**Lab11: String Manipulation using Loops**

Create a number of methods inside one class: StringManipulation

* 1. **Method: firstHalf**

Given a string of even length, return the first half. So the string "WooHoo" yields "Woo". Additional examples:
firstHalf("WooHoo") → "Woo"
firstHalf("HelloThere") → "Hello"
firstHalf("abcdef") → "abc"

**1.2 Method: endsLy**

Given a string, return true if it ends in "ly". Examples:
endsLy("oddly") → true
endsLy("y") → false
endsLy("oddy") → false

 **1.3 conCat**

Given two strings, append them together (known as "concatenation") and return the result. However, if the concatenation creates a double-char, then omit one of the chars, so "abc" and "cat" yields "abcat".
conCat("abc", "cat") → "abcat"
conCat("dog", "cat") → "dogcat"

**1.4 DoubleChar**

Given a string, return a string where for every char in the original, there are two chars.

doubleChar("The") → "TThhee"
doubleChar("AAbb") → "AAAAbbbb"
doubleChar("Hi-There") → "HHii--TThheerree"

**1.5 The Magic 8-Ball (today or Friday if you run out of time, however continue work on this at home)**

The [Magic 8-Ball](https://en.wikipedia.org/wiki/Magic_8-Ball) is a toy produced by Tyco Toys (now Mattel) and consists of a ball filled with a blue fluid. Suspended in the fluid is a icosahedron (a 20-sided polyhedron with each side consisting of an equilateral triangle). Each face has an answer to a yes/no question. [Check out this link](http://bgr.com/2015/02/26/whats-inside-a-magic-8-ball/) to see what a Magic 8-Ball looks like on the inside!

1. Get the EightBall.java program from Moodle or from the server
2. Rather than exit the program if the question is longer than 60 characters, read in the question again. Continue checking for the length until the user enters a question less than or equal to 60 characters.
3. Recall that the nextLine method can read in an empty string. If the question is empty (zero length), you should tell the user that's not allowed and then prompt for and read in the question again. Do this until the user enters a valid string.
4. A question should end with the '?' character. Check the question string and make sure that it ends with a question mark.
5. Add a new loop (surrounding most of the original code) that will execute the program until the user says they no longer wish to play. Note you are already prompting for and receiving a yes/no response to the question, "Do you want to ask a question (yes/no)?"

Your final version should allow you to keep asking questions until the user would say no, and it should verify that each question is at most 60 chars ending with a question mark, also checking for zero length questions.

You could achieve the previous features just by adding only ONE simple loop (properly written and properly editing the existing code).