Code smell 1 & Refactor

Date Found: 12/6/2023

Smell: **Dead Code** with update\_task() function. This function is no longer being used at all so it can be safely deleted.

Refactor: **Removing** the dead code from db\_crud.py. This will not break any functionalities since the update\_task() function was not being used anymore.

Location in Source tree: src 🡪 db\_crud.py

Code Smell 2 & Refactor

Date Found: 12/2/2023

Smell: **No SRP violation**. Db\_crud.py should only have functions that manipulate the sqlite database. Instead it has functions that help with dates and times as well.

Refactor: **Separating** all date and time conversion functions into their own separate modules. Now the convert\_string\_date\_to\_object() function and convert\_date\_string\_to\_standard() function are both in a DateHelper class. The convert\_to\_time() function is also in its own TimeHelper class now.

Location in Source Tree: src 🡪 db\_crud.py

Code Smell 3 & Refactor

Date Found: 12/2/2023

Smell: Multiple instances of **Magic Number** violation in program.py. Specifically when parsing the user input, there are several different indexes of arrays accessed throughout the program. Currently these seem like arbitrary numbers without a variable name to show what they represent.

Refactor: **Replace all** seemingly arbitrary indexes with meaningful variable names.

Location in Source Tree: src 🡪 program.py

Code Smell 4 & Refactor

Date Found: 12/3/2023

Smell: **Dead Code & Coupling** in program.py & db\_crud.py with many unnecessary print() statements. Some of these were used for debugging during development, but many of them simply print information that the user doesn’t need. Ex: certain print statements providing backend information like NoneType datatypes.

Refactor: **Remove** nearly all print() statements that don’t provide the user with relevant/valuable information. Any relevant print statements are being centralized to program.py to control the application, rather than have overlap & coupling in db\_crud.py with logging statements.

Location in Source Tree: src 🡪 program.py ***and*** src 🡪 db\_crud.py

Code Smell 5 & Refactor

Date Found: 12/5/2023

Smell: **Comments** are outdated and unnecessary in the main program file. Some of the if/else statement chains also need new comments to explain the logic behind their intent.

Refactor: **Remove** unnecessary comments, **update** old comments to reflect the current state of the script, and **add** new comments where necessary for further information.

Location in Source Tree: src 🡪 program.py

Code Smell 6

Date Found: 12/4/2023

Smell: **large parameter** list for the create\_task() function in db\_crud.py. All these parameters are necessary for the function to do it’s job, but it still make the function a little more involved to use properly.

Location in Source Tree: src 🡪 db\_crud.py

Code Smell 7

Date Found: 12/4/2023

Smell: **No SRP violation** with parse\_input\_string() in program.py. This method singlehandedly controls all command logic for the application. It has many responsibilities and many reasons to change.

Location in Source Tree: src 🡪 program.py

Code Smell 8

Date Found: 12/4/2023

Smell: **High Coupling** between parse\_input\_string() function in program.py and every other module in the project. DateHelper & db\_crud.py are both heavily used for parse\_input\_string to function properly. A slight change of method signatures, names, or return types could easily cause bugs.

Location in Source Tree: src 🡪 program.py

Code Smell 9

Date Found: 12/4/2023

Smell: **Duplicate Code** in program.py with the command **if/else** chain. If the app was being expanded to support a heavy amount of features, this code smell could be easily avoided by creating a class for each command run by the user.

Location in Source Tree: src 🡪 program.py

Code Smell 10

Date Found: 12/4/2023

Smell: **Duplicate** code for nearly every single db\_crud.py when creating and closing the sqlite3 database connections.