Singleton

I am producing a singleton pattern for the main classes of my program that only require one instance of each other.

Class Diagram before the modification:

interpreter\classes\_INTERPRETER.png

Location of code:

Singleton.py

Interpreter.py

Lines 37,137, 183-198, 248, 280, 311, 331, 589-596

The reasons why I have applied this pattern:

Lots of the classes of my interpreter are only meant to have one instance for each class. When testing the program, if new instances are set up, you cannot access any methods that have been loaded by the init process, or through the view. Having access to the actual classes that are used in the program means that i am not having to create new instances of the class for testing purposes. It also is a better way of managing the connections between classes in the program (for instance, when the controller creates an instance of the model class)

I have already in a way set up a singleton between the views and the controller. I will be able to revise names to make it obvious that I have used the singleton pattern.

The pattern also means that i can follow init scripts to determine when the new classes are created themselves. The way the pattern works, it must be accessed to give access to the class.

Class Diagram after the modification:

pattern1\classes\_SINGLETON.png