Branch

Exercise 1:

Create a service **branchSwitch1** with a single input String "**testValue**". Write Branch code to write a message based on the value of testValue.

Exercise 2:

Create a service **branchSwitch2** with a single input String "**testValue**". Write Branch code for **null** and **default** values of testValue.

Exercise 3:

Create a service **branchSwitch3** with a single input String "**testValue**". Write Branch code for **null** and **default** values of testValue, and give input as **include empty values for String type** and check flow using **Step** feature.

Exercise 4:

Create a service **branchOnExpression1** with a one input String "**testValue**" . Write Branch code to write a message based on the value of the input fields. The psuedocode for this service is as follows:

a. If **testValue** = "**100**" then write "**notAvaliable**".
b. Else if **testValue** starts with **"1000"** to **"5000"** then write, "**avaliable**" to the server log.
c. Else, write "**comingSoon**" to the server log.

Exercise 5:

Create a service **branchOnExpression2** with a two input String "**quantity**" and "**price"** . Write Branch code to write a message based on the value of the input fields. The psuedocode for this service is as follows:

a. If **quantity** = "**100**" to "**200**" then add **quantity** and **price** values.
b. Else if **price** is "**2000**" then subtract **quantity** and **price** values.
c. Else, write **failure message** to the server log.

Exercise 6:

Test above services **branchSwitch1, branchSwitch2, branchOnExpression1, branchOnExpression2** using the **Step** feature.

Exercise 7:

Create a service **branchOnExpression3** with a one input String "**code"** . Create a list of the conditional steps (target steps) and make them children of the BRANCH step.

a. If **code** start with **PRE0 to PRE6** then wite available to server log.

b. Else, write **not available** to server log.

Exerise 8:

Create a branch service which would return any paragraph in which the word ‘webmethods’ appeared anywhere after the 10th character position of the paragraph.

Exercise 9:

Create a branch service which would return any paragraph in which the word ‘webmethods’ started in any of the first 5 character positions of the paragraph.

Exercise 10:

Create a branch service which would return any paragraph in which the word ‘webmethods’ started in character position 2 through 5 of the paragraph.

Exercise 11:

Create a branch service which would return any paragraph containing the string ‘web’ followed by any single character and the string ‘ethods’. It would match both ‘webMethods’ and ‘webmethods’ on a single expression.

Exercise 12:

Create a branch service which would return any paragraph that contained the word ‘port’,but not paragraphs that contained these characters as part of a larger word, such as ‘import’, ‘support’, ‘ports’ or ‘ported’.

Exercise 13:

Create a branch service which would return any paragraph that contained the characters ‘555-A’ as part of a larger word such as AZ555-A, or Dept555-A, but not ‘555-A’ alone.

Exercise 14:

Create a branch service which would return any paragraph containing the string ‘webMethods’ preceded by a tab character.

Exercise 15:

Create a branch service which would return any paragraph containing a part number that starts with any digit 0 through 9, and is followed by the characters 555-A. Therefore, it would match ‘part 1555-A’ but not ‘part A555-A’ or ‘part #555-A’.

Exercise 16:

Create a branch service which would return any paragraph containing a part number that starts with any character other than 0 through 9, and is followed by the characters 555-A. Therefore, it would match ‘part A555-A’ and ‘part #555-A’, but not ‘part 1555-A’.

Exercise 17:

Create a branch service which would return any paragraph containing a part number that starts with a letter or digit and is followed by the characters 555-A. Therefore, it would match ‘part A555-A’ and ‘part 1555-A’, but not ‘part #555-A’.

Exercise 18:

Create a branch service which would return any paragraph containing a part number that starts with a character other than a letter or digit, and is followed by the characters 555-A. Therefore, it would match ‘part #555-A’ and ‘part -555-A’, but not ‘part 1555-A’ or ‘part A555-A’.