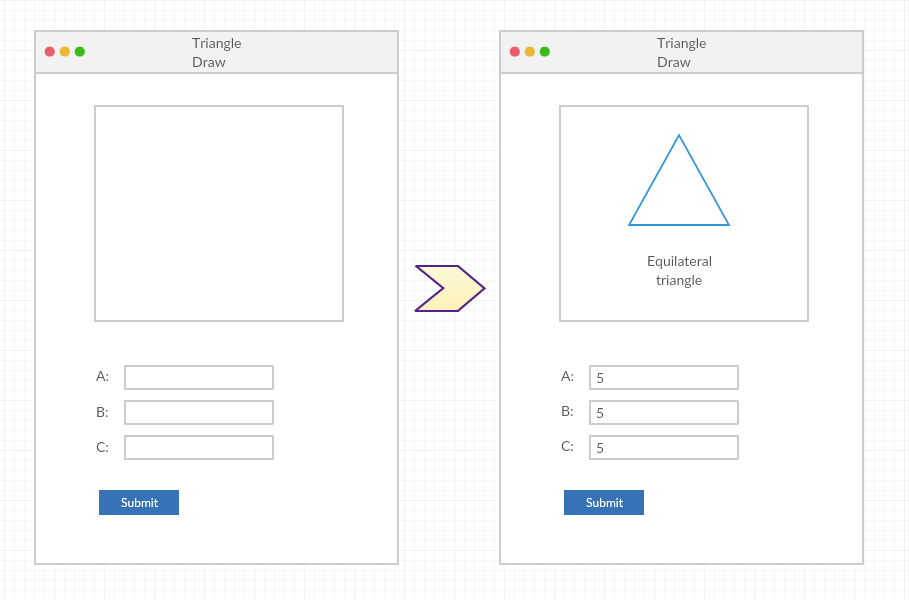
# Testing task “Triangle Draw”

The team has created a scientific software that has a 'Submit' button and 3 input fields ('A:', 'B:', 'C:').

Each input field represents the length of a triangle side.

It is stated that the software displays the triangle according to the input given.

In addition, it is given that there is text written below the triangle that states:



* If the triangle is ‘Equilateral’.

(An equilateral triangle has three equal sides)

* Or the triangle is ‘Isosceles’.

(An isosceles triangle has two equal sides)

* Or the triangle is ‘Scalene’.

(A scalene triangle has three different sides)

* Or state that such a triangle is ‘Not possible’.

## Try to list as many test case (attempts) as possible to test the software:

1. ['5','5','5'] (As shown in example)
2. ['5','4','3']
3. [‘a’,’b’,’c’]
4. [‘@’,’#’,’(’]
5. [‘1’,’1’,’9’]
6. [‘ ’,’ ’,’ ’]

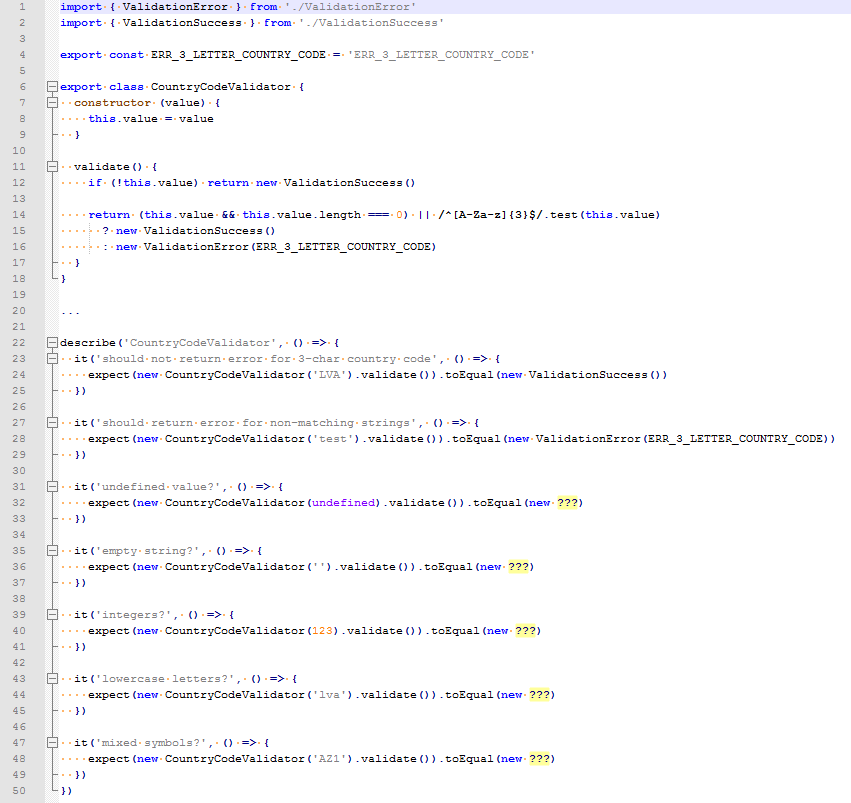
## Which of the listed test attempts are the 3 most important to validate that the functionality works correctly?

3,4,6

# React.JS code analysis exercise

The following JavaScript code contains a CountryCodeValidator class.

Please explain what this class might be used for and fill out the question marks in the testing function below the class to show what values are expected.



1. This class might be used to validate country code or some other data, before sending it on  
   server side.
   1. ValidationSuccess()
   2. ValidationSuccess()
   3. ValidationError(ERR\_3\_LETTER\_COUNTRY\_CODE)
   4. ValidationSuccess()
   5. ValidationError(ERR\_3\_LETTER\_COUNTRY\_CODE)

# Front-End Development Task

As already mentioned in the first task.

## Here are some User Stories which the You have to develop:

|  |
| --- |
| 1. Triangle Display Calculator   As a Scientific Persona,  I want to have a visual access to calculator, that can be used through web,  So that I can calculate different types of triangles  Acceptance Criteria:   * Triangle Calculator page that displays initial visual component:   + Drawing Window   + 3 sides input fields (“A side”, “B side”, “C side”)   + Calculate Button |
| 1. Triangle Calculation - Equilateral   As a Scientific Persona,  I want to figure out if the input given equals “Equilateral” triangle,  So that I can have fast input without spending a lot of time, drawing it  Acceptance Criteria:   * Input fields can be filled in only in whole numbers * All the triangle sides are equal * Triangle Calculator outputs text within the visual field – “Triangle is equilateral” |
| 1. Triangle Calculation - Isosceles   As a Scientific Persona,  I want to figure out if the input given equals “Isosceles” triangle,  So that I can have fast input without spending a lot of time, drawing it  Acceptance Criteria:   * Input fields can be filled in only in whole numbers * Two of the triangle sides are equal * Triangle Calculator outputs text within the visual field – “Triangle is isosceles” |
| 1. Triangle Calculation - Scalene   As a Scientific Persona,  I want to figure out if the input given equals “Scalene” triangle,  So that I can have fast input without spending a lot of time, drawing it  Acceptance Criteria:   * Input fields can be filled in only in whole numbers * All sides of the triangle are different * Triangle Calculator outputs text within the visual field – “Triangle is scalene” |
| 1. Triangle Calculation – Impossible triangle   As a Scientific Persona,  I want to know, if such triangle is not possible,  So that I get fast feedback that I need to adapt my calculations  Acceptance Criteria:   * Input fields can be filled in only in whole numbers * Triangle Calculator outputs text within the visual field – “Is not possible to calculate this kind of triangle” |
| 1. Triangle Draw (Bonus)   As a Scientific Persona,  I want to visually see, the triangle drawn,  So that I can quickly read through visualized drawing  Acceptance Criteria:   * After triangle is calculated, it is drawn in the drawing window |

## Additional requirements to consider, when developing the app:

* Use CSS & JS in Styling.
* Think about User Experience.
* Don’t forget about Error Handling.
* At least 3 Test Cases are Automated in jUnit.
* If the app developed in ReactJS – will be considered as an advantage.
* Use GIT to save code while developing and take commits by related stories

## Development Company for which You are building the app values:

* Code Readability.
* Code Maintainability.
* Clean Code.
* And other related Development Good Practices.

## The Lead Designer has suggested a design example for this app.

