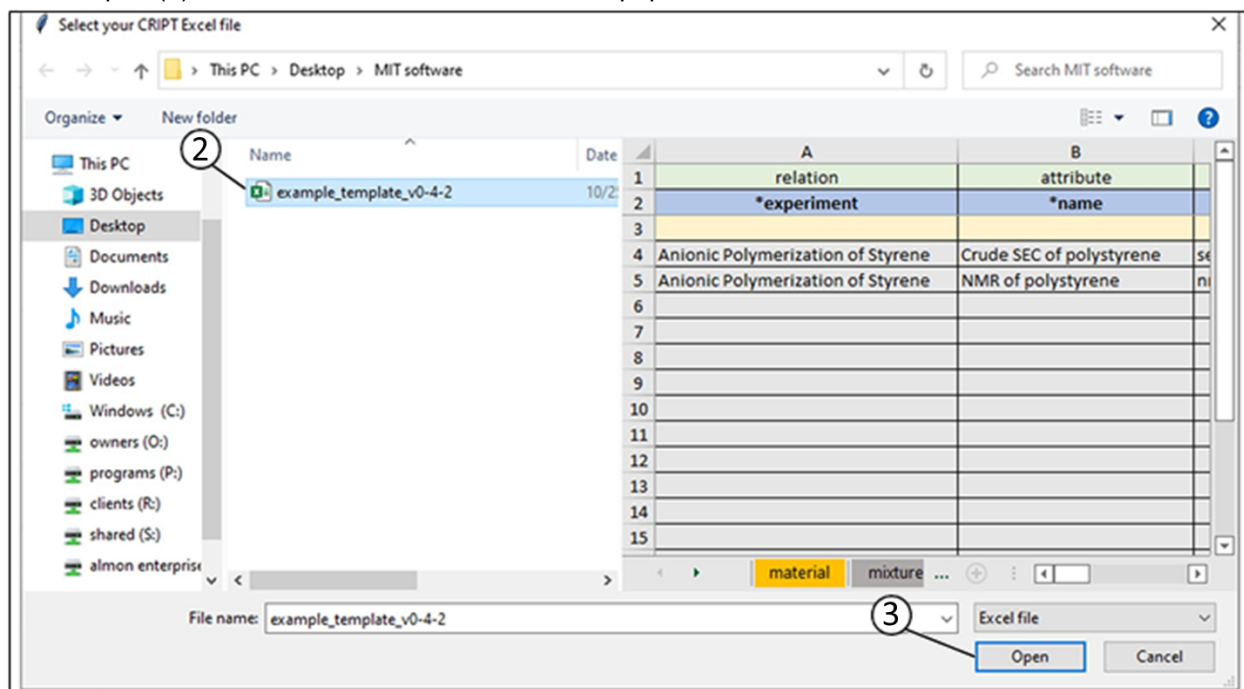
EXCEL FILE

This field allows the user to search for the file to upload.

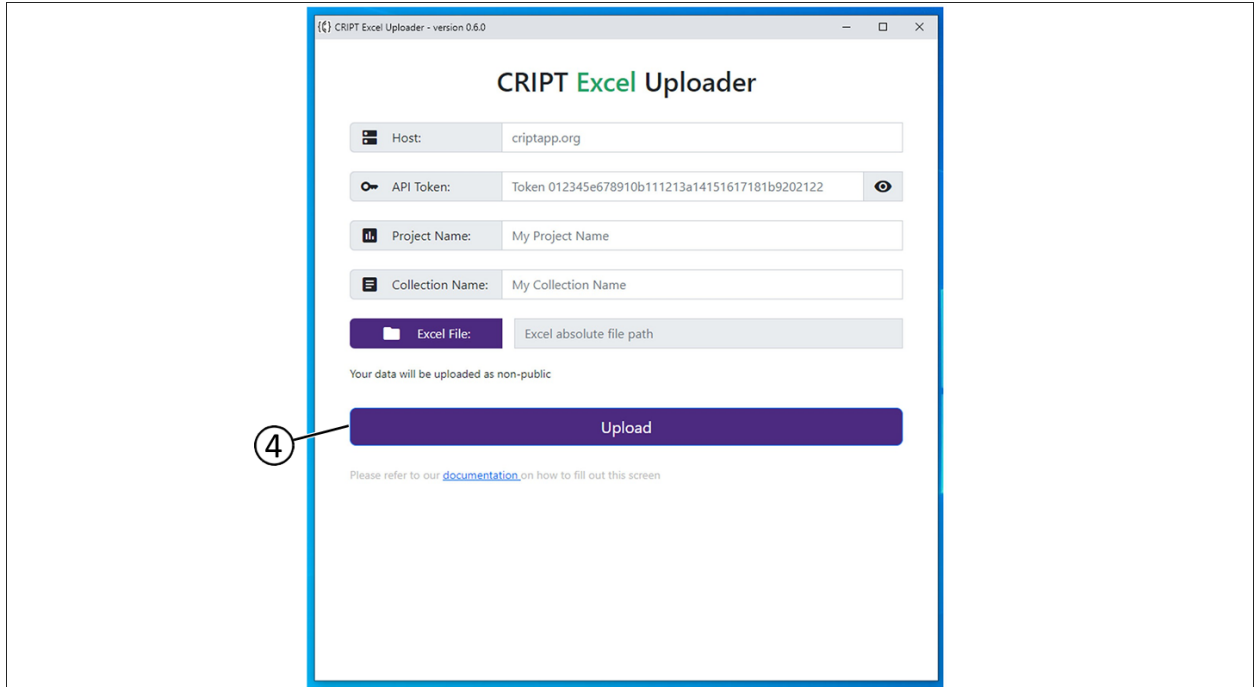
1. Select Excel File (1).



2. Locate file (2) to be uploaded and selected.
3. Select Open (3) and the Excel File location will auto populate the field.



4. Select the Upload button (4).



Excel Template

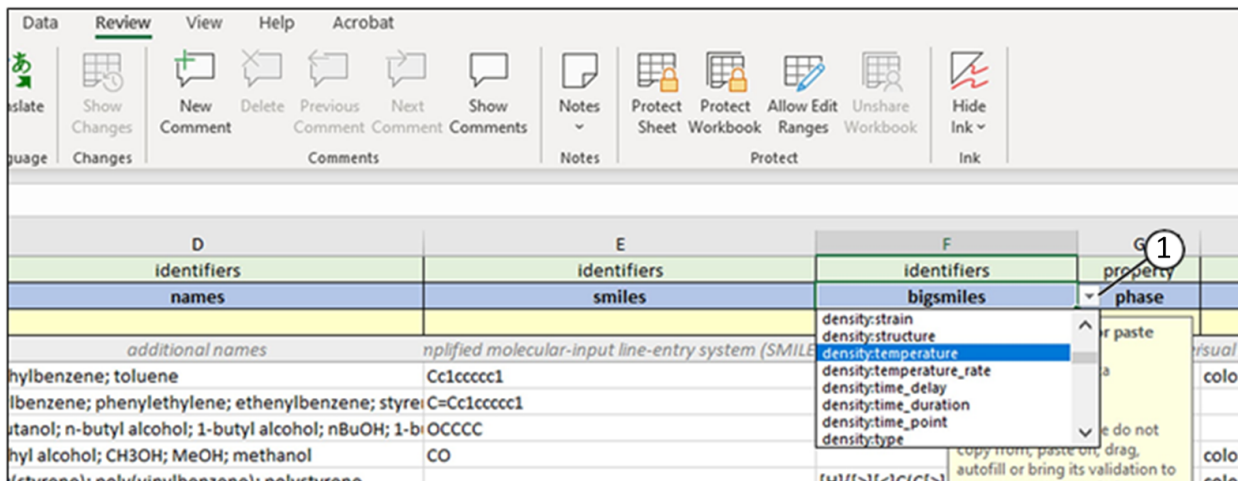
Column setup

There are two different methods for creating the column structure for data entry.

Autofill

1. Open Excel Template.
2. The Excel Template will have some of the columns started for reference points.
3. Create new columns by using the drop-downs (1).

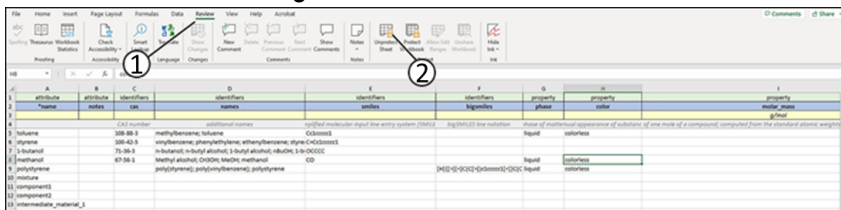
NOTE: Creating columns requires an understanding of the structure of the individual sheet, see **Chapter 2, “Structure of Excel Sheets”**.



MANUAL ENTRY

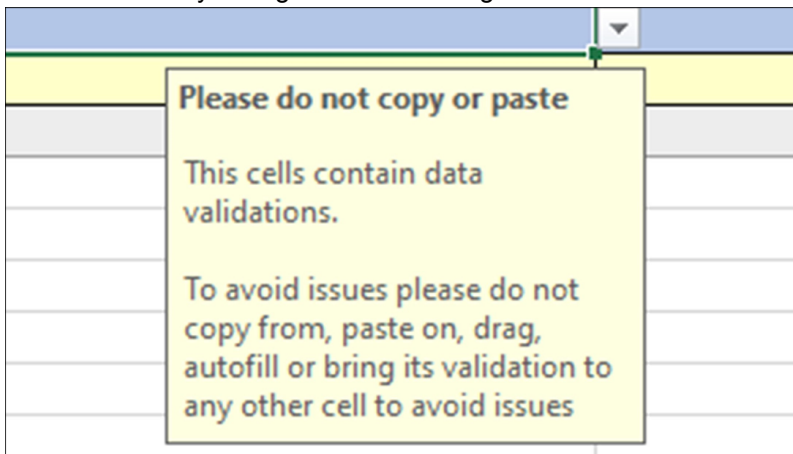
1. Open Excel Template.
2. The Excel Template will have some of the columns started for reference points.
3. Unlock the Excel Template by selecting Review (1), and then Unprotect Sheet (2).

NOTE: The sheets do not require a password and are locked to prevent accidentally deleting formulas within the individual cells.



- To prevent Data validation problems, do not copy, cut, paste, or drag cells with drop-down data validations into any other cell.

There are input messages on cells that contain drop-downs to alert the user not to accidentally change cells containing data validation.



Structure of Excel Sheets

The first (3) rows define the type of data that will be input into a column. Row 4 is used for user instructions and is not used by the program. Row 5 and beyond is used for data input.

attribute	quantity	identifiers
*name	volume	cas
	ml	
CAS number		
toluene	10	108-88-3
styrene	10	100-42-5

ROW 1: CATEGORY

Row 1 (1) defines the characteristic of row 2 (2).

	A	B	C
1	attribute	quantity	identifiers
2	*name	volume	cas
3		ml	
4	CAS number		
5	toluene	10	108-88-3
6	styrene	10	100-42-5

All possible options are:

- **attribute**
 - Defines the nomenclature of the column.
- **condition**
 - The condition under which the property was found.
 - Some examples include: *temperature*, *mixing_rate*, *stirring*, and *time_duration*.
 - A full list of conditions can be found at the CRIPT Docs hub. Navigate to CRIPT Home > Help > Controlled vocabulary > Condition key, or use the following hyperlink "[Condition controlled vocabulary](#)".
- **identifier**
 - Provides the naming method for material in the column.

- Some examples include: *smiles*, *bigSmiles*, *cas*, and *name*.
- A full list of material identifiers can found at the CRIPT Docs hub. Navigate to CRIPT Home > Help > Controlled vocabulary > Material identifier key, or use the following hyperlink "[Material identifier controlled vocabulary](#)".
- **property**
 - Describes the characteristic or unit combinations for the property in the column.
 - Some examples include: *phase*, *color*, *density*, and *molar_mass*.
- **relation**
 - Allows column row to reference another row in a different sheet.
 - An example would be to relate attribute (3) *name (4) to relation (5) *mixture (6) on another sheet. For more information on sheets, "See "Individual Excel Sheets" on page 2-17".

1	attribute	attribute	identifiers	identifiers	identifiers	identifiers
2	*name	note	cas	names	smiles	bigsmiles
3						
4			CAS number	additional names	simplified molecular-input line-entry system (SMILE)	bigSMILES line notation
5	toluene		108-88-3	methylbenzene; toluene	Cc1ccccc1	
6	styrene		100-42-5	vinylbenzene; phenylethylene; ethenylbenzene; styre	C=Cc1ccccc1	
7	1-butanol		71-36-3	n-butanol; n-butyl alcohol; 1-butyl alcohol; nBuOH; 1-bi	CCCCO	
8	methanol		67-56-1	Methyl alcohol; CH3OH; MeOH; methanol	CO	
9	polystyrene			poly(styrene); poly(vinylbenzene); polystyrene		[H]([>[<]C(C[>])c1ccccc1[<])C(C
10	mixture					
11	component1					
12	component2					
13	intermediate_material_1					
14						
15						

1	relation	relation
2	*mixture	*material
3		
4	Pick a value from *name column of material sheet that repersents the mixture	Pick a value from *name column of material sheet
5	mixture	component1
6	mixture	component2
7		
8		
9		
10		
11		
12		

- **quantity**
 - Column specifies the amount or combinations of quantities.
 - The numerical value can be combined with a unit of measure.

ROW 2: COLUMN NAME

Defines the name of the Column and can be edited by the user.

- Columns beginning with " * " are required (eg. *name).
- Columns beginning with " # " will be ignored (eg. #storage).
- The # columns are a good idea to use if you want to have some notes but don't want them necessarily read or uploaded to CRIPT.
- Some sheets have drop-downs' for row 2. After selecting an option, row 1, row 3, and row 4 are automatically populated with the correct information.

NOTE: Some sheets have drop-downs for row 2. After selecting an option, row 1, row 3, and row 4 are automatically populated with the correct information.

NOTE: As the controlled vocabulary is updated, it is very possible that the drop-down options and autofill can become outdated.

- Each Excel sheet has rows 1, 3, and 4 locked for protection.
- This is because rows 1, 3, and 4 contain formulas that if accidentally deleted or overwritten would break the autofill feature.
- The sheets are protected without a password and can be easily unprotected.

Nesting Headers in Row 2

Nesting can be defined as having a combination of names. Combined names are separated by a colon “:”.

- Example: density at a certain temperature.
- density:temperature
- Example: uncertainty of a material property condition.
- density:temperature:uncertainty

NOTE: Nesting “data” column values should derive from the “*name” column of the data sheet. For more information, see “Individual Excel Sheets”.

NOTE: Nesting “citation”, column values should derive from the “*name” column of the Citation sheet. For more information, See “Individual Excel Sheets” on page 2-17.

ID

The Id feature is used to allow for multiple measurements through time. With Id it is possible to take several measurements through an experiment, and later use nesting to record more details.

If there are multiple densities throughout time and we want to show each of their temperatures (or any other condition), we can use an Id field to differentiate between the different temperatures. We denote an Id with brackets and a number inside such as [1] or [2]. The Id is used to identify distinct properties/conditions of the same type.

1. To identify two density measurements at two different temperatures, we could create the following column headers:

[1]density (1), [1]density:temperature (2), [2]density (3), [2]density:temperature (4).

D	E	F	G
①	②	③	④
[1]density	[1]density:temperature	[2]density	[2]density:temperature
0.87	20	1	30

ROW 3: UNITS

Defines the units used in the column.

- Some examples include: *Celsius*, *g/ml*.
- A full list of supported units can be found at: https://github.com/hgrecco/pint/blob/master/pint/default_en.txt, or use the following hyperlink “supported units”.

ROW 4: INSTRUCTIONS

Many columns contain instructions on row 4. These clarify the data that should go in that column after the value for row 2 is picked.

ROW 5 AND BEYOND: USER INPUT

These rows are used to enter user data.

- Some columns allow for more than a single value. List values must use a semicolon “;” as a separator.
- Example of inputting multiple funders.

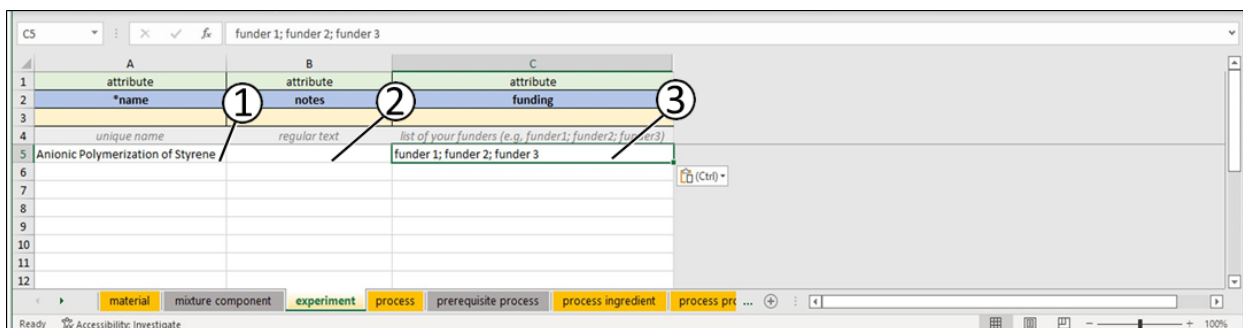
attribute
funding
<i>list of your funders (e.g, funder1; funder2; funder3)</i>
funder 1; funder 2; funder 3; funder4

Individual Excel Sheets

The Excel Template is comprised of 10 individual sheets in the sheet line (1) for specific characteristics. These are separated into two colored categories. Orange categories (2) are required and Grey categories (3) optional.

NOTE: Sheets can not be renamed, but can be removed. The optional column’s not needed can be left blank or deleted.

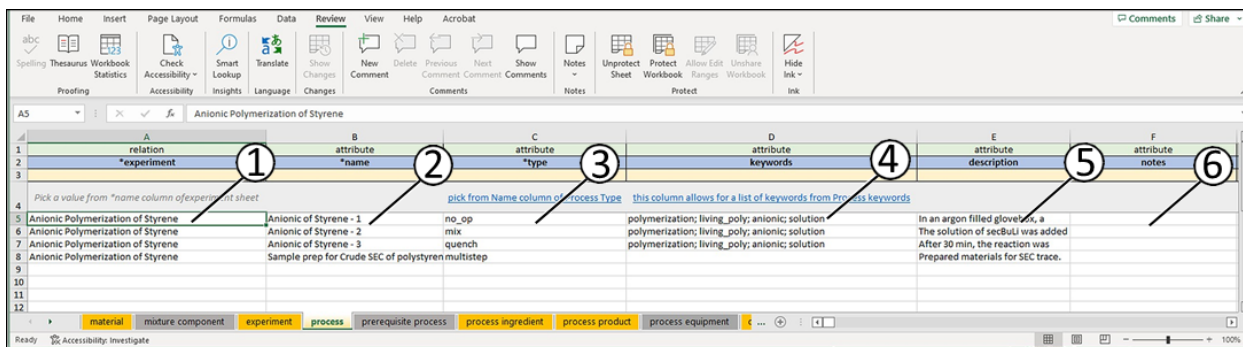
material	mixture component	experiment	process	prerequisite process	process ingredient	process product	process equipment	data	citation
①			②	③					



PROCESS SHEET

Define the processes of each experiment.

1. Experiment value (1) is derived from the “ *name” column of the experiment sheet.
2. Name (2) is a unique name for the process.
3. Type (3) is selected from the name column of Process Type file found at [CRIPT Home > Help > Controlled vocabulary > Process type](#), or use the following hyperlink “[Process Type](#)”.
4. Keywords (4) is selected from the name column of Process keywords found at [CRIPT Home > Help > Controlled vocabulary > Process keyword](#), or use the following hyperlink “[Process keywords](#)”.
5. Description (5) describes the process.
6. Notes (6) allows user to enter notes regarding process.



1. Equipment (7) is selected from the list of equipment found at [CRIPT Home > Help > Controlled vocabulary > Equipment key](#), or use the following hyperlink “[equipments](#)”.
2. Columns H and beyond are customizable. Some examples would include.
 - Process property temperature (8) is selected from the list of Process property found at [CRIPT Home > Help > Controlled vocabulary > Process property key](#), or use the following hyperlink “[process property](#)”.
 - Condition time_duration (9), is selected from the list of Condition Keys at [CRIPT Home > Help > Controlled vocabulary > Process property key](#), or use the following hyperlink “[process property](#)”.

1	attribute	property	condition	property
2	equipment	temperature	time_duration	enthalpy_activation
3		degC	min	J/mol
4	you can have a list of equipments ⑦	Temperature ⑧	rent occurred (also see time_dura ⑨	an'halpy of activation, molar ba
5		25		30
6		25		30
7		25		30
8				
9				
10				

PREREQUISITE PROCESS SHEET

Define the immediate prerequisites for each process.

1. Process (1) is derived from the “ *name” column of the process sheet.
2. Prerequisite (2) is derived from the “ *name” column of the process sheet.

NOTE: Each prerequisite must be from the previous process.

	A	B
1	relation	relation
2	*process	*prerequisite
3		
4	<i>Pick value from *name column of process sheet</i>	<i>Pick value from *name column of process sheet</i>
5	Anionic of Styrene - 2	Anionic of Styrene - 1
6	Anionic of Styrene - 3	Anionic of Styrene - 2
7		
8		

PROCESS INGREDIENT SHEET

Define the ingredients for each process and their respective quantities.

1. Process (1) is derived from the “ *name” column of the process sheet.
2. Material (2) is derived from the “ *name” column of the material sheet.
3. Keyword (3) is selected from the list of ingredients found at CRIPT Home > Help > Controlled vocabulary > Ingredient keyword, or use the following hyperlink [“Ingredient keyword”](#).
4. Columns D and beyond are customizable. Some examples would include.
 - Quantity mole (4) is selected from the list of quantity values found at CRIPT Home > Help > Controlled vocabulary> Quantity key, or use the following hyperlink [“Quantity key”](#).

A	B	C	D	E	F	G
relation	relation	attribute	quantity	quantity	quantity	
*process	*material	*keyword	mole	mass	volume	
			mmole	kg	ml	
Pick value from *name column of process sheet	Pick value from *name column of material sheet	Pick value from Name column of ingredients				
Anionic of Styrene - 1	toluene	solvent		0.455	10	
Anionic of Styrene - 2	styrene	monomer				
Anionic of Styrene - 3	methanol	workup				
Anionic of Styrene - 3	intermediate_material_1	workup				

PROCESS PRODUCT SHEET

This sheet describes the resulting product after completing a process.

1. Process (1) is derived from the “ *name” column of the process sheet.
2. Material (2) is derived from the “ *name” column of the material sheet.

A	B
relation	relation
*process	*material
Pick value from *name column of process sheet	Pick value from *name column of material sheet
Anionic of Styrene - 2	intermediate_material_1
Anionic of Styrene - 3	polystyrene

PROCESS EQUIPMENT SHEET

Define the equipment used in a process.

1. Process (1) is derived from the “ *name” column of the process sheet.
2. Key (2) is selected from the list of equipment found at [CRIPT Home > Help > Controlled vocabulary > Equipment key](#), or use the following hyperlink “Equipment key”.
3. Description (3) describes the type of equipment.
4. Citation (4) is derived from the “ *name” column of the citation sheet.
5. Condition temperature (5) is selected from the list of conditions found at [CRIPT Home > Help > Controlled vocabulary > Condition key](#), or use the following hyperlink “Condition key”.

A	B	C	D	E
relation	attribute	attribute	relation	Condition
*process	*key	description	citation	temperature
				degC
Pick value from *name column of process sheet	Pick value from Name column of equipments		Pick value from *name column of citation sheet	Temperature
Anionic of Styrene - 1	glass_rbf	Glass round bottom flask	Ref1	

DATA SHEET

This sheet defines the data files you want to upload to CRIPT, such as a CSV file from a robot, an image, or any other type of file.

1. Experiment (1) is derived from the “*name” column of the experiment sheet.
2. Name (2) is a unique name for the experiment.
3. Type (3) is selected from the list of data types found at CRIPT Home > Help > Controlled vocabulary > Data type, or use the following hyperlink “Data type”.
4. Source (4) can either be a path to a local file on your computer or a url to a website.
5. Sample_preparation (5) is derived from the “*name” column of the process sheet.

	A	B	C	D	E
1	relation	attribute	attribute	attribute	relation
2	*experiment	*name	*type	*source	sample_preparation
3		Crude SEC of polystyrene	sec_trace	https://google.com	sample prep for Crude SEC of polystyrene
4		NMR of polystyrene	nmr	C:\Users\myusername\example_template_v0-3-1.xlsx	
5					
6					
7					
8					
9					
10					
11					
12					

CITATION SHEET

This sheet can be used to reference any sources used in the experiments that you want to cite in CRIPT.

1. Required columns are the title (1) and Digital object identifier (doi) (2). Optional information can be added in the additional columns.

	A	B	C	D	E	F	G	H	I	J
1	attribute	attribute	attribute	attribute	attribute	attribute	attribute	attribute	attribute	attribute
2	*title	*doi	authors	journal	publisher	year	volume	issue	pages	issn
3										
4										
5	Ref1	doi/1	Brad;	journal	o'reilly media	2022	11	943	test	test
6										
7										
8										
9										
10										
11										
12										

Chapter 3

Python SDK

Quickstart

Prerequisites

- Python version 3.9 or higher
- Internet access
- API Token. For more information, See “API Token” on page 2-3

Installation

1. Download and install Python version 3.9 or higher. Refer to [this link](#) for details on downloading python
2. Open the command prompt by selecting the Windows icon and type “CMD”
3. Enter `pip install cript` to the command line (1) and press enter

Connect to CRIPT

1. Enter “python” at cursor (1) and press enter.
2. Open any text editor (Microsoft Notepad shown here).
3. Copy and paste the following example code:

```
host = "<endpoint_hostname>" # e.g., criptapp.org
token = "<your_api_token>"
cript.API(host, token)
```

4. Modify the host line (1) with the application programming interface (API) destination.
5. Modify the token line (2) with the user token. See “API Token” on page 2-3.



```
File Edit Format View Help
import cript
host = "criptapp.org"
token = "Token 5d357051969ef9f4390dc1896aff0cff44852bcc"
cript.API(host, token)
```

6. Copy and paste modified host line from Notepad to the command prompt cursor (3) and press enter.

7. Copy and paste modified token line from Notepad to the command prompt cursor (4) and press enter.
 8. Copy and paste cript.API line from Notepad to the command prompt cursor (5) and press enter.
 9. Confirm connected to API site (6).
- NOTE: Each line must be entered individually.

```

Command Prompt - python
Requirement already satisfied: globus-sdk>=3.7.0 in c:\users\sheld\appdata\local\programs\python\python310\lib\site-pack
ages (from cript) (3.14.0)
Requirement already satisfied: requests>=2.27.1 in c:\users\sheld\appdata\local\programs\python\python310\lib\site-packa
ges (from cript) (2.28.1)
Requirement already satisfied: cryptography!=3.4.0,>=3.3.1 in c:\users\sheld\appdata\local\programs\python\python310\lib
\site-packages (from globus-sdk>=3.7.0->cript) (38.0.3)
Requirement already satisfied: pyjwt[crypto]<3.0.0,>=2.0.0 in c:\users\sheld\appdata\local\programs\python\python310\lib
\site-packages (from globus-sdk>=3.7.0->cript) (2.6.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\sheld\appdata\local\programs\python\python310\lib\site-
packages (from requests>=2.27.1->cript) (1.26.12)
Requirement already satisfied: idna<4,>=2.5 in c:\users\sheld\appdata\local\programs\python\python310\lib\site-packages
 (from requests>=2.27.1->cript) (3.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\sheld\appdata\local\programs\python\python310\lib\site-pac
kages (from requests>=2.27.1->cript) (2022.9.24)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\sheld\appdata\local\programs\python\python310\lib\si
te-packages (from requests>=2.27.1->cript) (2.1.1)
Requirement already satisfied: cffi>=1.12 in c:\users\sheld\appdata\local\programs\python\python310\lib\site-packages (f
rom cryptography!=3.4.0,>=3.3.1->globus-sdk>=3.7.0->cript) (1.15.1)
Requirement already satisfied: pycparser in c:\users\sheld\appdata\local\programs\python\python310\lib\site-packages (fr
om cffi>=1.12->cryptography!=3.4.0,>=3.3.1->globus-sdk>=3.7.0->cript) (2.21)

C:\WINDOWS\System32>python
Python 3.10.8 (tags/v3.10.8:aaaf517, Oct 11 2022, 16:50:30) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import cript
>>> host = "criptapp.org"
>>> token = "Token 5d357051970ef9f4390db1896aff0cff44852bcc"
>>> cript.API(host, token)
Connected to https://criptapp.org/api
>>>

```

Sample Tasks

The sample tasks are excellent templates to be used to record data. Replace sample tags in quotes with the name that applies to the project.

Create a node:

Create a Project and Collection node.

```
proj = cript.Project(name="MyProject")
proj.save()
```

```
coll = cript.Collection.create(project=proj, name="MyCollection")
```

Create an Experiment node.

```
expt = cript.Experiment(
    collection=coll,
    name="Anionic Polymerization of Styrene with SecBuLi"
)
expt.save()
```

NOTE: The use of create() here, instantiates and saves the object at the same time.

Update a node:

Update the Project and Collection node created above.

```
proj.name = "OurProject"  
proj.save()
```

```
coll.update(name="OurCollection")
```

NOTE: The use of `update()` here, updates and saves a node at the same time.

Delete a node:

Delete the Collection node created above.

```
coll.delete()
```

Get an existing node:

Get the official CRIPT Project node.

```
proj = cript.Project.get(name="CRIP")
```

Get the official styrene Material node via name.

```
styrene = cript.Material.get(project=proj.uid, name="Styrene")
```

Get the official styrene Material node via UID.

```
styrene = cript.Material.get(uid="<material_uid>")
```

Get the official styrene Material node via URL.

```
styrene = cript.Material.get(url="<material_url>")
```

Get Material nodes:

Get an existing Inventory node from the database.

```
uid = "134f2658-6245-42d8-a47e-6424aa3472b4"  
inv = cript.Inventory.get(uid=uid, get_level=1)
```

NOTE: `Get_level` is set to 1 so that the Material nodes are auto-generated. This parameter defaults to 0, but can be set to any integer.

Notice that the Material node objects have been auto-generated.

```
type(inv.materials[0])  
# <class 'cript.data_model.nodes.material.Material'>
```

Run a search query:

Search for Material nodes with a molar mass of less than 10 g/mol.

```
res = cript.Material.search(  
  properties = [  
    {  
      "key": "molar_mass",  
      "value_lt": 10, "unit":  
        "g/mol"  
    }  
  ]  
)
```

Page through the results.

```
res.json()      # View the raw JSON for the query  
res.objects()   # Generate objects for the current page  
res.next_page() # Flip to the next page  
res.previous_page() # Flip to the previous page
```

Upload a file:

User will need a Project and Data node to upload a file.

```
proj = cript.Project.get(uid="<project_uid>")  
data = cript.Data.get(uid="<data_uid>")
```

Create a File node that points to a local file.

```
path = "path/to/local/file"  
f = cript.File(project=proj, source=path)  
file.save()
```

Download a file:

Download the file that was uploaded above.

```
path = "path/to/local/file"  
f.download_file(path=path)
```

NOTE: The default path for a download is in the current directory.

Create a Process node:

```
prcs = cript.Process(  
    experiment=expt, name="Anionic  
of Styrene", type = "multistep",  
    description = "In an argon filled glovebox, a round bottom flask was filled with 216 ml of dried toluene. The "solution  
of secBuLi (3 ml, 3.9 mmol) was added next, followed by styrene (22.3 g, 176 mmol) to "initiate the  
polymerization. The reaction mixture immediately turned orange. After 30 min, "  
  
    "the reaction was quenched with the addition of 3 ml of methanol. The polymer was isolated by " "precipitation  
in methanol 3 times and dried under vacuum."  
  
)  
  
prcs.save()
```

Add Ingredient nodes to the Process node:

Get the Material nodes from the Inventory node.

```
solution = inv['SecBuLi solution 1.4M cHex'] toluene =  
inv['toluene']  
styrene = inv['styrene'] butanol =  
inv['1-butanol'] methanol =  
inv['methanol']
```

Define Quantity nodes indicating the amount of each Ingredient.

```
initiator_qty = cript.Quantity(key="volume", value=0.017, unit="ml")  
solvent_qty = cript.Quantity(key="volume", value=10, unit="ml")  
monomer_qty = cript.Quantity(key="mass", value=0.455, unit="g")  
quench_qty = cript.Quantity(key="volume", value=5, unit="ml") workup_qty  
= cript.Quantity(key="volume", value=100, unit="ml")
```

Create Ingredient nodes for each:


```

initiator = cript.Ingredient(
    keyword="initiator",
    material=solution,
    quantities=[initiator_qty]
)

solvent = cript.Ingredient(
    keyword="solvent",
    material=toluene,
    quantities=[solvent_qty]
)

monomer = cript.Ingredient(
    keyword="monomer",
    material=styrene,
    quantities=[monomer_qty]
)

quench = cript.Ingredient(
    keyword="quench",
    material=butanol,
    quantities=[quench_qty]
)

workup = cript.Ingredient(
    keyword="workup",
    material=methanol,
    quantities=[workup_qty]
)

```

Add Ingredient nodes to the Process node:

```

prcs.add_ingredient(initiator)
prcs.add_ingredient(solvent)
prcs.add_ingredient(monomer)
prcs.add_ingredient(quench)
prcs.add_ingredient(workup)

```

Add Condition nodes to the Process node:

```

temp = cript.Condition(key="temperature", value=25, unit="celsius")
time = cript.Condition(key="time_duration", value=60, unit="min")
prcs.add_condition(temp)
prcs.add_condition(time)

```

Add Property node to the Process node:

```

yield_mass = cript.Property(
    key="yield_mass",
    value=0.47,
    unit="g",
    method="scale"
)

prcs.add_property(yield_mass)

```

Create a Material node (process product):

Instantiate the node.

```

polystyrene = cript.Material(project=proj, name="polystyrene")

```

Add Identifier nodes.

```

names = cript.Identifier(
    key="names",
    value=["poly(styrene)",
          "poly(vinylbenzene)"])
bigsmiles = cript.Identifier(
    key="bigsmiles",

    value="[H]([>][<]C(C[>])c1ccccc1[<])C(C)CC")

chem_repeat = cript.Identifier(key="chem_repeat", value="C8H8")
polystyrene.add_identifier(names)
polystyrene.add_identifier(chem_repeat)
polystyrene.add_identifier(bigsmiles)

```

Add Property nodes.

```

phase = cript.Property(key="phase", value="solid")
color = cript.Property(key="color", value="white")
polystyrene.add_property(phase)
polystyrene.add_property(color)

```

Save the Material and add it to the Process node as a product.

```

polystyrene.save()
prcs.add_product(polystyrene)

```

Save the Process node.

```

prcs.save()

```

Create a File node and upload a file:

Instantiate a File node and associate it with the Data node created above.

```
path = "path/to/local/file"  
f = cript.File(project=proj, source=path)
```

NOTE: The source field should point to a file on the local filesystem.

NOTE: Depending on the file size, there could be a delay while the checksum is generated.

Upload the local file by saving the File node. Follow all prompts that appear.

```
api.save(f)
```

Create a Data node:

```
sec = cript.Data( experiment=expt,  
                 name="Crude SEC of polystyrene",  
                 type="sec_trace",  
                 )
```

Add the uploaded File to the Data node.

```
sec.add_file(f)  
sec.save()
```

Associate a Data node with a Property node:

Create one more Property node for polystyrene.

```
mw_n = cript.Property(key="mw_n", value=5200, unit="g/mol")
```

Add the Data node to the new Property node.

```
mw_n.data = sec
```

Add the new Property node to polystyrene, then save it.

```
polystyrene.add_property(mw_n)  
polystyrene.save()
```

CRIPT Blog

The CRIPT blog is a space that aims to provide a channel for the CRIPT community to develop educational resources, share case studies and highlight the ways in which data and informatics can lead to innovation in polymer technology.

The CRIPT blog includes:

- Academic content written in an accessible format
- Demonstrations and explanations of the technology behind CRIPT
- Case studies from the CRIPT community

The CRIPT blog was created to be a community-driven resource, and participation from the CRIPT community is welcome. Contributions to the blog can be made through editorial input or peer reviewing, identifying topics where blog content would be useful, and authoring or co-authoring a post.

To get involved with the blog, contact the CRIPT team at cript@mit.edu.

Chapter 4

Search function

Table of synonyms