FizzBuzz Service

Overview

Following a test-first methodology, create an Angular service which exposes a public method named ‘fizzbuzz’ that takes a number input, processes it according to the fizzbuzz algorithm specifications, and returns a string.

Getting Set up

Clone the AngularTestingWorkshop repo and install dependencies via command line:

```
git clone https://github.com/JimTheMan/AngularTestingWorkshops
```

```
cd AngularTestingWorkshops/1_Fizzbuzz-Service/Fizzbuzz-Service_Start_Here
```

`npm install` and then `bower install`

Part 1. TDD / Unit Testing

Create this: an Angular “service service” with a public method `fizzbuzz()`

The fizzbuzz method should:

- return something.
- return a string.
- return ‘`fizz`’ if the number passed in is divisible by 3.
- return ‘`buzz`’ if the number passed in is divisible by 5.
- return ‘`fizzbuzz`’ if the number passed in is divisible by 3 and 5.
- if the number is divisible by neither 3 nor 5, return the number as a string[1].
- return ‘`Whoops! Please pass a whole number into fizzbuzz!`’ if the number passed into fizzbuzz is not a full digit value with data type Number.
- be able to handle whole numbers in string data type[2].
- return ‘`0 the hero`’ when the number passed in is zero.
- support negative numbers according to the same rules as above.

Part 2. E2e Protractor Testing

Create this: an html file with a text input and a <p> tag bound to a controller variable `inputValue`. `inputValue` variable is found by taking the value of the text input and sending it through the `fizzbuzz` method you created.

The page should:

- change the text in the <p> tag when the text in the input is changed.
- change the text in the <p> tag to ‘`fizz`’ when the text in the input is `3`.
- change the text in the <p> tag to ‘`buzz`’ when the text in the input is `10`.
- change the text in the <p> tag to ‘`fizzbuzz`’ when the text in the input is `3000`.
- change the text in the <p> tag to ‘`Whoops! Please pass a whole number into fizzbuzz!`’ when the text in the input is ‘`bazoinga`’.
- change the text in the <p> tag to ‘`fizz`’ when the text in the input is `-6`.
- change the text in the <p> tag to ‘`0 the hero`’ when the text in the input is `0`.
- change the text in the <p> tag to ‘`fizzbuzz`’ when the text in the input is `-30`.
- change the text in the <p> tag to `11` when the text in the input is `11`.
- change the text in the <p> tag to ‘`Whoops! Please pass a whole number into fizzbuzz!`’ when the text has no input.
- when app first loads up, show nothing in the <p> tag until the text input is changed by the user.

Note: You may use Angular 1 or Angular 2 for this workshop.

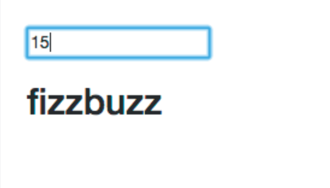
Run in browsersync: *gulp serve*

Run unit tests: *gulp test:auto*

Bonus Challenge UI & End-to-end Tests

Run e2e tests: *gulp protractor*

Page example:


15

fizzbuzz

[1] E.g. `fizzbuzz(42)` should return the string ‘`42`’.

[2] E.g. the string ‘`45`’ passed into `fizzbuzz` should be processed as just the number 45.

Created by @webWhizJim