\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[Announcements](https://learn.umgc.edu/d2l/lms/news/main.d2l?ou=766913) Python Application Template - PyLint Scoring

Python Application Template - PyLint Scoring

Posted May 10, 2023 12:01 AM

IMPORTANT - Make sure to look at all of the announcements and posts in the "Ask the Professor" sections. There is a lot of data already populated.

Here is an example application Template that you can use to base your Python applications off. Make sure to run Pylint or some other PEP8 code checking application on your submitted work. Assignments require that you use the PEP Python Style guide mentioned in the book and found here:

<https://www.python.org/dev/peps/pep-0008/>

Some examples of Python Coding Style best practices include:

* Limit all lines to a maximum of 79 characters (**for this class we will deviate and use 100 character width - the pylint default**) . I have attached a .pylintrc file that will enforce 100 characters for line length.
* Imports are always put at the top of the file, just after any module comments and before module globals and constants.
* Use 4 spaces for indentation.

Using a lint application like Pylint will automate these types of checks for you before you submit your code for evaluation. I will be grading code using Pylint. **You must receive an 80% or greater in Pylint on your code. You will lose -10pts for poorly coded Pylint scores.**

**A screenshot of a computer

Description automatically generated with medium confidence**

We will be following the PEP8 guidance on line length with a slight deviation - normally we would enforcing a line length of 79 characters, but for this class we will agree to use 100.  Pylint can do this for you if you create a .pylintrc file (I have attached one) with the following contents:

[FORMAT]  
# Maximum number of characters on a single line.  
max-line-length=100

Pylint actually uses 100 as its default setting. I have provided the .pylintrc file to encourage you to use one and incorporate it into your development practices.  You can see when enforcing 100 characters this file is rated a 10/10 when line 21 doesn't have the extra "# XXXXXXXXXXXXXXXXXXXX". But - with that violating line the code gets rated 9.47/10.  Please make sure you are enforcing the 100 character constraint and fixing lines that are too long.

For line length - you can just scan with a command-line option of:

pylint --max-line-length=100  hello\_world.py

NOTE: If you download the pylintrc.txt file you will need to rename it when you get it to your local machine from pylintrc.txt --> .pylintrc.

Respectfully,

Craig Poma  
Adjunct Associate Professor  
[craig.poma@faculty.umgc.edu](mailto:craig.poma@faculty.umgc.edu)

**Attachment(s)**

[hello\_world.py](https://learn.umgc.edu/d2l/le/news/widget/766913/FileProvider?newsId=3013145&fileId=74985716)

(730 Bytes)

[pylintrc.txt](https://learn.umgc.edu/d2l/le/news/widget/766913/FileProvider?newsId=3013145&fileId=74985717)

(78 Bytes)