IT 210 – Fundamentals of Programming – Lab 8

You will work with a partner or 2 for this lab. However, **each person is responsible to submit his/her final version of the program to D2L**.

Start with the file named WidgetStart.py from D2L. Write code for each of the FIXME seven sections to that your solution produces the same output as the test code that I have provided (after the 'if __name__ == "__main__": line). DO NOT CHANGE ANY TEST CODE!

When I test your solution, I will import your class into my test file and expect it to work similar to how the test code uses it. If your code produces the same results as mine did when the test code executes, it should be good.

Your name must be included as a comment on the first line. The names of others who you teamed with must be as comments on successive lines. Submit a copy of the lab to the D2L dropbox not later than **11:59 pm Friday** evening. Name your file **Lab8.py**. You should ensure that in your copy the team member names have your name first. 20 points will be deducted if you do this lab 'solo'.

A copy of WidgetStart.py is provided here for reference. Note that this program (class) requires no input from you at the terminal.

```
# Start code for Widget class

class Widget():
        # Note that the test code down below does not check if the defaults here are working
        # correctly, but this may be tested while grading.  If you don't change this it
        # will work correctly.
        def __init__(self, color="black", weight=10.0):
                self.color = color
                self.weight = weight

        def __str__(self):
                return "%s widget; = %.2f" % (self.color, self.weight)

        def __lt__(self, other):  # is self lighter than other?
                # FIXME

        def __le__(self, other):  # is self lighter or equal to other?
                # FIXME

        def __gt__(self, other):  # is self heavier than other?
                # FIXME

        def __ge__(self, other):  # is self heavier or equal to other?
                # FIXME

        def __eq__(self, other):  # is self equal in weight to other?
                # FIXME

        def __ne__(self, other):  # is self unequal in weight to other?
                # FIXME
```

```
        # Create a new widget by "adding" two existing widgets.  The weight of the new
        # widget will be the sum of the weights of the two.  The color of the new widget
        # will be the same as the other two if their colors are the same, otherwise,
        # the new widget's color will be "white"
        def __add__(self, other):
                # FIXME


if __name__ == "__main__":

# When this test code executes it should produce the following if your implementation
# of the code above is correct
#
# green widget; weight = 4.50
# red widget; weight = 12.20
# green widget; weight = 9.60
# blue widget; weight = 2.00
#
# Do comparison tests.
# <
# False  False  False  True
# <=
# True  False  False  True
# >
# False  True  True  False
# >=
# True  True  True  False
# ==
# True  False  False  False
# !=
# False  True  True  True
#
# test_widget + others.
# green widget; weight = 9.00
# white widget; weight = 16.70
# green widget; weight = 14.10
# white widget; weight = 6.50
#
# test_widget2 + others.
# white widget; weight = 9.50
# red widget; weight = 17.20
# white widget; weight = 14.60
# white widget; weight = 7.00

        widget_list = []
        widget_list.append(Widget("green", 4.5))
        widget_list.append(Widget("red", 12.2))
        widget_list.append(Widget("green", 9.6))
        widget_list.append(Widget("blue", 2))

        for widget in widget_list:
                print(widget)

        test_widget = Widget("green", 4.5)
        test_widget2 = Widget("red", 5)

        print("\nDo comparison tests.")
```

```
print("<")
for widget in widget_list:
        print(widget < test_widget, end="  ")
print()

print("<=")
for widget in widget_list:
        print(widget <= test_widget, end="  ")
print()

print(">")
for widget in widget_list:
        print(widget > test_widget, end="  ")
print()


print(">=")
for widget in widget_list:
        print(widget >= test_widget, end="  ")
print()

print("==")
for widget in widget_list:
        print(widget == test_widget, end="  ")
print()

print("!=")
for widget in widget_list:
        print(widget != test_widget, end="  ")
print()

print("\ntest_widget + others.")
for widget in widget_list:
        print(widget + test_widget)

print("\ntest_widget2 + others.")
for widget in widget_list:
        print(test_widget2 + widget)
```