Pylint

One of the most comprehensive static analysis tool currently available for python

1. Pylint is a tool which analyzes a Python program in order to look for various flows, warnings or potential errors. It is also one of the oldest static analyses tool currently available in Python being created by a French company called LogiLock.

Pylint is a tool that flags suspicious usage in software written in Python and also:

- A style checker, which tries to enforce the PEP 8 rules(style guide for python code).
- A type checker(it looks for various type errors in the program) such as adding strings to objects, raising objects which are not exceptions, or unpacking too many items in too few variables.
- A structural analyzer looking for various design anti-patterns that the code might have or various bad implementations of the methods and so on.

Pylint examples of returned warnings/errors:

- Unreachable code
- Statement seems to have no effect
- Unused variables
- Unused import and so on

Pylint can be used to detect more serious flows and bugs such as to use undefined variables, accessing undefined members, calling objects which arent callable.
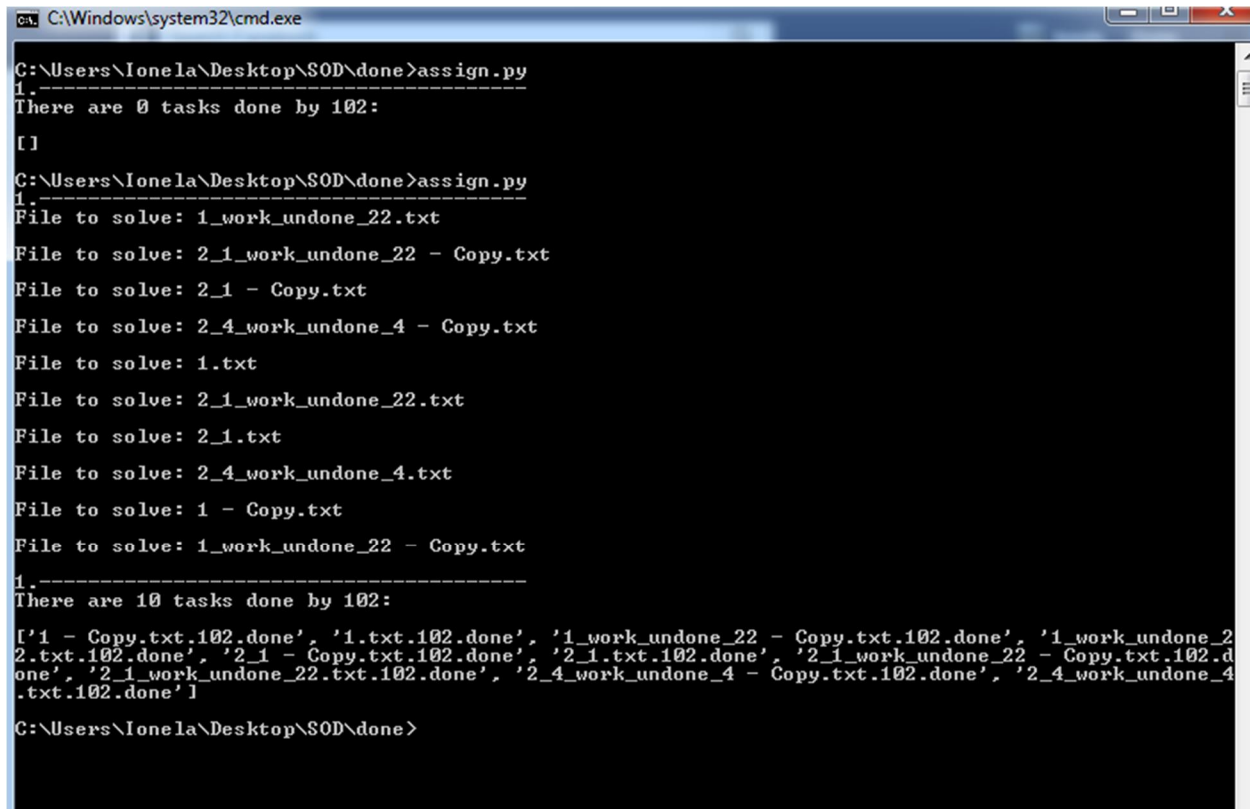
Pylint actually is not one component, meaning that its business logic is split across 2 components: pylint itself that basically is the project with the patterns of problems in the code, and the second component is the component that understands Python.

2. SOD project

SOD is a program written in python that basically gets a list of files from work folder, shuffles the list, and then processes the list and changes the extensions of the file, once into assign, showing that the file is being processed, and finally to done showing that

the file was processed. The files can be accessed by different users so conditions and different cases were treated in order to avoid errors of conflict.

A print of the program being run is available bellow:

```
C:\Windows\system32\cmd.exe

C:\Users\Ionela\Desktop\SOD\done>assign.py
1.-------------------------------------------
There are 0 tasks done by 102:

[]

C:\Users\Ionela\Desktop\SOD\done>assign.py
1.-------------------------------------------
File to solve: 1_work_undone_22.txt

File to solve: 2_1_work_undone_22 - Copy.txt

File to solve: 2_1 - Copy.txt

File to solve: 2_4_work_undone_4 - Copy.txt

File to solve: 1.txt

File to solve: 2_1_work_undone_22.txt

File to solve: 2_1.txt

File to solve: 2_4_work_undone_4.txt

File to solve: 1 - Copy.txt

File to solve: 1_work_undone_22 - Copy.txt

1.-------------------------------------------
There are 10 tasks done by 102:

['1 - Copy.txt.102.done', '1.txt.102.done', '1_work_undone_22 - Copy.txt.102.done', '1_work_undone_2
2.txt.102.done', '2_1 - Copy.txt.102.done', '2_1.txt.102.done', '2_1_work_undone_22 - Copy.txt.102.d
one', '2_1_work_undone_22.txt.102.done', '2_4_work_undone_4 - Copy.txt.102.done', '2_4_work_undone_4
.txt.102.done']

C:\Users\Ionela\Desktop\SOD\done>
```

After running the program, 10 files were processed by user 102. The files are provided in the screenshot above.

3. Results after running pylint analiser tool:

```
C:\Windows\system32\cmd.exe

Report
======
85 statements analysed.

Statistics by type
------------------

+-----------+---------+------------+------------+-------------+-----------+
|type       |number   |old number  |difference  |%documented  |%badname   |
+===========+=========+============+============+=============+===========+
|module     |1        |NC          |NC          |0.00         |0.00       |
+-----------+---------+------------+------------+-------------+-----------+
|class      |0        |NC          |NC          |0            |0          |
+-----------+---------+------------+------------+-------------+-----------+
|method     |0        |NC          |NC          |0            |0          |
+-----------+---------+------------+------------+-------------+-----------+
|function   |6        |NC          |NC          |0.00         |33.33      |
+-----------+---------+------------+------------+-------------+-----------+


Raw metrics
-----------

+-----------+---------+------+----------+-----------+
|type       |number   |%     |previous  |difference |
+===========+=========+======+==========+===========+
|code       |87       |70.16 |NC        |NC         |
+-----------+---------+------+----------+-----------+
|docstring  |0        |0.00  |NC        |NC         |
+-----------+---------+------+----------+-----------+
|comment    |4        |3.23  |NC        |NC         |
+-----------+---------+------+----------+-----------+
|empty      |33       |26.61 |NC        |NC         |
+-----------+---------+------+----------+-----------+


Duplication
-----------

+-------------------------+------+----------+-----------+
|                         |now   |previous  |difference |
+=========================+======+==========+===========+
|nb duplicated lines      |0     |NC        |NC         |
+-------------------------+------+----------+-----------+
|percent duplicated lines |0.000 |NC        |NC         |
+-------------------------+------+----------+-----------+
```

```
+------------------------------------------+----------+----------+-------------+

Messages by category
---------------------

+-------------+--------+----------+------------+
|type         |number  |previous  |difference  |
+=============+========+==========+============+
|convention   |98      |NC        |NC          |
+-------------+--------+----------+------------+
|refactor     |0       |NC        |NC          |
+-------------+--------+----------+------------+
|warning      |71      |NC        |NC          |
+-------------+--------+----------+------------+
|error        |0       |NC        |NC          |
+-------------+--------+----------+------------+


Messages
--------

+----------------------+-------------+
|message id            |occurrences  |
+======================+=============+
|mixed-indentation     |68           |
+----------------------+-------------+
|bad-whitespace        |43           |
+----------------------+-------------+
|trailing-whitespace   |26           |
+----------------------+-------------+
|invalid-name          |22           |
+----------------------+-------------+
|missing-docstring     |7            |
+----------------------+-------------+
|redefined-outer-name  |2            |
+----------------------+-------------+
|redefined-builtin     |1            |
+----------------------+-------------+


Global evaluation
-----------------
Your code has been rated at -9.88/10

C:\Users\Ionela\Desktop\SOD\done>
```

According to the report above for SOD project there were 85 statements analyzed.

Here are the some mentioned conclusions after checking the code with Pylint:

- the project has one module and 6 functions, and among the 6 functions, 33,33 % of them are not named according to python rules. The functions are provided by the tool.

- there were no duplicated lines in the project.

- there were 71 warnings and 0 errors.

- more findings are available in the screenshot above.

At the end, the tool makes a global evaluation of the project. It actually gives a rating to the project according to python standards: %our code has been rated as 9.88/10.+

4. Conclusion

A code quality assurance should be used because it enforces a consistent coding standard across the projects, it improves the general code quality of the project and it creates a bug legacy system in which no or very few tests are needed.

As a conclusion after running the tool on the SOD project, the analyzed program has a lot of warnings, missed-indentations, bad-white spaces, it is not documented and it does not respect the standard rules of naming the variables, therefore 15 invalid names were found. Before running the analysis tool, the project seemed to be flawless, but it turned out that some coding standard rules were not taken in consideration.

Here is a link to the open source project on which the tool was used:

  https://github.com/ionela23/SOD

Sworn declaration
I hereby declare, under oath, that this master thesis has been my independent work and has not been aided with any prohibited means. I declare, to the best of my knowledge and belief, that all passages taken from published and unpublished sources or documents have been reproduced whether as original, slightly changed or in thought, have been mentioned as such at the corresponding places of the thesis, by citation, where the extent of the original quotes is indicated.

The homework that violates the above statement will be rejected.

Ionela Ababi

MSD2