
Ontology Development 1

SOP: Extracting Term Annotations from OWL File

PURPOSE

To describe the procedure for extracting annotations of all terms in an ontology's OWL file, using the python program "*create_terms_for_review_file.py*", into a format that can be read and shared in Excel format.

SCOPE

This procedure is used by curators when producing term annotations in a format that can be reviewed by reviewers, or for other processes involved in ontology development, such as ontology translation.

GLOSSARY/DEFINITIONS

Python	The programming language in which the program used in this SOP was written.
Protégé	An ontology development software program.
SOP	Standard Operating Procedure: Detailed written instructions to achieve uniformity of the performance of a specific function.

SYSTEM REQUIREMENTS TO RUN PROGRAM

Instructions below are for Windows 10.

For the python program to run successfully, the following need to be installed:

1. Python.
Ensure that Python is setup correctly on your PC (Use ***Program Running 1 SOP***)
2. Desktop Protégé.
If you do not already have desktop Protege installed, download and install it from: <https://protege.stanford.edu/products.php>

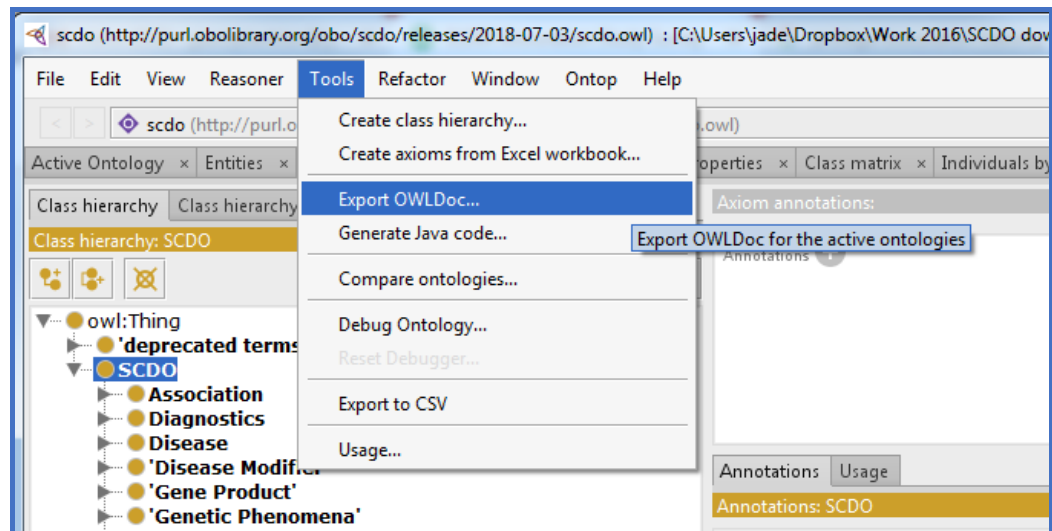
REQUIRED INPUT FOR PROGRAM

Before the program can be run, these required input files and folders need to be prepared and named in the program code:

1. Html files generated by Protégé
 - The locations of the "index.html" file in the "classes" folder and the location of this "classes" folder.
2. A file containing the list of terms to be extracted.
3. A file of important annotation properties, in the order of preference (used to output annotations in a certain order).

PREPARING INPUT FOR PROGRAM

1. To generate html files of an ontology:
 - a) Open the owl file in Protégé Desktop.
 - b) Export the entire ontology from Protégé as html by using the “Export OWL Doc” function from the Tools tab in Protégé [Note: export to a folder in the same location as the python program file.]



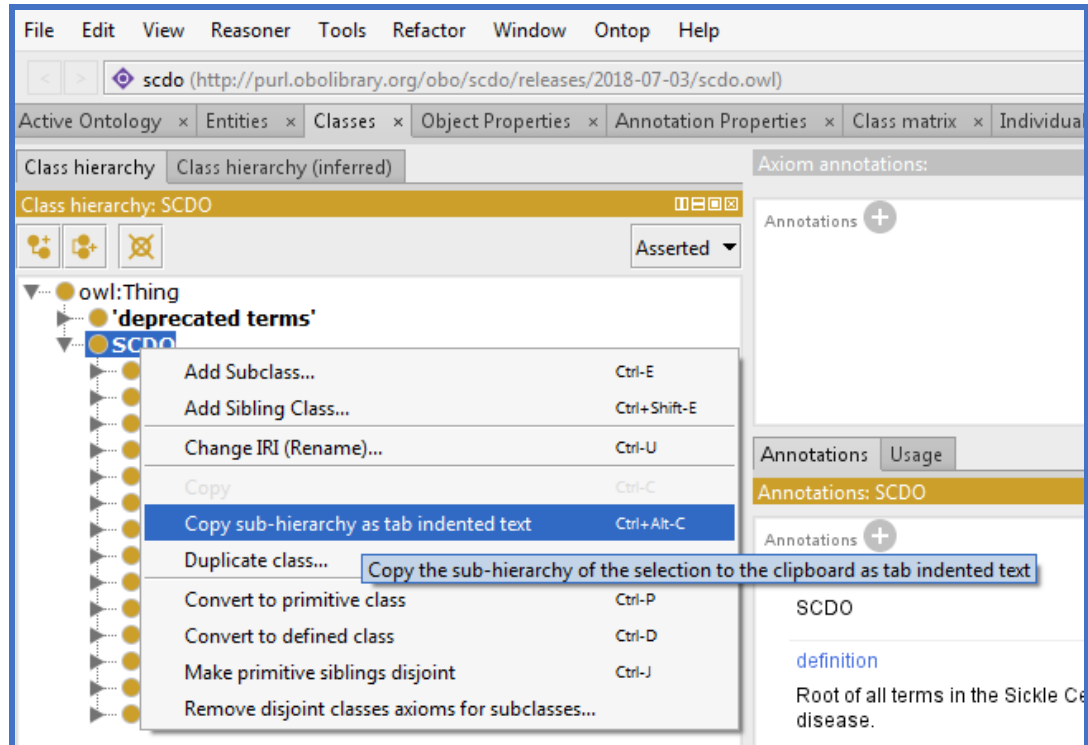
- c) Locate the “classes” folder that was exported and make sure the “classes_location” in the python program code (row 286) points to the location of the “classes” folder that contains the htmls of classes.

[Note: The location can be relative to the python program’s location, e.g. if the folder containing exported htmls from Protégé is called “Exported_from_Protege” and the “classes” folder is within it, then the location provided in row 286 should be “Exported_from_Protege\classes”.]

- d) Locate the “index.html” file (within the “classes” folder”) and make sure “input_file2” in the python program code (row 282) points to the location of the “index.html” file. [In keeping with the example already used in the previous step, the location would be “Exported_from_Protege\classes\index.html”.]

2. To prepare the file with the list of terms to be extracted:
 - a) Open the OWL file in Desktop Protégé

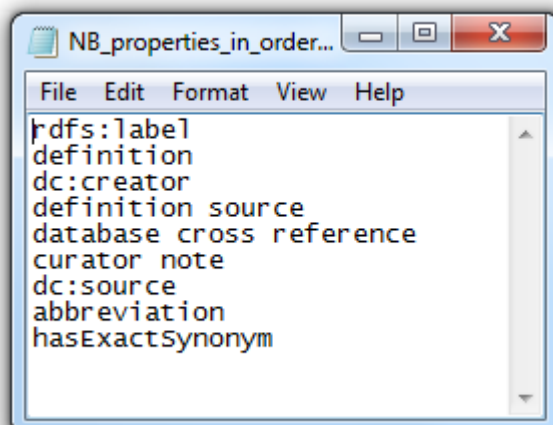
- b) Copy terms from classes to be reviewed (e.g. for all terms within the main "SCDO" term in the SCDO, right click on "SCDO" and select "Copy sub-hierarchy as tab indented text") and paste in a .txt file.



- c) Delete the empty row that automatically appears at the bottom of the copied terms.
d) Save the file in the same location as the python program file.
e) Make sure "input_file1" in the python program code (row 281) points to the name of this text file.
[The template file, "input_file_complete_ontology_template.txt" shows an example of what the content of the file could be.]

3. To create the input file that orders annotations as desired:

- a) Create a .txt file containing a custom list of important properties that are listed, one per row, according to the order in which they should appear in the header row of the exported terms. (This file can be re-used or edited as necessary for future runs of the program).



- b) Save the file in the same location as the python program file.
 - c) Make sure "input_file3" in the python program code (row 283) points to the name of this text file.
4. Use the "outputfile_name" in the python program code (row 287) to specify the name of the output file that will be generated by the program (Do not include a file extension; it will automatically be a .txt file).

RUNNING THE PROGRAM

Run the *create_terms_for_review_file.py* python program by following the procedure in the **Program Running 2 SOP**.

USING THE OUTPUT FILE

The output .txt file will appear in the same location as the python program file and can now be:

1. opened in Excel and saved in that format.
2. uploaded as a Google Spreadsheet which can be shared.
3. used in the process when the ontology is translated (In **Ontology Translation 1 SOP** and **Ontology Translation 5 SOPs**).

Version No.	Date	Authorizer	External Reviewer	Internal Reviewer	Author	Details of changes
					Jade Hotchkiss	n/a (first version)