Interface Design and Development

Pass Task 3.1: String Test Web App with VueJS

Overview

Control flow in VueJS enables you to easily add conditions and loops to dynamically generate your web application view. In this task, you will create a small web application that uses conditions to dynamically output custom messages to users.

Purpose:	Install and test the framework needed to get started.
Task:	Create a web app that tests a user's name and displays a custom message.
Time:	This task should be completed in your tutorial and submitted for feedback before the start of week 5.
Resources:	

- Lecture notes #3
- VueJS https://vuejs.org/

Submission Details

You must submit the following files:

- String Test source code (stringtest.html).
- Screenshot of the web app.

Make sure that your task has the following in your submission:

- The String Test web application is HTML5 compliant.
- Demonstrates understanding in using the VueJS framework.
- Demonstrates use of VueJS conditional directives.

SWINBURNE UNIVERSITY OF TECHNOLOGY

Instructions

The first task includes the steps needed for you to install the framework you will need in this unit. You will then use the VueJS framework (version 2.6.x) to create the 'Hello World' web page.

1.   Download and install the framework you need to get started. Ensure that you have:

- Installed VueJS (https://vuejs.org/)

    You should have the following files stored in appropriate folders

```
framework/
├── css/
│   ├── bootstrap.min.css
│   ├── bootstrap.min.css.map       ← map is used for debugging
│   ├── bootstrap-theme.min.css
│   └── bootstrap-theme.min.css.map
├── js/
│   ├── vue.min.js              (or vue.js)
│   ├── bootstrap.min.js
│   └── … …
```

Note: You may also want to download scripts to enable legacy versions of IE to fully support HTML5 and CSS3, and place them in the js folder. These are html5shiv https://cdnjs.com/libraries/html5shiv and respond https://cdnjs.com/libraries/respond.js.

Tip: You can also use direct link to specific versions of the scripts in your HTML file.

```
<!-- HTML5 shiv and Respond.js for IE8 support of HTML5 elements and media queries -->
<!-- WARNING: respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
 <script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
 <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
```

2.   If you don't already have one, make a directory (i.e., a 'folder') to store your framework (e.g. 'My Documents/cos30043/lab03'). On a Swinburne computer, you may wish to use a directory on your student drive or a USB storage device.

Note: Preferably replicate a copy of the framework files for each task.

3.   Save the file as stringtest.html in your lab03 directory.

4. Start the web application code with the template for VueJS found in lecture 03.

5. Implement a web application with the following logic:

   - It reads a name from the user, and displays back a message.

   - Check if the name entered is your name.

   - If the name is your name, output the message 'Awesome name!'

   - Otherwise output the 'not my name' message.

```
Web App: stringtest.html
————————————————

Uses: VueJS 3.x
------------------------------
--- Model:
- strName (which stores a String value)
------------------------------
--- Steps:
1: Assign strName using v-model with the prompt:
     'Please enter your name:'
2: v-show (if)"strName == '-place your name here-'"
3:        Output 'Awesome name!'
4: v-show (else)"strName != '-place your name here-'"
5:        Output strName, ' is a not my name'
```

Tip: Remember to use == as the conditional operator, = is an assignment operator.

Note: Conditional operator is case sensitive. For example, Caslon is not equal to caslon

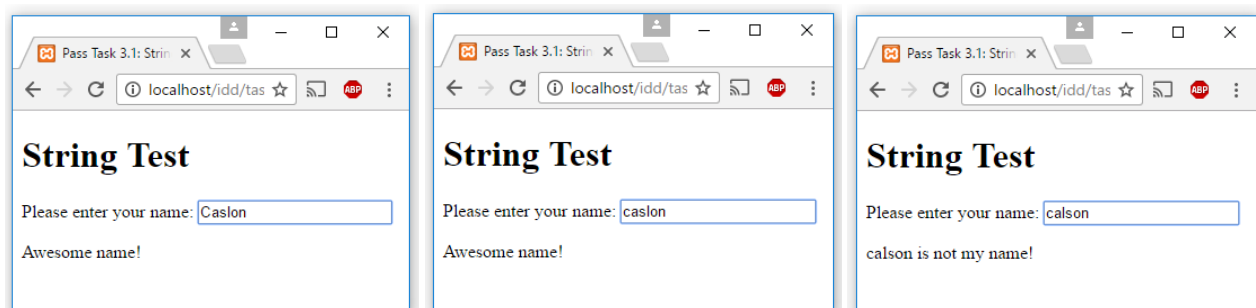**Hint:** Use the lowercase filter to convert the value of a variable into lowercase before you compare.

Figure 1: Screenshot of the web app with no Bootstrap mark up