

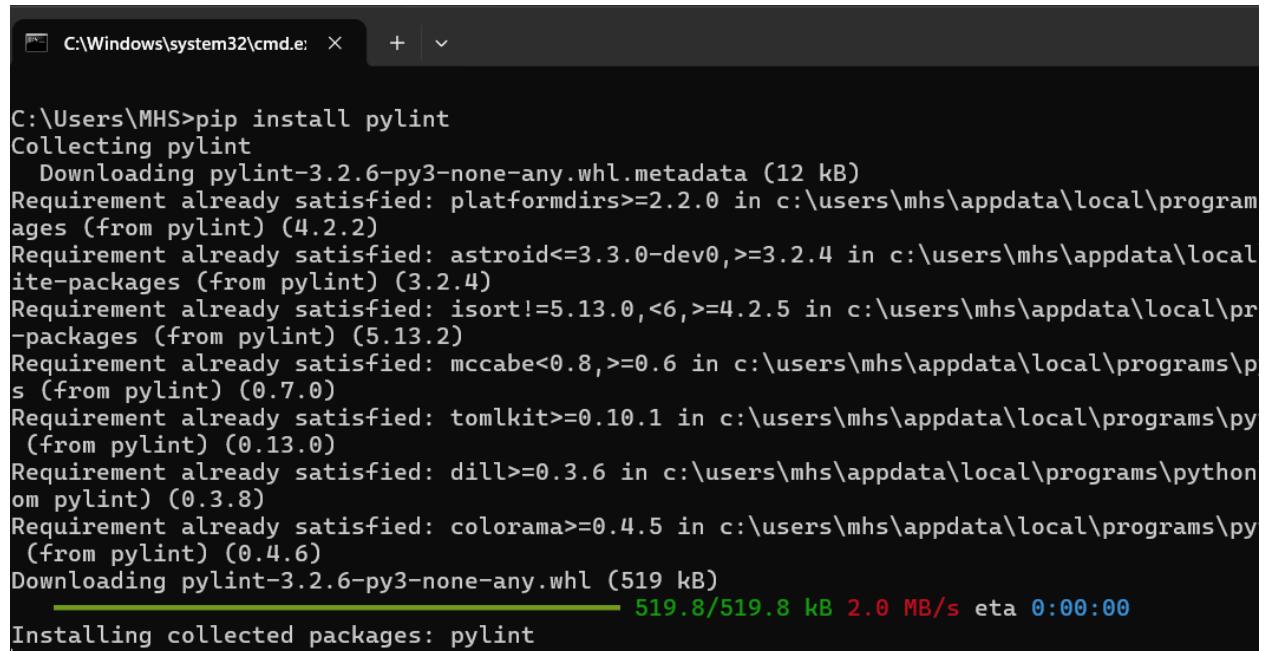
Modul: Menulis Clean Code Python dengan Bantuan Pylint

Pylint adalah alat statis untuk analisis kode Python yang digunakan untuk memeriksa kesesuaian kode terhadap standar penulisan (coding standards) dan mengidentifikasi potensi kesalahan, bug, atau masalah lain dalam kode. Pylint memberikan skor keseluruhan untuk kualitas kode dan menghasilkan laporan yang mencakup saran perbaikan, seperti penggunaan variabel yang tidak diperlukan, import yang tidak digunakan, atau inkonsistensi dalam gaya penulisan. Alat ini sangat berguna untuk memastikan kode yang bersih, dapat dibaca, dan bebas dari bug.

Pada praktikum ini, kita akan mempelajari cara menggunakan Pylint untuk menganalisis dan meningkatkan kualitas kode Python.

Percobaan 1: Instalasi Pylint

Buka command prompt pada sistem operasi anda lalu instal Pylint menggunakan pip dengan perintah:



```
C:\Users\MHS>pip install pylint
Collecting pylint
  Downloading pylint-3.2.6-py3-none-any.whl.metadata (12 kB)
Requirement already satisfied: platformdirs>=2.2.0 in c:\users\mhs\appdata\local\program
ages (from pylint) (4.2.2)
Requirement already satisfied: astroid<=3.3.0-dev0,>=3.2.4 in c:\users\mhs\appdata\local
ite-packages (from pylint) (3.2.4)
Requirement already satisfied: isort!=5.13.0,<6,>=4.2.5 in c:\users\mhs\appdata\local\pr
ojects (from pylint) (5.13.2)
Requirement already satisfied: mccabe<0.8,>=0.6 in c:\users\mhs\appdata\local\programs\p
y3\python3.10\lib\site-packages (from pylint) (0.7.0)
Requirement already satisfied: tomlkit>=0.10.1 in c:\users\mhs\appdata\local\programs\py
thon3\python3.10\lib\site-packages (from pylint) (0.13.0)
Requirement already satisfied: dill>=0.3.6 in c:\users\mhs\appdata\local\programs\pyt
hon3\python3.10\lib\site-packages (from pylint) (0.3.8)
Requirement already satisfied: colorama>=0.4.5 in c:\users\mhs\appdata\local\programs\py
thon3\python3.10\lib\site-packages (from pylint) (0.4.6)
  Downloading pylint-3.2.6-py3-none-any.whl (519 kB)
                                           519.8/519.8 kB 2.0 MB/s eta 0:00:00
Installing collected packages: pylint
```

Percobaan 2: Menjalankan pylint di terminal

Pada workspace di VSCode, Buatlah sebuah file dengan nama **percobaan2.py** lalu salin kode dibawah ini (silahkan copy-paste):

```
rr=float(input('Input jari-jari: '))
```

```

pa=float(input('Input panjang: '))
le=float(input("Input lebar: "))

import math, dataclasses

luasLingkaran = math.pi * rr * rr
keliling_lingkaran = 2 * math.pi * rr

luas_persegi_panjang = pa * le
keliling_persegi_panjang = 2 * (pa + le)

print(f"Luas Lingkaran: {luasLingkaran}")
print("Keliling Lingkaran: ", keliling_lingkaran)
print("Luas Persegi Panjang: ", luas_persegi_panjang)
print("Keliling Persegi Panjang: ", keliling_persegi_panjang)

```

Jalankan kode diatas pada terminal. Pastikan output dapat ditampilkan tanpa error. Screenshot hasilnya disini:

```

PS C:\Users\WHS\Documents\3312501108> & C:/Users/WHS/AppData/Local/Program
13/python.exe c:/Users/WHS/Documents/3312501108/p06/percobaan.py
Input jari-jari: 4
Input panjang: 6
Input lebar: 5
Luas Lingkaran: 50.26548245743669
Keliling Lingkaran: 25.132741228718345
Luas Persegi Panjang: 30.0
Keliling Persegi Panjang: 22.0

```

Selanjutnya, kita akan melakukan analisis kode statis menggunakan pylint. Masih pada terminal, jalankan perintah ini:

```
pylint percobaan2.py
```

Perhatikan kesalahan-kesalahan kode, rekomendasi dan skor yang didapat dari hasil:

```

coba.py:1:0: C0114: Missing module docstring (missing-module-docstring)
coba.py:5:0: C0410: Multiple imports on one line (math, dataclasses) (multiple-imports)
coba.py:5:0: C0413: Import "import math, dataclasses" should be placed at the top of the module (wrong-import-position)
coba.py:5:0: C0413: Import "import math, dataclasses" should be placed at the top of the module (wrong-import-position)
coba.py:5:0: W0611: Unused import dataclasses (unused-import)

-----
Your code has been rated at 5.83/10 (previous run: 7.50/10, -1.67)

```

Skor diatas menunjukkan nilai 5.83 dari 10. Artinya masih banyak yang perlu diperbaiki agar kode lebih clean.

Kita juga bisa menggunakan parameter pada pylint untuk menampilkan detail lengkap dan lebih colorful (agar lebih enak dibaca). Jalankan perintah berikut:

```
pylint percobaan2.py -r y --output-format=colorized
```

Bagaimana hasilnya? Screenshot disini:

The screenshot shows the PyCharm IDE's 'Messages' tool window. It displays a table of code analysis results with the following data:

message id	occurrences
wrong-import-position	2
unused-import	1
multiple-imports	1
missing-module-docstring	1
missing-final-newline	1

At the bottom of the window, a message states: "Your code has been rated at 5.00/10 (previous run: 5.00/10, +0.00)".

Penjelasan:

- **-r y** → tampilkan laporan lengkap (default biasanya **-r n**).
- **--output-format=colorized** → biar hasilnya lebih mudah dibaca.

Apakah bedanya dengan format sebelumnya?

Ya sangat berbeda

Dari kelima komponen diatas, maka kita dapat memperbaikinya menjadi seperti berikut ini:

```
"""Program sederhana menghitung luas dan keliling lingkaran serta persegi panjang."""

import math

rr = float(input('Input jari-jari: '))
pa = float(input('Input panjang: '))
le = float(input("Input lebar: "))

luas_lingkaran = math.pi * rr * rr
keliling_lingkaran = 2 * math.pi * rr

luas_persegi_panjang = pa * le
```

```
keliling_persegi_panjang = 2 * (pa + le)

print(f"Luas Lingkaran: {luas_lingkaran}")
print("Keliling Lingkaran: ", keliling_lingkaran)
print("Luas Persegi Panjang: ", luas_persegi_panjang)
print("Keliling Persegi Panjang: ", keliling_persegi_panjang)
```

Jalankan kembali pylint. Berapa skornya kali ini? Screenshot hasilnya disini:

The screenshot shows a terminal window with the following output:

```
Messages
-----
+-----+-----+
| message id      | occurrences |
+=====+=====+
| missing-final-newline | 1           |
+-----+-----+
-----
Your code has been rated at 9.17/10 (previous run: 5.00/10, +4.17)
```

Masih ada banyak lagi parameter yang dapat kita gunakan. Untuk informasi lengkap tentang pylint, gunakan perintah:

```
pylint --help
```

lint

Tugas:

1. Pilih 2 kode program percabangan ataupun perulangan pada praktikum sebelum-sebelumnya. Analisis kode tersebut menggunakan Pylint. Jika ada kesalahan, maka perbaiki berdasarkan rekomendasi Pylint.

Kode program #1 yang anda pilih:

```
nilai = int(input("Input sebuah nilai: "))

if nilai%2==0:
    print("Bilangan Genap")
else:
    print("Bilangan Ganjil")
```

Screenshot hasil analisis kode #1 dari Pylint **sebelum** diperbaiki:

```
C:\Users\User\Documents\Tugas Oktober>pylint hihi.py
*****
Module hihi
hihi.py:6:0: C0304: Final newline missing (missing-final-newline)
hihi.py:1:0: C0114: Missing module docstring (missing-module-docstring)

-----
Your code has been rated at 5.00/10
```

Kode program #1 setelah diperbaiki:

```
"""percabangan"""
nilai = input("Masukkan Nilai: ")

nilai = float(nilai)

if nilai >= 60:
    print("Anda Lulus")
else:
    print("Anda gagal")
```

Screenshot hasil analisis kode #1 dari Pylint **setelah** diperbaiki:

```
C:\Users\User\Documents\Tugas Oktober>pylint hihi.py

-----
Your code has been rated at 10.00/10 (previous run: 8.00/10, +2.00)
```

Kode program #2 yang anda pilih???

```
for i in range(10):
    print(i, 'Helo world! Inrement!')

for i in range(1,11):
    print(i, 'Hello World! Increment')

print('=====')

for i in range(10, 0, -1):
    print(i, 'Hello Wolrd! Decrement')
```

```
print('=====')  
\
```

Screenshot hasil analisis kode #2 dari Pylint **sebelum** diperbaiki:

```
C:\Users\User\Documents\Tugas Oktober>pylint hihi.py  
***** Module hihi  
hihi.py:12:0: C0304: Final newline missing (missing-final-newline)  
hihi.py:1:0: C0114: Missing module docstring (missing-module-docstring)  
  
-----  
Your code has been rated at 7.50/10 (previous run: 10.00/10, -2.50)
```

Kode program #2 setelah diperbaiki:

```
"pengulan dengan i range"  
for i in range(10):  
    print(i, 'Helo world! Inrement!')  
  
for i in range(1,11):  
    print(i, 'Hello World! Increment')  
  
print('=====')  
  
for i in range(10, 0, -1):  
    print(i, 'Hello Wolrd! Decrement')  
  
print('=====')
```

Screenshot hasil analisis kode #2 dari Pylint **setelah** diperbaiki:

```
C:\Users\User\Documents\Tugas Oktober>pylint hihi.py  
  
-----  
Your code has been rated at 10.00/10 (previous run: 8.75/10, +1.25)
```