Lab Exercise 1010 – Getting Started

Overview: Considered by many to be a basic professional concept, students might be surprised at how many software developers do not know how to create flowcharts!

Objectives:

- Practice lecture concepts
- Create a flowchart from scratch
- Practice using Python's print(), input(), and basic if / else constructs
- Practice working with string "member functions" (**.isalnum()**, etc.)

Startup File: (none)
Solution File: (none)

Step 01: Requirements & Documentation

Focusing upon the basics, in this session we discovered how we can capture requirements using flowcharts. Designed for the first-time software developer, a little practice to directly review what have covered will help build student confidence!

1. Using a piece of paper:

   a) Create a flowchart with an entry point named "main."

   b) Feel free to refer to the video presentation as required.

2. Your flowchart should document the requirement to:

   a) Input a string in response to an "Enter String" request

   b) Print the string that was entered on the next line

   c) Print the word "punctuation" if a string is not alphanumeric

   d) Print the word "alnum" otherwise

Step 02: Strings & Printing

Basing your implementation upon the video presentation:

1. Use **input()** to prompt for, as well as to accept, a string from the keyboard.

2. Use **isalnum()** to test for alphanumeric characters.

3. Use **if**, **else**, and a pair of associated statement blocks (**:**) to select between the two (2) required responses

4. Run your application (F5 on IDLE / IDLE3) to verify that your program operates properly.

Step 03: Mapping the Results (OPTIONAL)

Verifying that any implementation matches a design specification is often required.

To verify that your code relates back to a designed requirement:

1. Number each graphical step in your design / flowchart

2. Place a single-line comment (#) that relates to each enumerated requirement on your flowchart

3. Add a print() statement to show the flowchart number as your program executes.

4. Run your results several times so as to verify that the program-flow matches your depicted requirements.

(end)