**GISEle code structure rules**

**General structure:**

* Maximum Line Length: 79 characters for coding lines and 72 characters to docstrings and comments.
* Tab indentation (4 spaces).
* Break lines before binary operators (+, - ).
* Imports on the top of the code following the hierarchy: standard library, third-party, local.
* Single quote for code strings and double quote for docstrings.
* Don't compare boolean values to True or False using ==:
* For sequences, (strings, lists, tuples), use the fact that empty sequences are false:

**Blank lines:**

* Surround top-level function and class definitions with two blank lines.
* Method definitions inside a class are surrounded by a single blank line.
* Extra blank lines may be used (sparingly) to separate groups of related functions. Blank lines may be omitted between a bunch of related one-liners (e.g. a set of dummy implementations).
* Use blank lines in functions, sparingly, to indicate logical sections.

**Comments:**

* Comments that contradict the code are worse than no comments. Always make a priority of keeping the comments up-to-date when the code changes!
* Comments should be complete sentences. The first word should be capitalized, unless it is an identifier that begins with a lower case letter (never alter the case of identifiers!).
* You should use two spaces after a sentence-ending period in multi- sentence comments, except after the final sentence.
* Block comments generally apply to some (or all) code that follows them, and are indented to the same level as that code. Each line of a block comment starts with a # and a single space (unless it is indented text inside the comment).
* Paragraphs inside a block comment are separated by a line containing a single #.
* Use inline comments sparingly. An inline comment is a comment on the same line as a statement. Inline comments should be separated by at least two spaces from the statement. They should start with a # and a single space.
* Inline comments are unnecessary and in fact distracting if they state the obvious.

**Docstrings:**

* Write docstrings for all public modules, functions, classes, and methods. Docstrings are not necessary for non-public methods, but you should have a comment that describes what the method does. This comment should appear after the def line.
* Note that most importantly, the """ that ends a multiline docstring should be on a line by itself

**Consistency is key!**