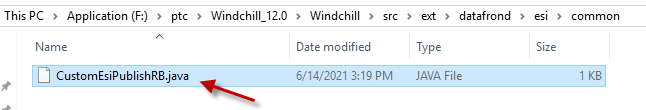
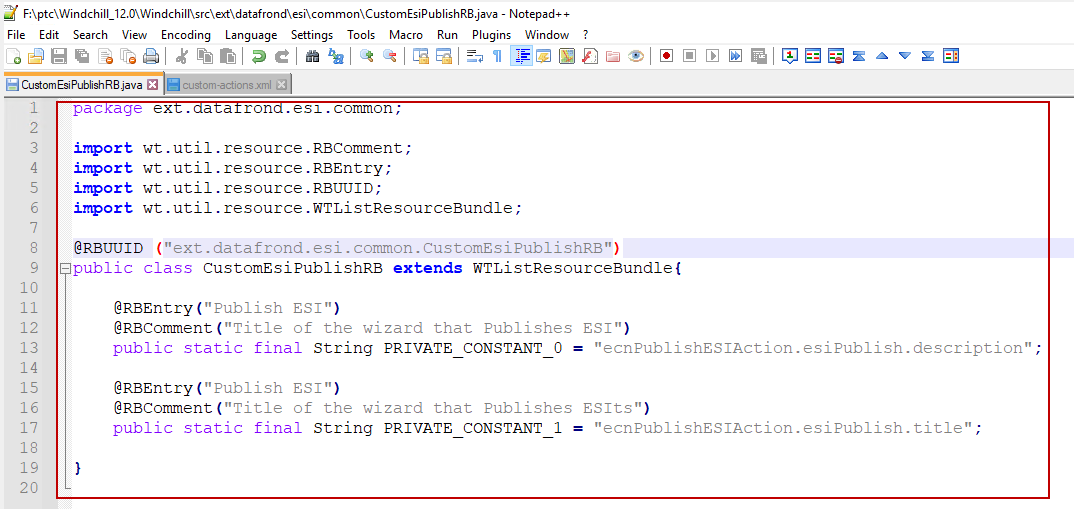
# Table of Contents

## How to create Actions on Windchill Objects

1. Follow the below procedure to create custom actions on Windchill Objects.
2. Create a java file with below info in your project related java directory (In my case project directory is <WC\_Home>/src/ext/datafrond/esi/common).

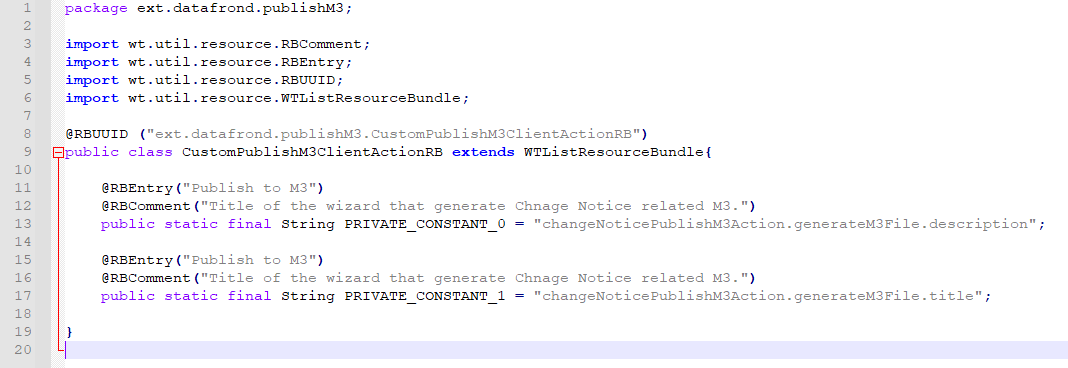




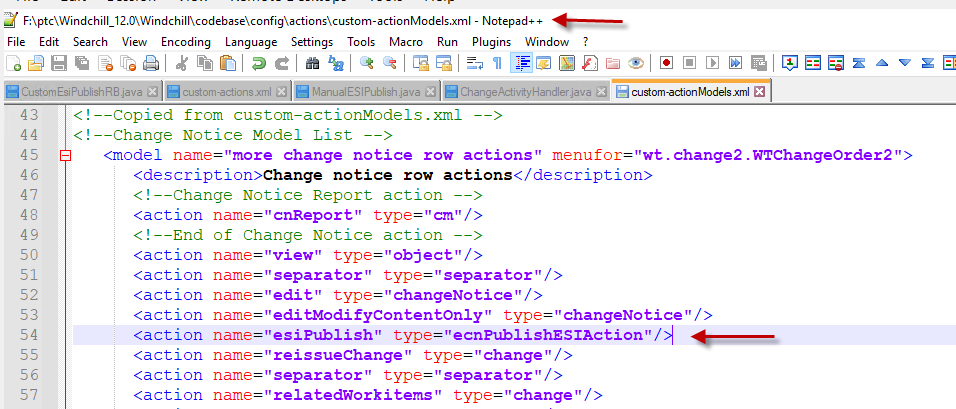
1. Below parameters describes the above java file entries.

* We need to import RBComment, RBEntry, RBUUID and WTListResourceBundle libraries definitely without missing.
* **Line-8: @RBUUID:** We need to pass the java file directory along with class name as argument to **@RBUUID**
* Corresponding class should extends **WTListResourceBundle** class.
* **Line-11: @RBEntry:** We need to pass the action name that we want in Windchill UI.
* **Line-12: @RBComment:** We need to pass some description as argument to @RBComment.
* **Line-13:** Need to create final String variable for describing description.
* **Line-15: @RBEntry:** We need to pass the action name that we want in Windchill UI.
* **Line-16:** **@RBComment:** We need to pass some title as argument to @RBComment.
* **Line-17:** Need to create final String variable for describing title.

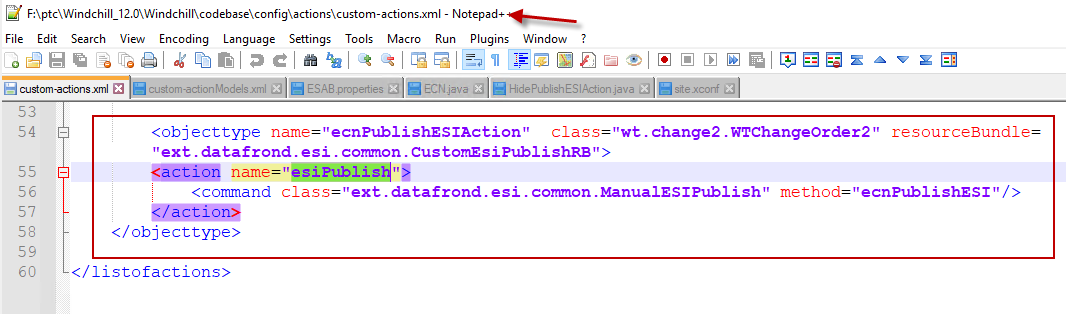
1. For more details refer below screenshot for another action.



1. Compile the corresponding java file and ensure class file got generated with latest time stamp.
2. Add the entry in custom-actionmodels.xml as shown below at corresponding class for which we need the action. In this case it is Change Notice (wt.change2.WTChangeOrder2). Provide any name and type that should be unique to the action.



1. Add the entry in custom-actions.xml as shown below.



Where **Line-54: name** should be the type name provided in custom-actionmodels file.

**class**: provide the class name of the WTObject on which this action needs to be visible.

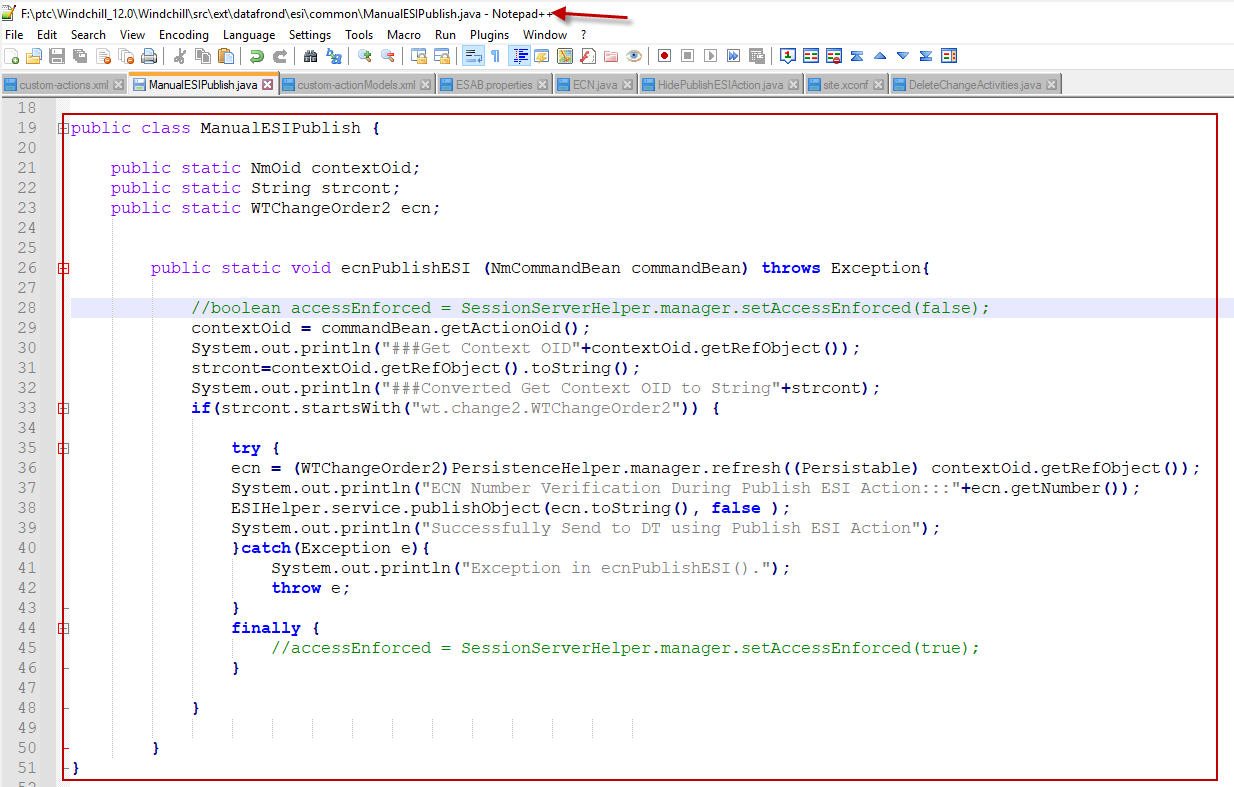
**resourceBundle**: provide the and directory and RBJava file name created for action.

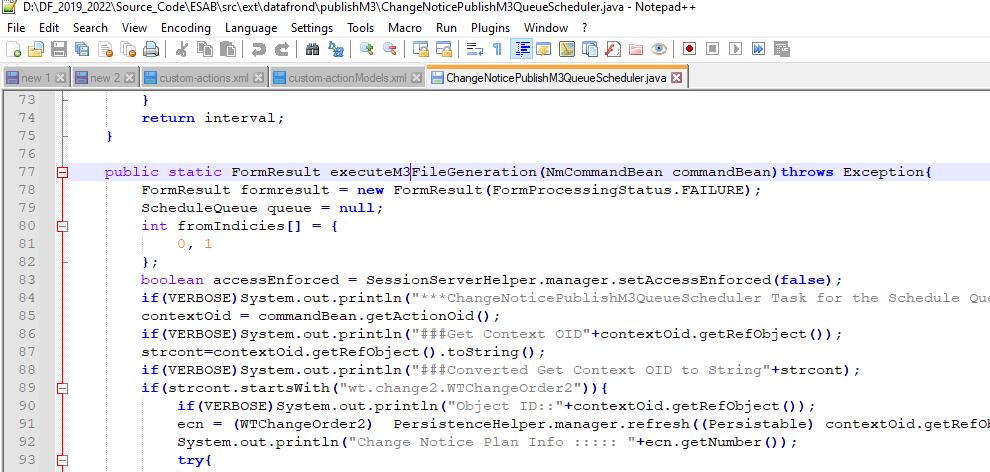
**action**:In between this action tag “**name”** should containthe action name provided in custom-actionmodels.

**command**: In between this command tag “**class**” should contain the class name where our required method that needs to be executed when we click on the action. "**method**” provide the method name that needs to be triggered when we click on the action.

**Note**: For every method which needs to be triggered when we click on the action we need to pass NMCommandBean as argument.

**For Example**:





## Operations for visibility and greying out custom action based on different conditions

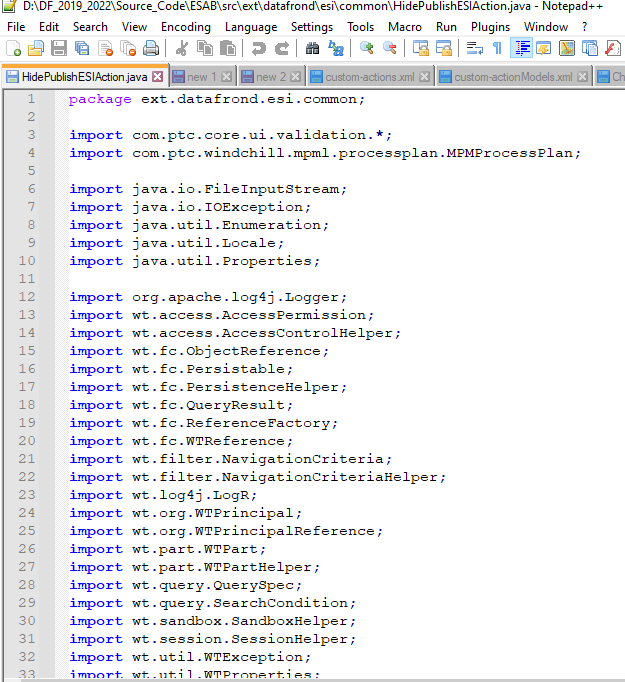
## Example-1:

For example if our custom actions needs to visible for particular cases on UI and it should be greyed out for particular cases on UI.

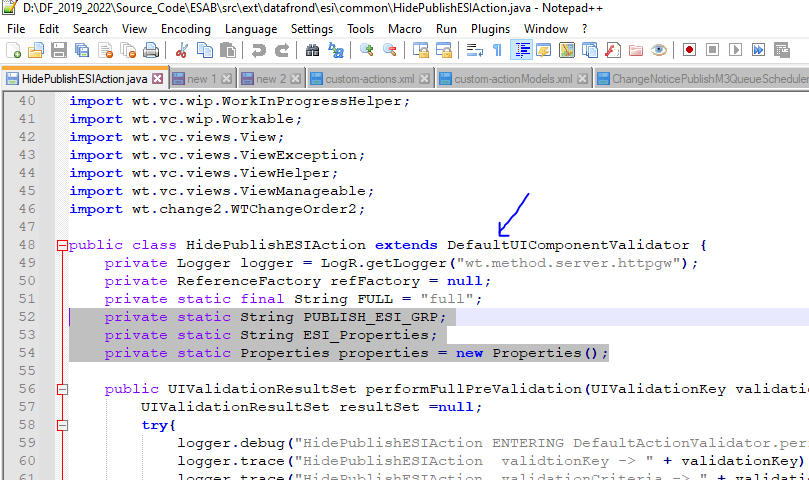
1. To achieve this functionality we need to develop a code.
2. To hide the action if the user is not available in particular group and ECN is not in Completed state developed the below code (Note: We can reuse the code to perform similar action visibility operations).
3. Please find the attached file for reference.



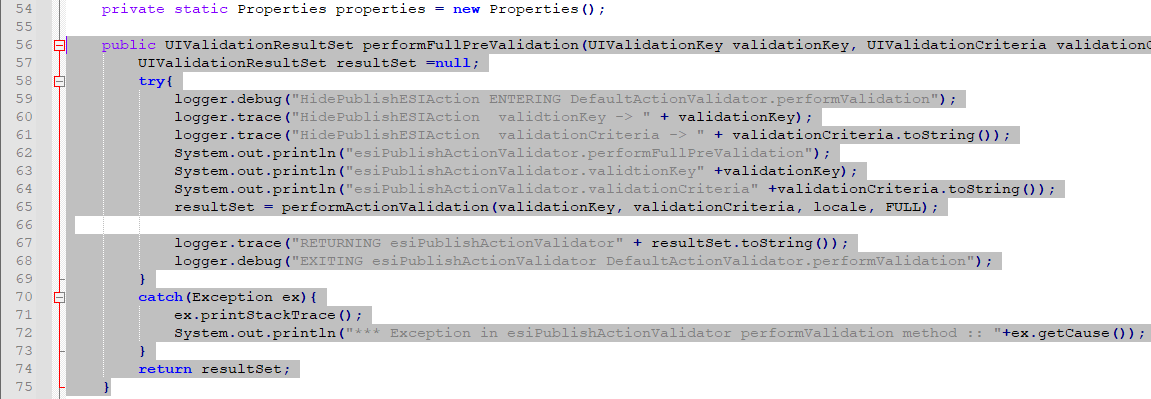
1. Import required packages.



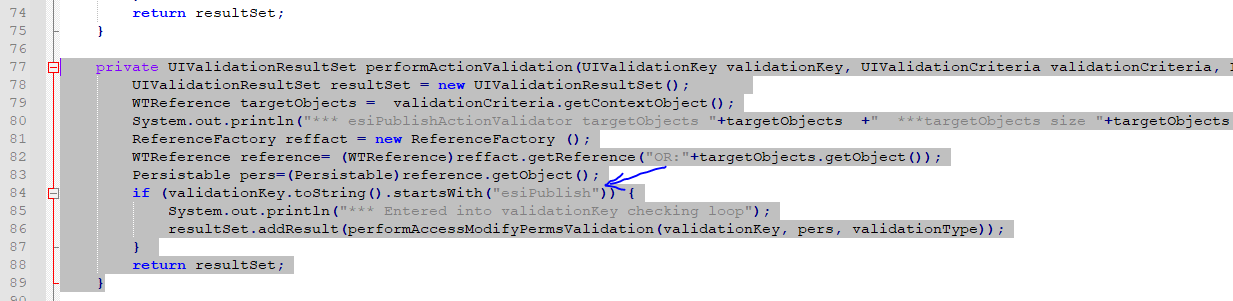
1. Our main class should extends DefaultUIComponentValidator class as a mandatory and declare required variables as private. In the below screenshot highlighted variables are custom one and remaining are OOTB once.



1. Below method is the mandatory OOTB method and we need to update the println statements as required.



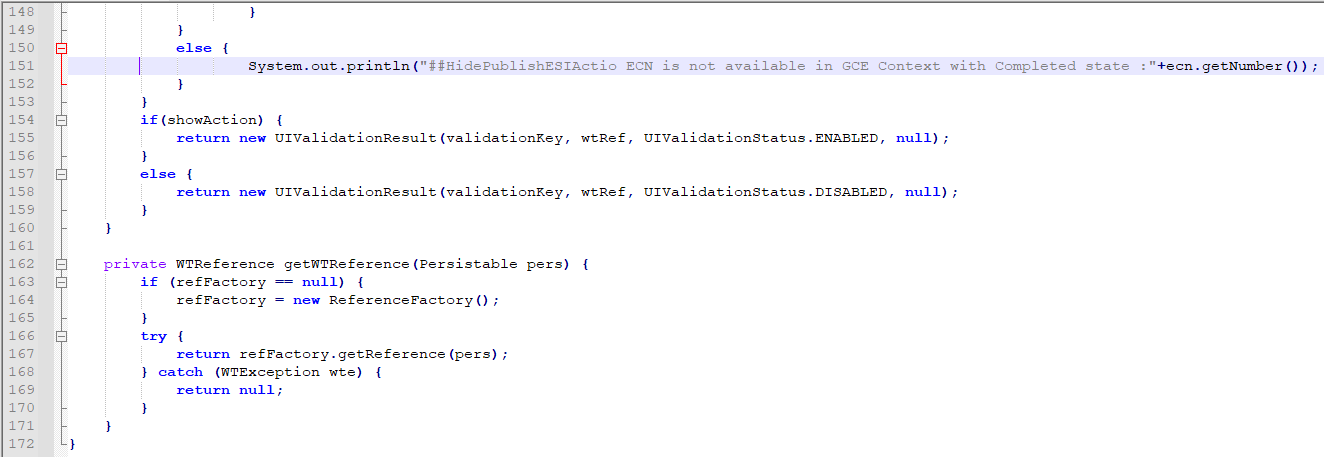
1. Below method is also OOTB method and in this method we have to provide our custom action internal name to perform action validation.



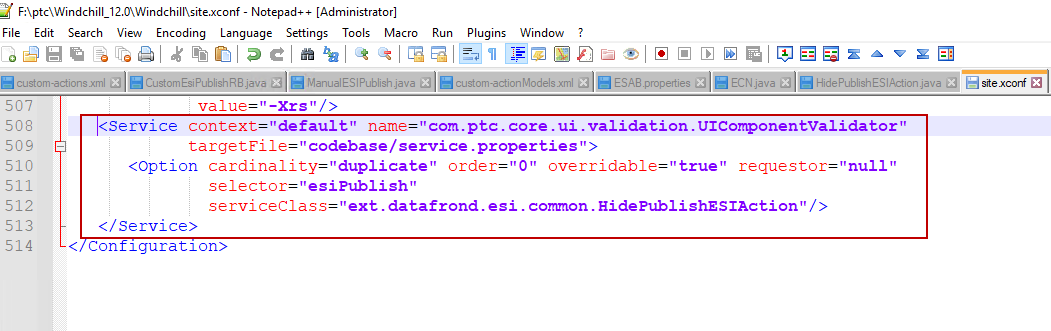
1. Provide the custom code in below method to hide the action if the user is not available in specific group and ECN is not at particular state.







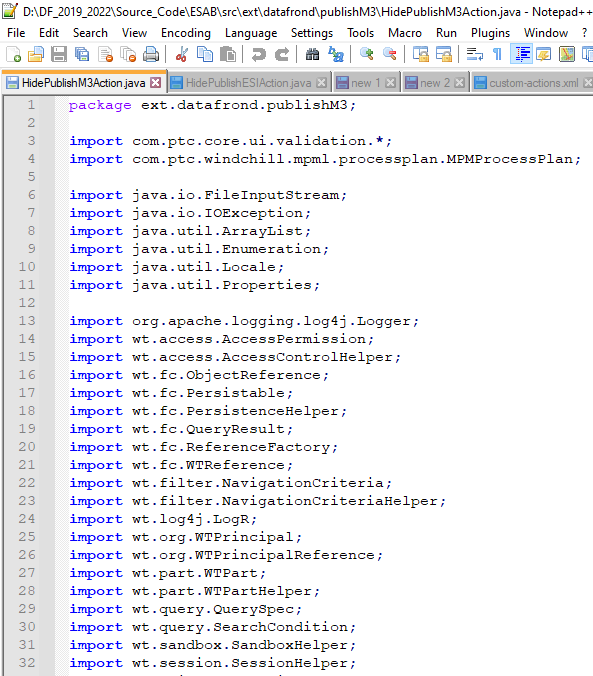
1. Compile the code and ensure new class file got generated.
2. Add the below property in site.xconf file.

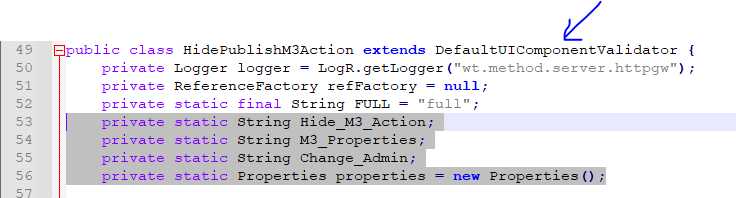


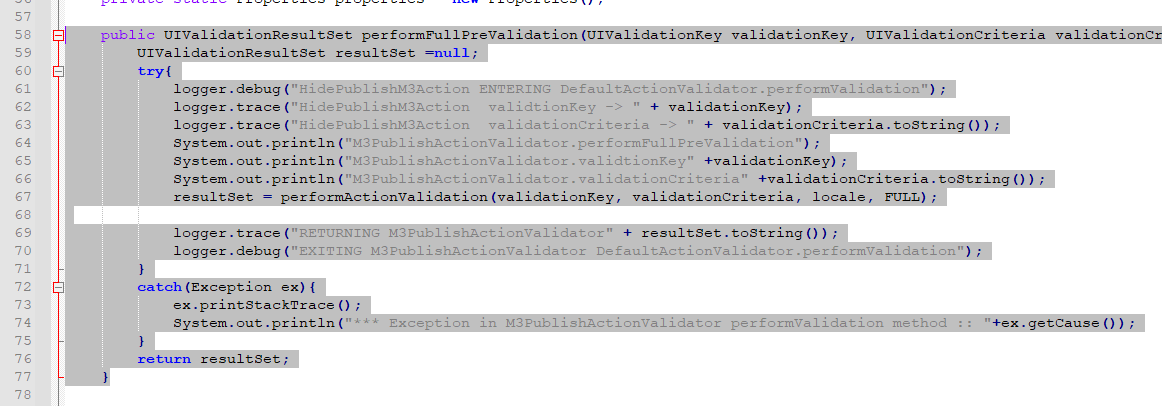
1. Execute xconfmanager –p and restart Windchill.

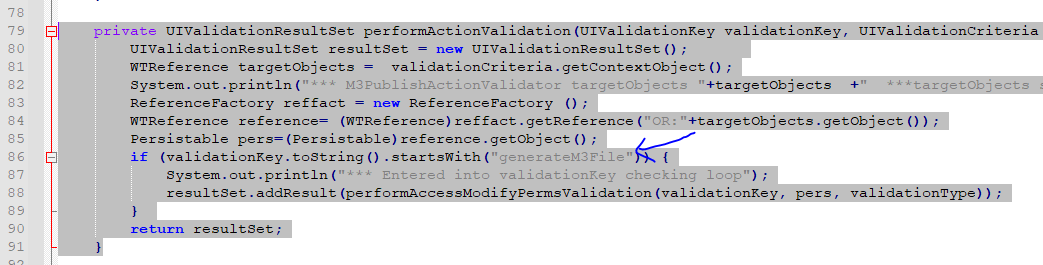
## Example-2:

1. Please refer the below screenshots for another action of same functionality.

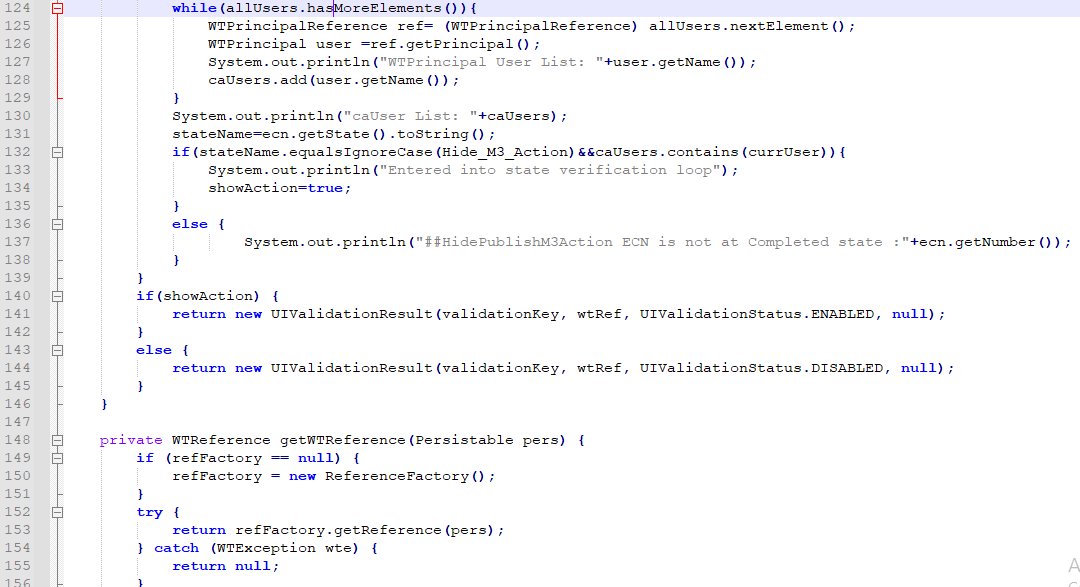












Timeline

Description automatically generated

## **PTC Article Links**

1. List of possible PersistenceManagerEvent events in Windchill

<https://www.ptc.com/en/support/article/CS214710>