1. Write a loop that makes seven calls to console.log to output the following triangle:

```
#
##
###
####
#####
######
#######
```

2. Write a program that uses console.log to print all the numbers from 1 to 100, with two exceptions. For numbers divisible by 3, print "Fizz" instead of the number, and for numbers divisible by 5 (and not 3), print "Buzz" instead. When you have that working, modify your program to print "FizzBuzz", for numbers that are divisible by both 3 and 5 (and still print "Fizz" or "Buzz"for numbers divisible by only one of those).
3. Write a program that creates a string that represents an 8×8 grid, using newline characters to separate lines. At each position of the grid there is either a space or a “#” character. The characters should form a chess board.

   Passing this string to console.log should show something like this:

```
 # # # #
# # # #
 # # # #
# # # #
 # # # #
# # # #
 # # # #
# # # #
```

   When you have a program that generates this pattern, define a variable size = 8 and change the program so that it works for any size, outputting a grid of the given width and height.

4. Write a function min that takes two arguments and returns their minimum.

5. You can get the Nth character, or letter, from a string by writing"string".charAt(N), similar to how you get its length with "s".length. The returned value will be a string containing only one character (for example,"b"). The first character has position zero, which causes the last one to be found at position string.length - 1. In other words, a two-character string has length 2, and its characters have positions 0 and 1.

   Write a function countBs that takes a string as its only argument and returns a number that indicates how many uppercase “B” characters are in the string.

   Next, write a function called countChar that behaves like countBs, except it takes a second argument that indicates the character that is to be counted (rather than counting only uppercase “B” characters). Rewrite countBs to make use of this new function.