

# Homework 6 - MyPL Interpreter

---

This program will interpret the code that is written in a text file and will actually execute the program

## Steps

---

- Copy all of the given code that Dr. Bowers gave to us into a new file called `mypl_interpreter.py`.
- Create the functions for the visitors.
  - These will be the functions that will help execute the code in the source file.

## Challenges

---

- I think that I underestimated this assignment because it took a lot longer than expected. It was very difficult for me to go from computing with types to computing with values.
- One of my biggest issues that I was running into was that when I would read an int or string into a variable `x`, then printed out a string, and then tried to print the value stored in `x`, the string that was printed would be stored into the variable `x` so that string would be printed twice. It is pretty difficult to explain what was happening without showing it.
  - Eventually I found out that it was an issue with my assignment statement and symbol table. I didn't realize that I needed to add the value to the symbol table again if it didn't exist. This is the case because we already popped all of the environments in the type

checker, which means the symbol table was lost with it.

- I was getting a while loop that would recurse forever, which is bad.
  - To solve this all I needed to do was add an else clause to my if statement in my `visit_while_stmt` method. I would break out of the loop if the flag was no longer true.
- My else clause in an if, elseif, else block would be executed no matter what.
  - To solve this I added another flag called `has_executed`. If neither the if part nor any of the elseif parts had been executed, then the `has_executed` flag will cause the else block to execute. If not, then one of the other if statements executed so the else should not be.
- Finally, I had trouble with a negation of an if statement.
  - To solve this I had to have a large if else block. The first half says that a boolean expression is negated, and has all of the truth values for if the boolean expression is. In the else block, it has the same thing but all the truth values are reversed.

## Tests (all done without lists)

---

- I used the tests found on Piazza
- I used the original example MyPL code that was given to us in the first assignment.