Unit 9 ePortfolio Activity

Improving Code Quality

Refer to the Mertz (2019) resource. Use some Python code which you have developed in the past and apply at least 3 of the strategies presented at the source to improve its quality. You can use the Jupyter Notebook workspace in Codio and save your work to your GitHub repository.

Strategy 1: Follow a Style Guide, Consider Indentation

```
def __init__(
        self, title,
        menu_options,
        allow_quit = True):
        self.title = title
        self.options = menu_options
        self.can_quit = allow_quit
        self.quit_option = len(menu_options) + 1
        self.continue_screen = True
```

Strategy 2: Follow a Style Guide, Imports on separate lines

Some imports are not specified on separate lines:

```
import os, re
from screen import Screen
from authentication import Authentication
from safety_store import SafetyStore
```

The code is updated according to the style guidelines to specify each import on a new line

```
import os
import re
from screen import Screen
from authentication import Authentication
from safety_store import SafetyStore
```

Strategy 3: Follow a Style Guide, Line Length

The maximum line length is proposed to be no more than 99 characters. In this source file, the maximum line length is 127 characters

```
        title = "{} (Logged on as: \033[93;40m{}\033[0m, Admin = {})".format(self.title, user_name, Authentication.is_admin())
```

To follow the style guideline, this line of code must be broken into multiple lines using the "\ character, as follows:

```
title = "{} (Logged on as: \033[93;40m{}\033[0m, Admin = {})" \
            .format(self.title, user_name, Authentication.is_admin())
```

References

Mertz, J. (2019). Writing Clean and Pythonic Code. Available from https://trs.jpl.nasa.gov/bitstream/handle/2014/51618/CL%2319-5039.pdf?sequence=1 [Accessed 15 May 2022]