

Example

In this sample, we will extend the current Oracle View DROP statement. In the Default template, we have the Oracle view drop function. In this sample, the simple macro is presented, and it generates the following text:

```
DROP VIEW view_name;
```

See the following sample:

```
#macro (dropView $data )
#writeLine( "DROP VIEW $utils.getQualifiedNames( $data )${semicolon}"
$nospace)
#end
```

We will add a new tag to identify the "CASCADE CONSTRAINTS" clause in the DROP VIEW statement. The following script will be generated:

```
DROP VIEW view_name CASCADE CONSTRAINTS;
```

There are three steps to do this:

Step #1. Creating a new stereotype and tags

We need to create a new stereotype with the Boolean property, or extend the default "View" stereotype with a new property. Let's name a new property "cascade_clause", apply the stereotype to the View object, and set value to *true*.

We added a new tag to the view stereotype in the default profile.

Step #2. Changing the template file

Add the following lines to the template file:

```
#set($cascadeOption =false)
##this line is required to for setting a new variable to the false
state.
#set($cascadeOption = $oracleHelper.getBooleanValueFromDefaultProfile
($data,$VIEW_STEREOType, "cascade_clause") )
```

This line retrieves a boolean value from the given tag in the given stereotype "\$view_stereotype" and sets it to a newly created \$cascadeOption value.

Then add a new checking clause into the drop statement. See the final method:

```
#macro (dropView $data )
## sets a value to a new variable
#set($cascadeOption = $oracleHelper.getBooleanValueFromDefaultProfile
($data,$VIEW_STEREOType, "cascade_clause") )
#writeLine( "DROP VIEW $utils.getQualifiedNames( $data )
#if ($cascadeOption) CASCADE CONSTRAINTS
#end${semicolon}" $nospace)
#end
```