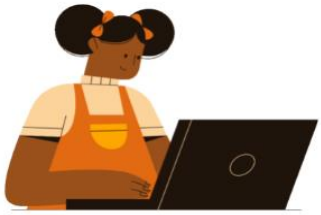

Lab 01: Basic Methods

In this lab you will be writing methods from scratch. There will be test cases. So make sure your method names are correct.

Exercises

In this lab, you'll be writing the method headers for EACH problem. Don't forget the access modifiers "`public static`". Also, for the test cases to work, you MUST use the correct method name and parameters.

1) Create a `getString()` method that returns whatever the user typed. Basically, it's your Scanner. It needs to accept 1 String parameter. This parameter is your user prompt, so print it before scanning the next line. Return whatever the user typed.

2) Create a `diagonal()` method that returns nothing. It needs a single String parameter. The method should print that parameter like you see below. Hint: You many need nested loops for this (1 for spaces and 1 for the character).

   Now go to the **main()** method and find `//FOR #2`. Under this, use your scanner method to ask the user for a single word. Then call `diagonal()` and send it the user's input.

   What the run should look like.

```
Enter a single word: <Howdy!>
H
 o
  w
   d
    y
     !
```

   (Note: there is a return after this last character)

3) Create a `diagonals()` method that also returns nothing. Ask the user for a sentence or phrase. Then print each word in their input like you did in `diagonal()`. Do NOT recreate `diagonal()`. Rather call it for each word in the phrase.

Continue…

Exercises

You can break the input at a space and store it in an array like this. Note, `userInput` is a String variable.

```
String[] words = userInput.split(" ");
```

Example:

```
Enter a sentence or Phrase: <Howdy Y'all!>
H
 o
  w
   d
    y
Y
 '
  a
   l
    l
     !
```

4) Create `letterToBinary()` method. It takes a single char as a parameter. It converts that char's integer form to a binary String. For example: if `'a'` is sent, that's a 97 which is "`01100001`" in binary. The return should be a string that's 8 characters long.

You can store a `char` in an `int` like this
(assume `x` is an `int` and `letter` is a `char`)

```
x = Integer.valueOf(letter);
```

5) Create a `sentenceToBinary()` method. It should return a binary String representation of a user's input. Start by asking the user for a sentence. DO NOT recreate your scanner. Then use `letterToBinary()` to convert each character in the user's input. Do NOT add any spaces or other characters between the binary numbers. Return the resultatnt String.

Example:

```
Enter a word or phrase: <Howdy Y'all!>
```

Return:

```
0100100001101111011101110110010001111001001000000101100
1001001110110000101101100011011000010000 1
```