Computer Science 1 (../index.html) CPSC 1301K (../index.html)

Project 4: Multiplication Table Generator

< Previous (project3.html) Next > (project5.html)

Due Date

See the calendar (../calendar.html#currentWeek) for due date.

Objectives:

- Use functions
- Use the Math module

Description

This project provides a multiplication table (with column and row headers) based on the user's specifications.

Requirements

Write a Python 3 script that has a `main()` function.
Additionally, have a function that returns valid user input. If the user enters something that is not just digits, then display:

```
Please enter only digits 0-9!
```

Limit the input to at most 20 rows and 10 columns. If the user enters a number that is too large, then display:

```
Please enter only numbers between 1 and 20!
```

for rows and

```
Please enter only numbers between 1 and 10!
```

for columns.
Have another function that displays the multiplication table (based on a number of rows and columns). Separate the numbers in the multiplication table with the TAB character (`\t`).
Include a descriptive comment before each function and each major section of your code. Describe in English (not code) what the function or section does. Be sure to include any assumptions. Write it for another software developer to read, meaning one that already knows Python.

Example A (user input in **bold face blue**, required elements for autograded are underlined)

```
Welcome to Multiplication Table Generator!

Please enter the number of rows: ninety
You entered ninety
Please enter only digits 0-9!
Please enter the number of rows: -10
You entered -10
Please enter only digits 0-9!
Please enter the number of rows: 9999
You entered 9999
Please enter only numbers between 1 and 20!
Please enter the number of rows: 10
You entered 10
Please enter the number of columns: 7777
You entered 7777
Please enter only numbers between 1 and 10!
Please enter the number of columns: 777
You entered 777
Please enter only numbers between 1 and 10!
Please enter the number of columns: -3
You entered -3
Please enter only digits 0-9!
Please enter the number of columns: -2
You entered -2
Please enter only digits 0-9!
Please enter the number of columns: -1
You entered -1
Please enter only digits 0-9!
Please enter the number of columns: three
You entered three
Please enter only digits 0-9!
Please enter the number of columns: two
You entered two
Please enter only digits 0-9!
Please enter the number of columns: one
You entered one
Please enter only digits 0-9!
Please enter the number of columns: zero?
You entered zero?
Please enter only digits 0-9!
Please enter the number of columns: 5
You entered 5
Multiplation table (10 x 5):
        1       2       3       4       5
1       1       2       3       4       5
2       2       4       6       8       10
3       3       6       9       12      15
4       4       8       12      16      20
5       5       10      15      20      25
6       6       12      18      24      30
7       7       14      21      28      35
8       8       16      24      32      40
9       9       18      27      36      45
10      10      20      30      40      50
Good-bye
```

Example B (user input in **bold face blue**, required elements for autograded are underlined)

```
Welcome to Multiplication Table Generator!

Please enter the number of rows: 15
You entered 15
Please enter the number of columns: 10
You entered 10
Multiplation table (15 x 10):
        1       2       3       4       5       6       7       8       9       10
1       1       2       3       4       5       6       7       8       9       10
2       2       4       6       8       10      12      14      16      18      20
3       3       6       9       12      15      18      21      24      27      30
4       4       8       12      16      20      24      28      32      36      40
5       5       10      15      20      25      30      35      40      45      50
6       6       12      18      24      30      36      42      48      54      60
7       7       14      21      28      35      42      49      56      63      70
8       8       16      24      32      40      48      56      64      72      80
9       9       18      27      36      45      54      63      72      81      90
10      10      20      30      40      50      60      70      80      90      100
11      11      22      33      44      55      66      77      88      99      110
12      12      24      36      48      60      72      84      96      108     120
13      13      26      39      52      65      78      91      104     117     130
14      14      28      42      56      70      84      98      112     126     140
15      15      30      45      60      75      90      105     120     135     150
Good-bye
```

Submission

Submit your `project4.py` and your `rubric-project4.txt` to the appropriate Assignment tab/folder in CougarVIEW (https://cougarview.columbusstate.edu). Optionally, you can submit your project to codePost.io (https://codePost.io).

Rubric:

```
Points       Item
----------   ----------------------------------------------------------------
_____ / 10   Comments before each function and major section of code (written for another software developer; not too many com
ments and not too few)
_____ / 25   Input validation
_____ / 25   Multiplication table
_____ / 10   Tab-delimited output
_____ /  2   Completed rubric (estimates for each line including hours spent)

_____ / 72   Total


_____  Approximate number of hours spent

I, (REPLACE WITH YOUR FULL NAME), affirm that the code that I submitted is my own work and that I did not receive help that wa
s not authorized by the instructor.
```

Notes

1. Before writing any code, write the major steps as comments. Then, iteratively go through and implement each step. For example, start with welcoming the user. Verify that it works. Then, work on the next step.
2. To get the numbers to match exactly, only round the values when they are displayed (and don't store the rounded value).

Optional

1. After requesting the number of rows and columns, then requests the starting row number and the ending column number (and make sure that there's a valid range for each one).

Hints

1. Each time you do input validation, it requires a loop. Enter the input shown in the examples and make sure you get the same output.

 Last Modified: 05/06/2021 09:24:33