12 Checkpoint: Using Objects

Purpose

Improve your ability to write object-oriented code.

Problem Statement

There are several types of commands that are commonly found in object oriented programs. These types of commands are so common, that a programmer must be able to recognize and write them. One of these types of commands is calling the methods of an object using the dot operator (.) as shown in this template:

```
    variable = object.method(arg1, arg2, ...)
```

Helpful Documentation

In Python, lists are objects with attributes and methods, and a programmer can modify a list by calling those methods. The list methods are documented in a web page titled More on Lists.

Assignment

Write a small Python program named `fruit.py` that demonstrates object oriented programming by modifying a list. Do the following:

1. Open a new blank file in VS Code and save it as `fruit.py`
2. Copy and paste this code at the top of your `fruit` program:

```
def main():
    # Create and print a list named fruit.
    fruit_list = ["pear", "banana", "apple", "mango"]
    print(f"original: {fruit_list}")
```

3. Add code to reverse and print *fruit_list*.
4. Add code to append "orange" to the end of *fruit_list* and print the list.
5. Add code to find where "apple" is located in *fruit_list* and insert "cherry" before "apple" in the list and print the list.
6. Add code to remove "banana" from *fruit_list* and print the list.
7. Add code to pop the last element from *fruit_list* and print the popped element and the list.
8. Add code to sort and print *fruit_list*.
9. Add code to clear and print *fruit_list*.
10. At the bottom of your program write a call to the `main` function.

Testing Procedure

Verify that your program works correctly by following each step in this testing procedure:

1. Run your program and ensure that your program's output is the same as the output shown below.

```
> python fruit.py
original: ['pear', 'banana', 'apple', 'mango']
reversed: ['mango', 'apple', 'banana', 'pear']
append orange: ['mango', 'apple', 'banana', 'pear', 'orange']
insert cherry: ['mango', 'cherry', 'apple', 'banana', 'pear', 'orange']
remove banana: ['mango', 'cherry', 'apple', 'pear', 'orange']
pop orange: ['mango', 'cherry', 'apple', 'pear']
sorted: ['apple', 'cherry', 'mango', 'pear']
cleared: []
```

Ponder

After you finish your program, examine the code in `main` and realize that you wrote object oriented code when you used the `fruit_list` and the dot operator (.) to call the `reverse`, `append`, `insert`, `remove`, `pop`, `sort`, and `clear` methods.

Sample Solution

When your program is finished, view the sample solution [↓] for this assignment to compare your solution to that one. Before looking at the sample solution, you should work to complete this checkpoint program. However, if you have worked on it for at least an hour and are still having problems, feel free to use the sample solution to help you finish your program.

Submission

When complete, report your progress in the associated I-Learn quiz.