LAB 7 – ANNUAL FLOWERS (4%)

- Make a new Java project titled "Lab6-Annuals". Create two Java classes – AnnualFlowers, OrderFlowers and use StringUtility class made lately.
- Suppose that four kinds of annual flowers are on sale.
- The kinds of annuals are Marigold, Pansy, Zinnias and Petunia and cost, respectively, $2.30, $1.50, $5.12 and $3.25.
- The program should read the name of a flower and the quantity desired by a customer. The inputs must be entered in the same line and splitted by a comma.
- The program then computes and displays the cost of purchase.
- Here is the sample output:

```
Enter the name of flower followed by quantity
marigold,10

**Price Per Stem**
Marigold  2.30
Pansy     1.50
Zinnias   5.12
Petunia   3.25

***************Bill***************
Marigold costs $23.00 for 10 stems
```

StringUtilityClass:

- This class contains static methods to capitalize the first letter of a parameter and to format full name in a proper case.
- Write another static method that takes a 2D array of Strings as a parameter and return a String.
  - This method prints the values in the array that is passed as an argument to it.
  - This method iterates elements in every row and corresponding columns.
  - As each element is being read in nested loop, append it to the String variable.
  - Return the String variable.

AnnualFlowers class:

- Create a 2D array of Strings that holds the name and price of each flower.
- Here is the sample 2D array:

```
private String[][] flowers = {{"Marigold", "2.30"},
                              {"Pansy",    "1.50"},
                              {"Zinnias",  "5.12"},
                              {"Petunia",  "3.25"}};
```

flowers[rowNum][1]

flowers[rowNum][0]

- Declare a field to represent flower name and price, and getter and setter methods for flower name.

- Make a method which locates the flower in the flowers array and use that index to find the price per stem in the array. You can name this method as getPrice() or searchPrice() and it returns price per stem.
- Override toString() of Object class.
  - Call the static method from StringUtility class and pass flowers array as an argument.

OrderFlowers Class:

- Declare a field to represent the quantity of stems ordered by a customer.
- Aggregate AnnualFlowers class with this class by declaring a variable of AnnualFlowers type.
- In the constructor, add the following statement to initialize AnnualFlowers.
  - yourReferenceVariableName = new AnnualFlowers();
- Make a method to split input taken from the user via keyboard. An input is entered as two values with comma between them.
  - Split the input into separate strings, parse them in appropriate data types and set the flower name and quantity.
  - To set flower name, call setter from AnnualFlowers class.
- Make a method to calculate total cost of purchase.
- Override toString() of Object class.
  - Call and append toString() from AnnualFlowers class.
  - Append total cost of purchase and quantity.

- **Tester class:**
  - Create and initialize an object of OrderFlowers and Scanner class.
  - Prompt the user to enter annual flower name and quantity and split input by calling the appropriate method.
  - Print results by calling toString().

Here is another set of sample output:

```
Enter the name of flower followed by quantity
zinnias,8

**Price Per Stem**
Marigold  2.30
Pansy     1.50
Zinnias   5.12
Petunia   3.25

***************Bill***************
Zinnias costs $40.96 for 8 stems
```