**Pig Latin**

Open BlueJ, and create a new BlueJ project titled **Lab20-PigLatin** in your CS 1 folder.

Create a new class, open it (double click), delete BlueJ's starter code, and **type** in our code skeleton:

//Name:

****//Period:

import java.util.\*;

public class PigLatin

{

public static void main(String[] args)

{

Scanner console = new Scanner(System.in);

}

}

Remember, Strings are ***immutable*** (they can't be changed once created). To get around this, there are built-in methods in the String class we can use that will perform some common operations. More information can be found on our [website](bit.ly/mrbunnCS).

**Before each problem, insert a COMMENT with the problem number.**

1. Get the value of a String variable str from the keyboard. Print a new version of the String with the first letter capitalized.
2. Get a new value of str from the keyboard. Convert the String to all lower case, then print a new version of the String with the last two letters capitalized.
3. Get a new value of str from the keyboard. **Using the** substring() **method(s),** write the code to print the number of times the letter "e" occurs in the String.

//sounds like a job for a loop – check the powerpoints for more info

1. Get a new value of str from the keyboard. Write the code to print the number of times "hi" occurs in the String.
2. Get a new value of strand a new String target from the keyboard. Write the code to print "Found it!" if target appears *anywhere* in str, or print "Nope!" if it doesn't.
3. Get a new value of strfrom the keyboard, and a value for an integer n. Write the code to print the last n letters of str, repeated nnumber of times. Example (**user input shown in red**):

Enter a String >>> **Hello**

Enter an integer >>> **3**

llollollo

1. (Don't forget to add a console.nextLine() before this problem – you're getting a String after an int.) Ask for the user's email. Print out the 'domain' of the email address (what comes after the @ character). Use the indexOf() method to find the location of the @ character.

//given "some\_name@website.com", your code should print "website.com"

1. Ask for the user's full name*.*  Print the name with the first name as entered, and the second name all in capital letters. You can assume that there will be two names, and that they are separated by a space character.

//use the indexOf() method to find the location (index) of the space character.

1. (Riddle) A man, desperate for a drink of water, dies of thirst in his own home - yet there is no problem with the plumbing. How is this possible?
2. Get a new value for str*,* and print out a what strwould look like with all the \* characters "removed". For example, if the user entered \*\*abc\*\*, your code would print abc.
   1. Check the powerpoints for more help; a loop and a second "result" String will help.

**Single word Pig Latin app**

Write a program that converts a String containing a single word into the Pig Latin version of that word. The syntax rules of Pig Latin are:

* If the word starts with a consonant (i.e. not a vowel), it moves **all leading consonant(s)** to the end of the word and adds "ay". For example, your program should translate "bring" to "ingbray", "book" to "ookbray", "throw" to "owhthray", etc.
* If the word starts with a vowel (a, e, i, o, u), simply add "hay" to the end of the word. For example, your program should translate "apple" to "applehay".

Knowing the index of the first vowel in the String will be very useful for this lab. Put the entire program in a loop that will allow them to keep converting words until they enter the String "quit".

**(Advanced) Full sentence Pig Latin**

Convert an entire sentence (one word at a time) to Pig Latin. This can be done with some advanced concepts (like arrays and the split()method). You can also use Scanner class methods to parse a String\*.

*\*The Scanner class works on more than just System.in (keyboard input). You can use Scanner objects on Strings too, like this:*

String str = "This is a sentence.";

Scanner chopper = new Scanner(str); //scanning *str* instead of *System.in*

while (chopper.hasNext()) //loops while there's another token (another thing to scan)

{

//splits the String when it finds whitespace (a space character)

System.out.println(chopper.next());

}

**Would print:**

This

is

a

sentence.