Python Programming

Python Debugging and Intermediate Variables: Practice Problem + Reading Quiz

In this homework, you're going to write code for a challenge problem and answer some reading comprehension questions.

You will practice these programming concepts we've covered in class:

- Type conversion
- Escape characters and string formatting
- Debugging techniques
- Variable scope

Deliverables

Part of this homework will be code challenges and part will be reading with comprehension questions.

For the reading quiz, make a text file called `answers.txt` and use it to compile your answers to the numbered questions.

For each of the code challenges listed below, you will create a new `.py` file and write code to solve the problem. For example, you would create `problem1.py` with your solution code to the first problem. Run the file from the command line to check your work.

Reminder: On your laptop, you can run the file from your command line with the following:

```
python problem1.py
```

> **Hint:** Make sure you are printing something out with the `print` statement. Otherwise, you won't see any output from running your program!

Requirements:

By the end of this, you should have: * One `.py` file for the code challenge. * One text file with answers to the five reading comprehension questions.

Code Challenge

Problem 1: We're in a Good Place!

Skill you're practicing: Debugging techniques and variable scope.

Jason is a huge Jacksonville Jaguars fan. The team isn't doing great now, but he has faith: "All we need is a defense, and an offense, and some rule changes!"

Starter Code

```
offense = False
defense = False
rule_changes = False

def get_offense():
  offense = True

def get_defense():
  defense = True

  def get_rule_changes():
    rule_changes = True

  if offense and defense:
    get_rule_changes()

get_offense()
get_defense()

print("How are the Jags doing?\n")
print("We have offense:", offense)
print("We have defense:", defense)
print("We have some rule changes:", rule_changes)

if offense and defense and rule_changes:
  print("We're going to the Super Bowl!")
else:
  print("I can't predict the future, but no, the Jaguars will never win the Super Bowl.")
```

Expected Output

```
How are the Jags doing?

We have offense: True
We have defense: True
We have some rule changes: True
We're going to the Super Bowl!
```

Actual Output

```
How are the Jags doing?

We have offense: False
We have defense: False
We have some rule changes: False
I can't predict the future, but no, the Jaguars will never win the Super Bowl.
```

If you want to run the code in a repl.it, the code is also written here.

> **Hint:** Include a bunch of `print` statements everywhere to print out the values of the variables at various times. For example, inside `get_offense()`, put a `print` statement like `print("offense is", offense)`.

Reading Material

Read through the examples in this Data Camp article about data types and type conversion. Then, answer the following questions.

1. *Coercion* is another term for which of the following concepts in Python?

- a) Encapsulation
- b) Inheritance
- c) Explicit type conversion
- d) Implicit type conversion
- e) Floor division

2. *Type casting* is another term for which of the following concepts in Python?

- a) Encapsulation
- b) Inheritance
- c) Explicit type conversion
- d) Implicit type conversion
- e) Floor division

3. What function in Python can we use to check a variable's type?

- a) `type()`
- b) `typeof()`
- c) `typeof`, but it is an operator not a function
- d) `get_type()`

4. Which of the following are NOT primitive data structures?

- a) Float
- b) Integer
- c) List
- d) String
- e) Boolean

5. According to the article, what is the main reason to convert a tuple into a list?

See Ya Later!

You're all done. Bye now!