# Assignment 2 – JavaScript

# Group Size: 1

# Value: 30% of course mark

# Due Date: Monday, 2nd May, 5.00 pm

## Summary:

You are asked to programmatically create an ordering web application for wine bottles. Your task is to create the necessary JavaScript for the application to correctly calculate the total cost of ordering different amounts of wine bottles and provide relevant feedback to the user about their order. Your application should be able to add a tax overhead to the total cost when the products are shipped to Auckland or Waikato. The specific tax amount should be retrieved dynamically using Ajax from a Web server. You should calculate the tax cost using the total price of the wine bottles, not the wine bottles price + shipping. Your application should also be able to factor in the costs of the user choosing different shipment options. If the user hasn’t selected a State from the corresponding input list or entered a valid email address in the corresponding text input field, upon clicking the “Estimate Total” Button, an alert box should inform the user to fill up the relevant input fields and focus should be placed in the corresponding input element. Once the user has filled up all the required input fields, your application should calculate the total cost of the order and update the webpage accordingly to provide relevant feedback to the user. A simple animation using the canvas element should also be fired up at this stage. An illustration is provided of how the application should look like after filling up the order and clicking the “Estimate Total “ button but you are encouraged to check the video demo of how the finished application should look like in the I: drive



## Requirements:

1. Your application must be built with JavaScript (no tinkering with the HTML or CSS.). No server-side coding neither.
2. Your application should be able to detect an unselected Shipping State list element and provide relevant feedback to the user in the form of an alert box. After the user clicks on the okay button of the alert box, the application should activate the focus on the list element for selecting a State.
3. Your application should be able to detect invalid or missing email addresses using a regular expression and inform the user through an alert box. Feedback about the invalid email address should be provided to the user in the form of an alert box. After the user clicks on the okay button on the alert box, the application should activate the focus on the email input element.
4. Your application should create the relevant event handler/listener for when the user presses the “Estimate Total” button.

Caution: I would recommend to wait at least 2 seconds before calculating the total cost, to give the AJAX functionality of your web application enough time to retrieve the tax information from the JSON file stored at the provided URL in requirement 7.

1. Your application should calculate and display the total amount of bottles being ordered
2. Your application should calculate and display the total shipping cost
3. Your application should apply and display the correct tax depending on the user selection of Shipment State. The correct “GST state tax” amount should be retrieved using AJAX from the url <https://dl.dropboxusercontent.com/u/10089854/Web3/Assignment2/stateTaxInfo.json> (namely 10% for Auckland, 5% for Waikato and 0% for Otago, Southland and Canterbury).

Disclaimer: I know this requirement is a bit silly since GST state tax doesn’t exist in New Zealand. My main goal for this assignment is not to create a realistic challenge, but rather an opportunity for you to show that you have mastered the content we have covered in class.

1. Your application should calculate and display the total cost of the order accurately
2. After your application displays the total cost of the order, create and store in local storage a Json object named “purchaseDetails” containing the number of Bottles ordered and the State selected in the ordering form.
3. Your code must be modular, readable, scalable and robust.
4. Your code must be properly documented
5. Your application should show a canvas animation upon the user clicking the “Estimate Total” button as shown in the video demo of the assignment.
6. The entry point to your application should be the “shop.html” file.
7. All code files and resources in a legible file structure should be compressed into a zip or rar single file and named in the following format: “YourlastnameFirstnameWeb3Assignment1”.

## Deliverables:

1. Please do not email the compressed file to me. Instead, the compressed web application should be dropped at: M:\Assessments Swallow Box\IN712Web3\Assignment2