Exercises with loops

In this lab you will write a bunch of short functions using `while` loops and `for` loops. I've provided the function headers and parameter lists in the accompanying code file. Do not change these! For now the body of the function consists of just a single return or print statement---you will replace these with the appropriate function code. None of the solutions requires more than 6 lines of code, usually less.

You can test your solutions by choosing the Run Module menu option and then typing the name of the function in the shell together with any required arguments. So, for example, typing

```python
cuberoots1()
```

should display the table of cube roots

1. Write a function that prints the cube roots of the integers from 1 to 20, with one number on each line.
2. Write a function that prints the cube roots of the integers from 1 to 20. Now the output should have two columns, with the cube roots in the right hand column and the integers 1 through 20 themselves in the left-hand column. You should use a `for` statement iterating through a range for this instead of a `while` statement.
3. Write a function that carries on a dialogue with the user: It will repeatedly prompt for input until the user enters something whose first character is either lower- or upper-case A. The function should behave properly if the input is the empty string.
4. Write a function that carries on a dialogue with the user: It will repeatedly prompt to enter a number of the word `done`. When `done` is entered, the program will print the largest value that the user enters. (Do not worry about trapping errors like an entry that is neither a number nor `done`.) (This problem is a bit tricky: You have to always keep track of the largest value that has been entered so far, and update it as needed.)
5. Write a function that returns the number of occurrences of a lower-case letter 'a' in the argument.
6. Write a function that uses a `for` statement iterating over a range to return the value

\[ 1 + \frac{1}{2} + \cdots + \frac{1}{100}. \]

7. Write a function that with one parameter \( n \) that returns the product
\[ n! = 1 \times 2 \times \cdots \times n. \]

8. Write a function using `for` iterating over a string, that takes a string \( s \) as an argument and returns a string in which every vowel in \( s \) is replaced by the symbol `#`. For example, if the argument is the string

```
Kansas City won the Super Bowl this year.
```

then the function will return the string

```
K#ns#s C#ty w#n th# S#p#r B#wl th#s y##r.
```