

We are using pycodestyle as a linter. Its documentation can be found at this link:

<https://pycodestyle.pycqa.org/en/latest/>

Here are its rules:

- Indentation rules:
  - Use 4 spaces for indentation
  - There should be no python errors due to indentation
  - There should be no extra indentations where they are unnecessary.
  - “Continuation lines” must be indented
  - Continuation lines must not have the same indentation as the next logical line.
  - Do not put a line break to continue a line directly after a binary operator
- Whitespace
  - No whitespace before or after opening parentheses
  - No whitespace before closing parenthesis
  - No whitespace before colons
  - No tabs before or after operators
  - Exactly one space before and after an operator
  - A space should follow every comma, semicolon, or colon
  - Tabs or multiple whitespaces should not follow a comma
  - All comments begin with # followed by a space and not quotes
  - Keywords must be followed by a space
  - Keywords should not be followed by tabs or more than one space
  - No trailing whitespace at the end of a line.
- Blank Lines
  - 2 empty lines should follow every function or class declaration
  - Nested definitions must be preceded by a single blank line
  - Never have a blank line after a decorator
  - Blank lines should not contain whitespace
  - A new line should be present at the end of every file.
- Imports
  - All imports are at the top of the file
  - Only one import per line
- Line length
  - Lines must not be longer than 79 characters
- Statements
  - Do not end statements with semicolons
  - Only one statement should be on each line.
  - comparison to True should be ‘if cond is True:’ or ‘if cond:’
  - comparison to None should be ‘if cond is None:’
  - Any test for membership should be “not in”
  - Any test for object identity should be “is not”
  - Do not compare types, use isinstance()
  - Specify any exceptions, do not use a bare except
  - Do not assign a name to a lambda, instead use a def

- Do not name anything "I" or "O" or "l"
- Deprecated statements
  - Do not use <>, use != instead
  - Use "in" instead of .has\_key()
  - Use repr() instead of backticks
  - Do not use async or await as names.