Global variables with imports

Sharing global variables between modules

If for whatever reason you need to share global variables you need to keep in mind that global variables in Python are global to a module, not across modules. Which is different than say `static` variable in C.

To make your global variable be able to be shared cross module you would obviously declare a global variable in a module, and the best practice is to declare it in a dedicated module file called `config.py` and have everyone that needs that global variable import it into their scope.

```
# config.py
a = 69 # Variable declared on top level are global by default
```

```
# cool.py
import config
print(config.a)
```

> A word of warning, don't use `from` import unless you want the global variable to be a constant. If you do `from config import a` this would create a new `a` variable that is initialized to whatever `config.a` is at the time of the import, and this new `a` variable would be unaffected by any assignments to `config.a`. So it is not global global anymore.

Revision #1
Created 2023-04-26 22:53:25 UTC by Tamarine
Updated 2023-04-29 00:49:18 UTC by Tamarine