

ASSIGNMENT 8

Python Decisions

All submissions must include the functions as defined and anything outside of those functions in a `main()` with the 'python incantation' at the bottom of your submission.

EXERCISE 1

Write a Python function called `the_difference` that:

- Accepts two parameters.
- Ensures that they are integers and **returns** `None` if either is not an integer.
- Using ONLY `if`, `elif`, or `else` statements
- Subtracts the smaller number from the larger number and **returns** the result
- You can use `abs()` but that is the only function allowed. **** DO NOT USE ANY OTHER FUNCTIONS IN YOUR FUNCTION **** - I'll just flat out assume you're using an auto-coder if you do. No mercy. Gloves are off.

Example function output:

```
the_difference(13,28)
15
```

```
the_difference(123,45)
78
```

```
the_difference(52,52)
0
```

```
the_difference(-45,12)
57
```

```
the_difference(-67,-112)
45
```

```
the_difference(5, 'Chuck Norris')  
None
```

Hints:

- Use an `if..else` statement to determine which number is largest and which number is the smallest.

GRADESCOPE SUBMISSION

A Python script named `A8E1.py`

EXERCISE 2

Write a Python script that plays the rock-paper-scissors game, where rock beats scissors, scissors beats paper, and paper beats rock. It will be a one-player game against the computer.

The script must first ask the player to choose rock, paper, or scissors, by entering the first letter of its name (i.e., R, P, or S). Upper- and lower-case letters must be accepted. Invalid choices must also be accepted but will result in the player losing the game.

The script must randomly select the computer choice of rock, paper, or scissors. The computer cannot make an invalid choice. The script must determine the game result as win, lose, or draw for the player.

The script must print the player's choice, computer's choice, and the game result exactly as shown in the example script output below.

Script Structure

`get_player_choice()`

- Accepts no parameters
- Asks the user select R, P or S to indicate Rock, Paper, or Scissors
- Prints the user choice in lowercase similar to the examples below (*Doesn't have to be exact, anymore!*)
- **Returns the full name of the user choice in uppercase (even if it is invalid)**

`get_computer_choice()`

- Accepts no parameters
- Randomly selects rock, paper, or scissors for the computer
- Prints the computer choice similar to the examples below (*Doesn't have to be exact, anymore!*)
- **Returns the computer choice in uppercase**

`game_result()`

- Accepts two parameters: user choice, computer choice
- Determines the game result as win, lose, or draw for the player
- Prints the game result similar to the examples below (*Doesn't have to be exact, anymore!*)
- **Returns game result as an integer:**
win = 1, lose = -1, draw = 0

Example script output:

```
PS C:\> python A8E2.py
(R)ock, (P)aper, or (S)cissors? R
You chose rock.
The computer chose rock.
It's a draw.
```

```
PS C:\> python A8E2.py
(R)ock, (P)aper, or (S)cissors? s
You chose scissors.
The computer chose paper.
You win!
```

```
PS C:\> python A8E2.py
(R)ock, (P)aper, or (S)cissors? p
You chose paper.
The computer chose scissors.
You lose.
```

```
PS C:\> python A8E2.py
(R)ock, (P)aper, or (S)cissors? masala
You chose an invalid option. The computer chose paper.
You lose.
```

Hints:

- Use the `choice()` function from the `random` module to select the computer choice.
- Use the `upper()` method of the `str` class to convert the user choice to uppercase.
- Check out the coin toss game example in the lecture slides. The script for this exercise will be similar, but instead of two choices (i.e., heads and tails) there are three choices (i.e., rock, paper and scissors), so there are more combinations that must be considered in the `if...elif...else` statements.

GRADESCOPE SUBMISSION

A Python script named `A8E2.py`