

## Creating a COMTOR Module

To create a COMTOR module, create a class that implements the `ComtorDoclet` interface.

Therefore, it must define the following methods:

- `public Properties analyze(RootDoc root)`– This method does whatever is needed to analyze the code. It should return a property list containing the output of the module. The property names in this returned list should follow the following convention. A three-character property name composed of numbers will become a header in the report. To create a subheading under heading xxx, the property name xxx.yyy is used where yyy is a three-character string of numbers. Information under this subheading should have a property name of xxx.yyy.z where z is a string of undefined length. The last parameter set by this function should have the name “score” and the contents should be the grade returned by `getGrade()`.
- `public void setGradingBreakdown(String section, float maxGrade)`– This method is used to indicate the maximum number of point obtainable for each section of the module. A module should be broken down into the different characteristics that it analyses; each of these becomes a section. It is possible that the entire module is a single section. This method will most likely just store the information for later use in the `getGrade()` function.
- `public float getGrade()`– This method is should return the grade for the code analyzed. It should use the maximum grades assigned through the `setGradingBreakdown()` function and the parameters assigned to determine this grade.

- `public void setGradingParameter(String param, String value)`– This method is designed to allow professors to customize the module. You define parameter names that influence the way that the `analyze()` function runs. If you do not have any parameters, give this function an empty body.
- `public void setConfigProperties(Properties props)`– This method is present to allow configuration information to be passed in from the Java configuration file. This is useful for configuration information such as paths to essential files. If you do not need any, give this function an empty body.

After compiling the module, you need to add it to the web interface. To do so,

1. Log into the web interface as an administrator.
2. Click the Add Doclet link in the menu.
3. Fill in the form with the following details:
  - The name that the user will see for this module.
  - The name of the java class created including the package information (Ex: `com.tor.analyzers.CheckForTags`).
  - A description that the user will see for the module.
  - The name and description of the grading sections for the module along with the default grade for each.
  - The name and description of the parameters for the module along with the default value for each.
4. Ensure that the information is correct, as there is currently no interface to edit modules without manually doing so in the database.
5. Submit the form.