1. Write the code to perform the following (a few lines at most):
   - Print *Performing Mind Meld* to the screen 10 times.
   - Create a tuple called `unlikely_pets`, initializing it with the strings: “slug”, “smilodon”, “honey badger” and “tape worm”.
   - Generate a random integer from 0 through 20.
   - Initiate a loop that will cycle through the values 1200, 2400, 3300, 4700.
   - Given a list called `power` that is filled with floating point values, determine the average of the largest value and the smallest value.
   - Given the list above, print out the second, third and fourth values.

2. Write the output of the `do nothing useful` program, below.

   ---------------Output--------------
   # Skeezix Sneezes!
   x = 3
   y = 1
   print("\n\tReady!\n")
   for c in range(3):
       z = x
       while z < 5:
           z = z + y
           print(c, z, y)
           print("Set!")
   print("Go!")

Answers follow.
Perform Mind Meld to the screen 10 times.

# Assuming we want each on their own line, this is quick
print("Performing Mind Meld\n"*10)

# This will also work
for i in range(10):
    print("Performing Mind Meld")

# as will this (least desirable)
i=0
while i < 10:
    print("Performing Mind Meld")
i+=1

- Create a tuple called unlikely_pets, initializing it with the strings: “slug”, “smilodon”, “honey badger” and “tape worm”.

  # () yields a tuple, [] yields a list
unlikely_pets=(“slug”,“smilodon”,“honey badger”,“tape worm”)

- Generate a random integer from 0 through 20.

  # Make sure you import random first
r = random.randrange(20)

  # This works but its the long way home:
r = int(20*random.random())

- Initiate a loop that will cycle through the values 1200, 2400, 3300, 4700.

  for x in 1200,2400,3300,4700:
      # looped code goes here

  # If you need to use these values in another context you could
  # put them in a list and then access the list:

  y=[1200,2400,3300,4700]
  for x in y:
      # looped code goes here
- Given a list called `power` that is filled with floating point values, determine the average of the largest value and the smallest value.

```python
# There are a few ways to determine the max and min. The most straightforward is to sort it. The min will be at the front and the max will be at the back end.
power.sort()
pmin=power[0]
pmax=power[len(power)-1] # a shortcut is pmax=power[-1]
pavg=(pmin+pmax)/2

# or you could put it altogether like so
power.sort()
pavg=(power[0]+power[-1])/2
```

# If we had a tuple, we couldn’t use the member function `.sort()`. Instead, we’d make a sorted copy using the sorted() function as in `powercopy=sorted(power)` and use the copy.

# Sorting large sequences can be time consuming so an alternate method is to search the list for min and max values, like we did in the maximum power theorem lab exercise.

```python
pmin=pmax=power[0] #initialize to first value in list
for p in power:
    if p < pmin:
        pmin=p
    elif p > pmax:
        pmax=p
pavg=(pmin+pmax)/2
```

- Given the list above, print out the second, third and fourth values.

```python
# Remember, Python starts counting at 0 not 1
print( power[1:4] )

# If you don’t want to print them out as a sub-list but prefer to see them individually, loop through them for i in range(1,4):
    print( power[i] )
```
2. Write the output of the do nothing useful program, below.

```python
# Skeezeix Sneezes!
x = 3
y = 1
print("\n\tReady!\n")
for c in range(3):
    z = x
    while z < 5:
        z = z + y
        print(c, z, y)
    print("Set!")
print("Go!")
```

```
-----------Output--------------
|                               |
|  Ready!                        |
| 0 4 1                         |
| 0 5 1                         |
| Set!                          |
| 1 4 1                         |
| 1 5 1                         |
| Set!                          |
| 2 4 1                         |
| 2 5 1                         |
```

```
|                               |
| Go!                           |
|                               |
```