Angular Developer Coding Exercise

Objective

Create a HTML5 page using Angular 7 to serve as a visual tool for creating an “expression.”

Inputs:

A list of JavaScript functions (for the purposes of this exercise you can assume that the list will be complete and available when the page loads and will not change).

Initial Display:

For every function in the list, a unique icon will be displayed on screen, captioned with the function name. Additionally, a blank “expression area” will be present on the screen into which the user can drag and drop the name icons

Initial user interaction:

The user can drag and drop one of the icons into the blank “expression area”. This will trigger additional placeholder areas (a simple <div> with boundary) to appear on the right of the icon. The number of placeholder areas is dependent on the number of arguments that the function represented by the icon is supposed to receive.

Advanced Extension (Extra Credit):

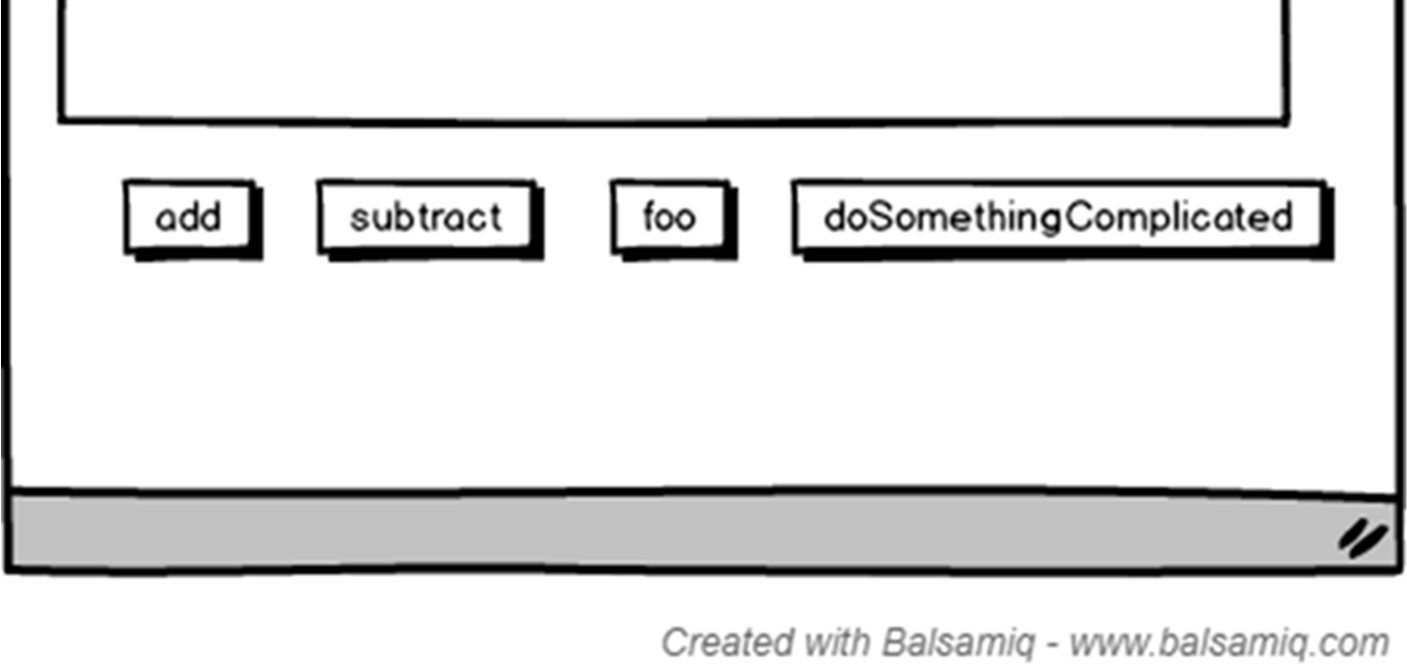
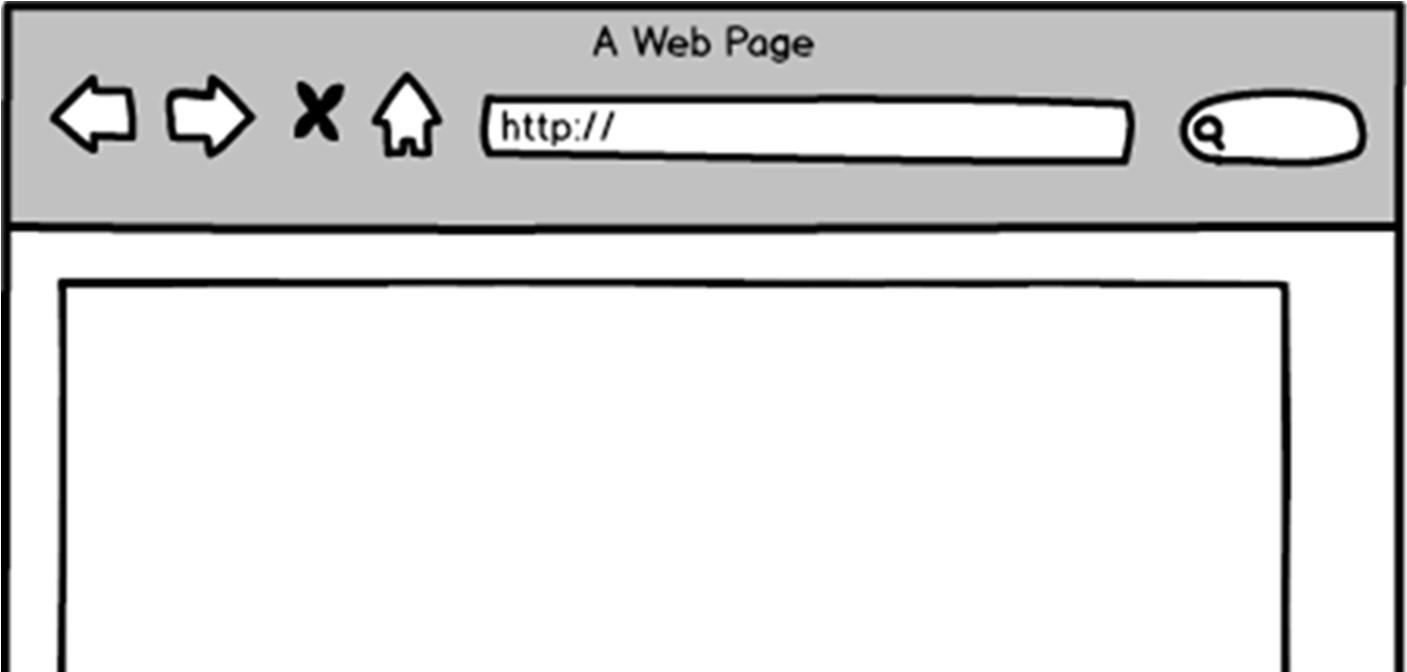
The user can drag more function icons into the new placeholder areas created by dropping of the previous icon. This process can continue recursively indefinitely to create complex chained expressions EXAMPLE

Sample function list provided: [ add, subtract, foo, doSomethingComplicated ]

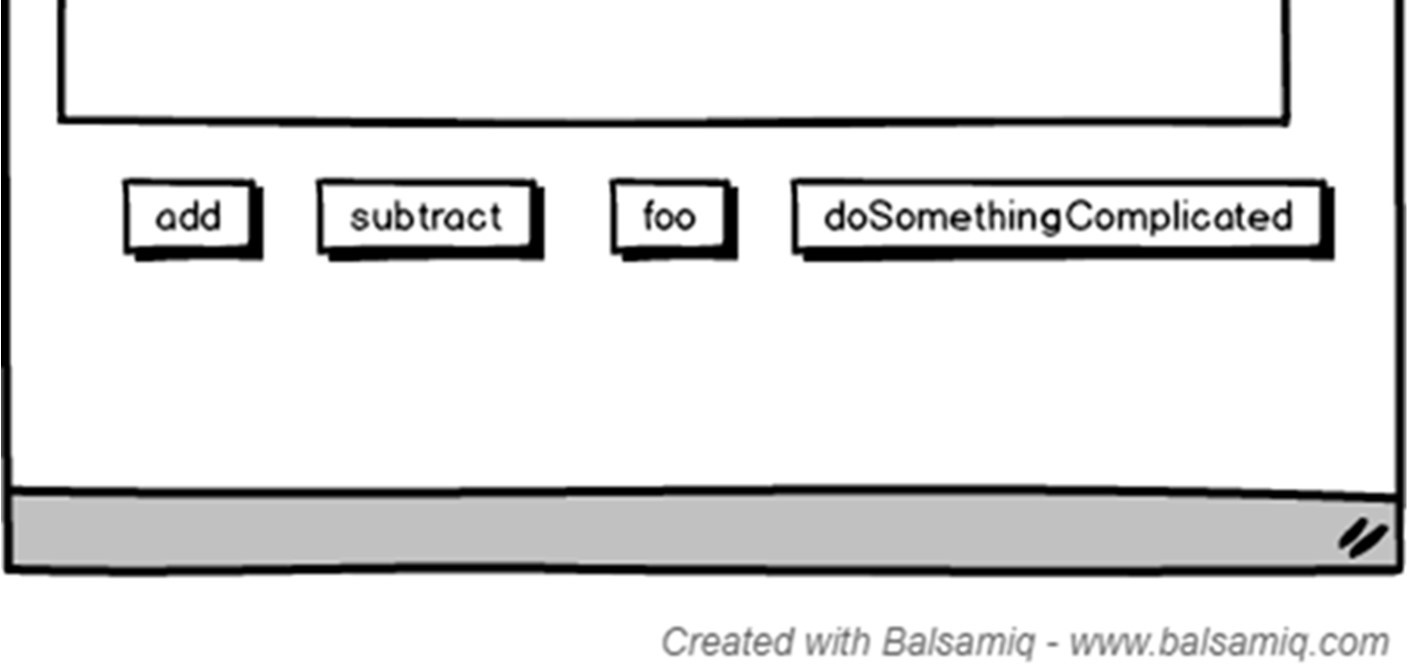
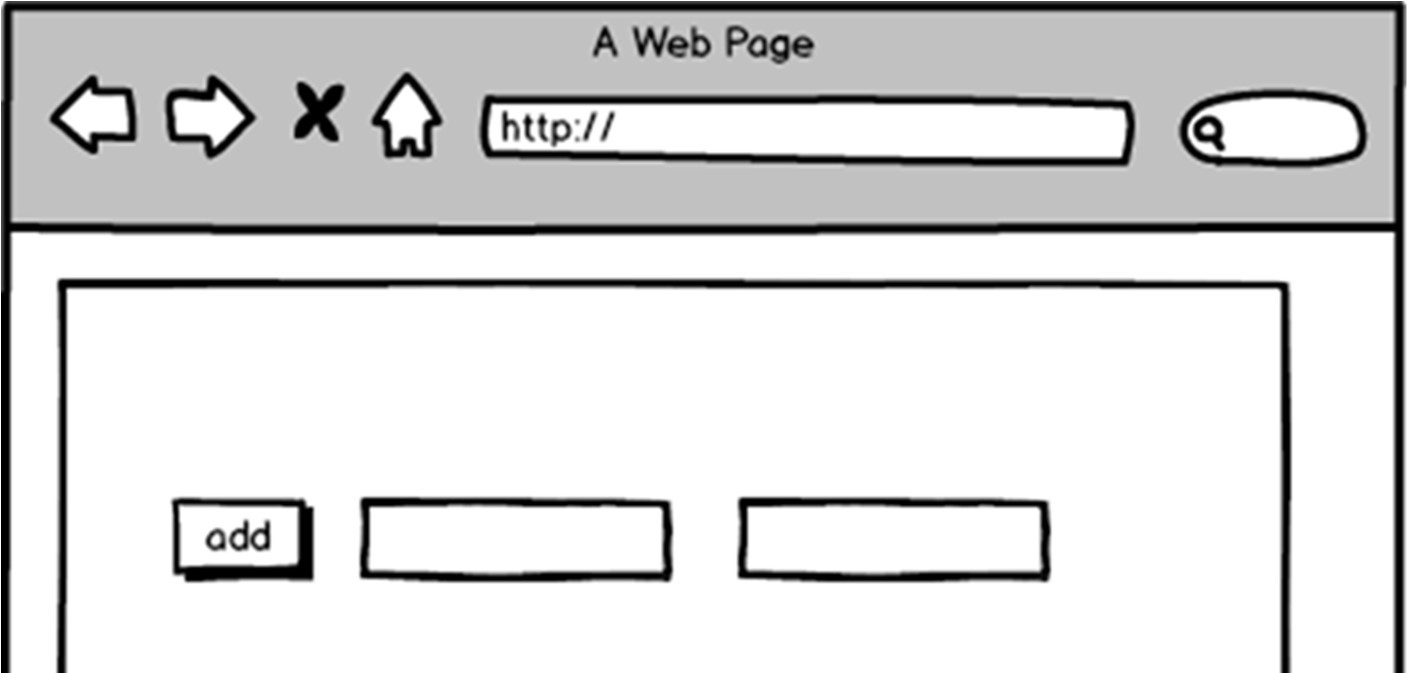
Signatures of the functions function add(x,y) { … } function subtract(x,y) { … } function foo(x,y,z) { … }

function doSomethingComplicated(x) { …}

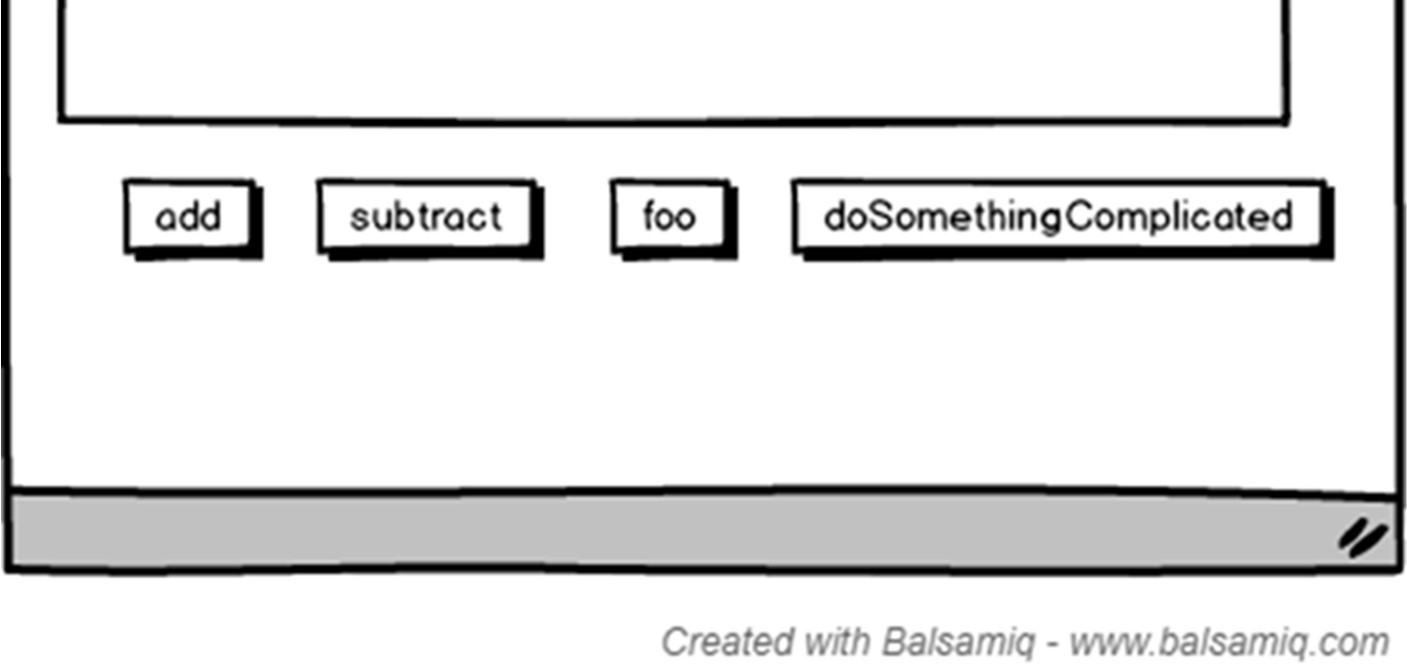
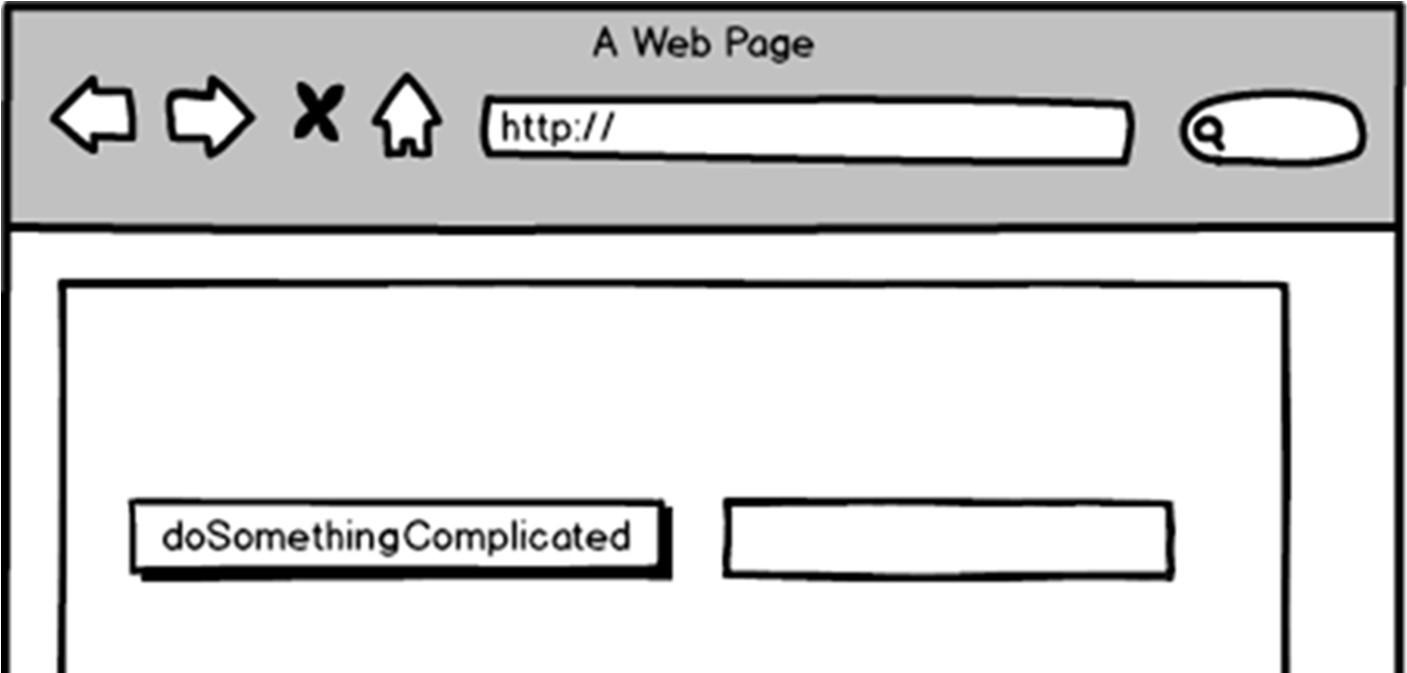
Screen begins blank



User can drag and drop the “add” function (which receives 2 arguments)



Alternatively, user can drag and drop the “doSomethingComplicated” function (which receives 1 argument)



In the more advanced case, the process can be done recursively – dragging and dropping a function into one of the newly created boxes.

