Assignment 2 CodeSmell

Vanessa Martinez Montero

Language: Python

Code Smell detection tool name: PyLint

I HAVE UPDATED THE GITHUB WITH THE NEW CODE IF YOU NEED THE OLD ONE LET ME KNOW

Target Project: <https://github.com/FishSticks-stack/United-States-Offenses-2019>

This is a project I created back in 2021. It displays an interactive map of the united states and includes a bar graph option in the drop down menu. Data displayed is the total number of offenses in each state in 2019.

Results of running code smell on the project: when I ran the pylint on my ‘testDash.py’ file it showed me a list of code smell issues within my project and included the line number at which they occurred, it gave me an overall code rating of 3.95/10. I also saw that it was using snake\_case as the default naming style (see screenshot below), I had to change the naming style to be set to camelCase, since that is my preferred naming style. After I ran it again using camelCase, my code rating went up to 4.19/10, (see 2nd screenshot posted).

Text

Description automatically generated

What code smells were found: I noticed that each time I ran the pylint command, I would end up with more code smells than the previous test (see screenshot below). I got many code smells of lines being too long and I was still getting smells for snake\_case, after changing it to camelCase. It looks like a majority of my smells are coming from naming styles.Text

Description automatically generated

Optional

How were you able to remove the design smells: removed the unused import that I used to download pylint. Changed the order of the imports. Changed the naming style of variables and some functions. Attempted to do something on the ‘line too long’ portion the first time. After first attempt ran the code to make sure it works correctly and rerun pylint to see updated code smells. This is the code smells after my first attempt, the code rating was bumped up to a 4.29. Text

Description automatically generated

After deleting extra code from line 19, in an attempt to make it smaller and remove the ‘line too long’ code smell, I reran and got a code rating of 6.90/10. Text

Description automatically generated

When I reran the pylint some of the ‘line too long’ code smells went away and I was able to get a higher code rating of 7.50/10. Some of the code smells I didn’t completely understand, but after looking them up I’ve been able to fix them. I deleted some parts of code that related to the dataset, that I thought I needed, because I remember having to do that in order to avoid problems I was having when I origanally wrote it in 2021. However, I’ve been able to delete it with no issues and the program still works correctly. Text

Description automatically generated

After fixing all the ‘line too long’ smells, I get a code rating of 9.25/10 with only the bottom 3 code smells left. The code smell for line 21, using a ‘with’, was difficult to fiqure out how to implement because I could not find anything related on google. However, I did eventually find my answer on a website and I was able to implement ‘with’ into reading and opening the json file that contained coordinates for each state in the dataset. Now the rating improved to a 9.76/10.

Graphical user interface, text

Description automatically generated with medium confidence

Now its time for the last code smell. I was avoiding this one because when I tried to implement it, I couldn’t understand how it would work, since the code was for displaying my graphs. After looking up the code smell message, I was able to change my code correctly by removing the ‘else’ and the program still works correctly. Now I am left with no code smells and I got a code rating of 10/10!Graphical user interface, text

Description automatically generated

What must a developer consider prior to refactoring code smells: must consider when changing naming style of functions or variables and make sure they are correctly changed everywhere they are called/used in the program. We must consider throughout the changes that we make, if our code still properly functions.

What other artifacts/assets might be affected: other files that call our main file ‘test\_dash.py’ could be affected by the changes we are making to resolve these code smells. Also I have to consider the changes the code smell wants me to make in lines of code that pertain to my dataset, json, and code used to grab data from these files. For example, code for grabbing data from a specific row or column and isnerting elements into a list in order to read the data properly to make the appropriate graphs/maps work and display the correct information in the dataset.

Side note: I chose this old project I made in 2021 because I knew it had a lot of junk code that I knew could be improved, but at the time I did not know how to do so, since it was my first attempt at a project and dataset this large. I really enjoyed this project so I had always wanted to go back and improve the code and make it better. Im really happy with how it turned out and its going to help me with future projects that I make that involve datasets.