



This lab serves as a recap and reinforcement of concepts covered till now: like program modularity, String methods, Loops, ...

Exercise 1. Numbers Input

Write a program that keeps on taking numbers from the user, until the user enters any input that it is not a number. It will then stop reading input and calculate the summation and the average of the entered numbers.

Exercise 2. Numbers Input until Quit

Write a program that keeps on taking numbers from the user, until the user enters the word “quit”. Note that, if the user enters any text other than “quit”, it should skip it and continue taking other inputs. Some sample run:

```
Enter sequence of numbers ('quit' to stop):
5 12
three 3
4
-10
yes 4 2 quit 1000
Total = 20
```

```
Enter sequence of numbers ('quit' to stop):
5
10
20
twenty
40
quit
Total = 75
```

Exercise 3. Nested Loops Review

Write a method **triangle** that takes an integer **n** and draws the following **n x n** triangular pattern. You are allowed to use only two for loops (nested loop). Implement a main method to test your method on different user input.

```
* * * * *
○ * * * *
○ ○ * * *
○ ○ ○ * *
○ ○ ○ ○ *
○ ○ ○ ○ ○
```

Exercise 4. Loops Review

Write a program `Largest.java` that asks the user to enter an integer `N` (greater than zero), and then your program should ask the user to enter `N` integers. You are to print the largest integer, and count its occurrences. Example:

```
Enter a positive number: 4
Enter number 1: 2
Enter number 2: 3
Enter number 3: 2
Enter number 4: 3
The largest number is 3. The occurrence count of it is 2.
```

Exercise 5. Caesar Cipher

A simple way to encrypt information is called a Caesar cipher. In a Caesar cipher, a number between 0 and 25 is selected as the key for the message. Each letter is then shifted **circularly** by the selected key. For example, if the key is 1 and the original letter is 'A', then the encrypted letter will be 'B'. If the key is 3 and the original letter is 'b', then the encrypted letter will be 'e'. If the key is 3 and the original letter is 'z', then the encrypted letter will be 'c'.

Write a method, `cipher(String s, int n)`, that returns the encrypted string after shifting all the characters by the corresponding key (`n`). Some examples:

```
System.out.println(cipher("Caesar", 4));    → Geiwev
System.out.println(cipher("West OR East", 5) → Bjxy TW Jfxy
```

Exercise 6. Fermat Primes

In a letter dated December 25, 1640, the great mathematician Pierre de Fermat wrote that he just proved that an odd prime p is expressible as $p = a^2 + b^2$ if and only if p is expressible as $p = 4c + 1$. One hundred years later, Euler proved this theorem. To illustrate, each of the following primes can be expressed as the sum of two squares:

$$5 = 2^2 + 1^2 (= 4 * 1 + 1)$$

$$13 = 3^2 + 2^2 (= 4 * 3 + 1)$$

$$17 = 4^2 + 1^2 (= 4 * 4 + 1)$$

$$41 = 5^2 + 4^2 (= 4 * 10 + 1)$$

whereas, the primes 11, 19, 23, and 31 cannot be expressed as a sum of two squares. Write a JAVA program to display all the primes that can be expressed as sum of squares within a given interval (take the interval as two integers from the user: e.g. 10 100). Implement any needed method to simplify your work.