

day18 try/except  
Tuesday 9/20, Wednesday 9/21/22

Today's program has four parts. Please first name your program "day18 try/except"

Start with a while True: loop, as we're going to put this entire program into a "run again?" loop.

Part 1:

Ask the user to enter an integer. Use a try/except structure to make sure that the program does not crash if the user enters a decimal number or text instead. Give a clear error message if the user does not enter an integer (for example "That was not an integer, please try again.")

Once the user enters the integer, after the while True: try/except loop, use a for loop to print the numbers from 1 to the integer, with a space between each number, all on a single line like this:

```
1 2 3 4 5 6 7 8 9
```

You do this using the `print(x, end = " ")` call that we learned on Monday. Go look it up in the slide deck.

Part 2:

Print a blank line.

Ask the user to enter an integer between 10 and 20 inclusive. Like with the first part, give a clear message if the user does not enter an integer. Also, include appropriate messages if the user enters an integer that is smaller than 10 or larger than 20. After the user succeeds in entering a good number, leave the while True: try/except loop and print "Thanks" as many times as the number they entered, one per each line like this:

```
Thanks  
Thanks  
Thanks  
Thanks  
Thanks  
Thanks  
Thanks  
Thanks  
Thanks  
Thanks  
Thanks
```

You can do this with a simple for loop (`for x in range(num):`). We learned this Monday. Go look it up.

Part 3:

Print a blank line.

Write a program that asks the user for two integers, where the second is larger than the first, then prints all the numbers from the first to the second in three columns. The input calls that ask for the integers must each be separately protected by its own try/except structure and while True: loop. If the user enters a valid first number, for example, and then enters something other than an integer for the second number, they are not forced to re-enter the first number.

#### Part 4:

Finally, ask if the user wants to run again, make it so if they hit Enter without typing anything that the program runs again (the entire program, all three parts.) You can do this by checking if the length of what they entered is greater than zero or by using the != not equals sign with empty quotes.)

#### Sample run:

```
Part 1:
Please enter an integer: hi
That was not an integer, please try again.
Please enter an integer: 34.3
That was not an integer, please try again.
Please enter an integer:
That was not an integer, please try again.
Please enter an integer: 9
1 2 3 4 5 6 7 8 9

Part 2:
Please enter an integer from 10 to 20:
That was not an integer, please try again.
Please enter an integer from 10 to 20: 1
Too small, please try again.
Please enter an integer from 10 to 20: 111
Too big, please try again.
Please enter an integer from 10 to 20: 14
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks
Thanks

Part 3:
Please enter an integer: 3
Please enter a larger integer:
That was not an integer, please try again.
Please enter a larger integer: 1
Too small, it needs to be larger than 3, please try again.
Please enter a larger integer: 28
3 4 5
6 7 8
9 10 11
12 13 14
15 16 17
18 19 20
21 22 23
24 25 26
27 28

Hit Enter to run again, anything else to quit: bye
```