Purpose: Practice program development skills, review earlier concepts (loops, strings, etc.)

Summary: See how *Pride and Prejudice* sounds in Pig Latin! You will write a program that can convert any file of text into Pig Latin. Today we will just do some warm-ups.

Details:

**Lab 5, part One — Warm-up Exercises**

**First Warmup.** Log on to the XInteract system and select problem “Lab5_1”. For this problem, you are to provide two assignment statements. The first assignment statement selects the i-th character from a *String* variable named *line* and stores it in a *String* variable named *letter*. The second assignment statement finds the location of *letter* in the *String* variable *alphabet* (which simply contains the twenty-six letters of the English alphabet) and stores this result in an *int* variable named *position*. Here is the program (the box indicates where XInteract will insert your two assignment statements):

```java
import java.util.Scanner;
import java.util.Random;

/**
 * Lab 5 warm-up exercise: locate a character within a string; more
 * precisely, determine the position of a character within the alphabet.
 *
 * @author Bob Roos
 */
public class Lab5_1
{
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        // The following string is used for "looking up" letters:
        String alphabet = "abcdefghijklmnopqrstuvwxyz";
        // Read in an entire line, followed by an int value "i". We will
        // examine the i-th character of the line and look up
        // its position in the alphabet string:
        String line = input.nextLine();
        int i = input.nextInt();

        String letter; // the i-th character in the input line
        int position; // the position of 'letter' in variable 'alphabet'

        // Handed out on 28 February 2006
        Handout # 9
```
Second warmup. The second warm-up exercise builds upon the first. To do this exercise, select problem Lab5_2 in XInteract. Now you are to find the location in the alphabet of every character in the input string named line. For instance, if the input line is “Test string” you should print “-1 4 18 19 -1 18 19 17 8 13 6”. (The “-1”s correspond to the uppercase “T” and the blank in the middle of string “Test string” — these does not appear in the alphabet of lowercase letters, so the indexOf method returns a value of -1.) Obviously you will need to write some kind of loop to do this.

Here is the program that XInteract will use to test your answer:

```java
import java.util.Scanner;

/**
 * Lab 5 warm-up exercise: find alphabetic positions of all letters
 * in the input line.
 *
 * @author Bob Roos
 */
public class Lab5_2
{
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        // The following string is used for "looking up" letters:
        String alphabet = "abcdefghijklmnopqrstuvwxyz";

        String line = input.nextLine();

        // XInteract will place your solution here //
        // XInteract will place your solution here //

        System.out.println();
    }
}
```

As of this writing, there is an unexplained error in the test program. You may have a correct program that registers an error. Download the file from the Web page and experiment with it that way.

Third warmup. The third warmup exercise turns things around a little bit. Instead of simply looking up a particular character in a line to find its index in another string, we will identify the first character in the line that appears in another string. Specifically, given a string word, suppose
we want to find the location of the first vowel in word. One way to do it is to define a string named vowels containing "aeiou" and then look at each letter of word until we find one that has a non-negative index in the vowels string.

To do this exercise, select problem Lab5_3 in XInteract. Write the Java statements needed to find the location, in String variable word, of the first letter that appears in a second String variable named vowels. You will need to assign values to three different variables: set the boolean variable foundVowel to true if the word contains a vowel; set the String variable letter equal to the first vowel in word (if one exists), and set the int variable position equal to the location of the first vowel in word. For example, if word is "schnecken" and vowels is "aeiouAEIOU", your program should set the variable foundVowel to true and should set the variable vLocation to 4.

Here is the Java program used by the XInteract program:

```java
import java.util.Scanner;

/**
 * Lab 5 warm-up exercise: find the first vowel in an input string.
 * @author Bob Roos
 */
public class Lab5_3
{
    public static void main(String[] args)
    {
        Scanner input = new Scanner(System.in);
        // The following string is used for "looking up" vowels:
        String vowels = "aeiouAEIOU";

        String word = input.nextLine();
        boolean foundVowel = false;
        int vLocation = 0;

        // XInteract will place your solution here //

        if (foundVowel)
        {
            System.out.println("First vowel is "+ word.charAt(vLocation) + " at position "+ vLocation);
        }
        else
        {
            System.out.println("No vowels found");
        }
    }
}
```

HANDED OUT ON 28 FEBRUARY 2006