

MADS4008 LAB TEST 2

SUBMISSION REQUIREMENTS

1. Compress your project files and any related assets (html,js,css,images and audio files) into a zip file
2. Name your zip file: `labTest2-petersmith-10123456.zip`
Substitute `petersmith` and `10123456` with your student name and id.
3. Upload the zip file to Blackboard by the date specified in the dropbox.

Please note that a submission will not be graded if:

1. The project is missing required files or assets
2. The submission is late
3. The submission contains academic dishonesty.

ACADEMIC INTEGRITY

This assessment must be completed individually.

Providing **any help** to other students is **not permitted**.

Help includes, but is not limited to:

- Exchange of ideas, advice about how to approach the problem
- Sharing website links or reference materials
- Sharing of devices, such as laptops, physical IOS devices, or MacInCloud accounts
- Sharing of credentials to college provided or personal email accounts, Google Drives, Dropbox, Blackboard, or similar

Communication with individuals either in the class, or outside the class, is strictly prohibited.

Any learner suspected of breaking the above rules will not have their submission marked; and, a report will be filed with the College's Academic Integrity department.

To prevent issues of academic integrity, it is imperative that you are vigilant that your code is not duplicated, in part or whole, by another learner. Duplication can mean: identical code; or code that is semantically identical in naming and overall construction. Should the instructors discover academic dishonesty as defined above and per the College Academic Integrity Policies, **a grade of 0** will be assigned to all parties involved.

PROBLEM DESCRIPTION

Notes for the Learner: This Lab Test 2 is designed to give you practical experience in building **simple(one screen)** cross platform applications using React Native that consume Restful APIs

Overview:

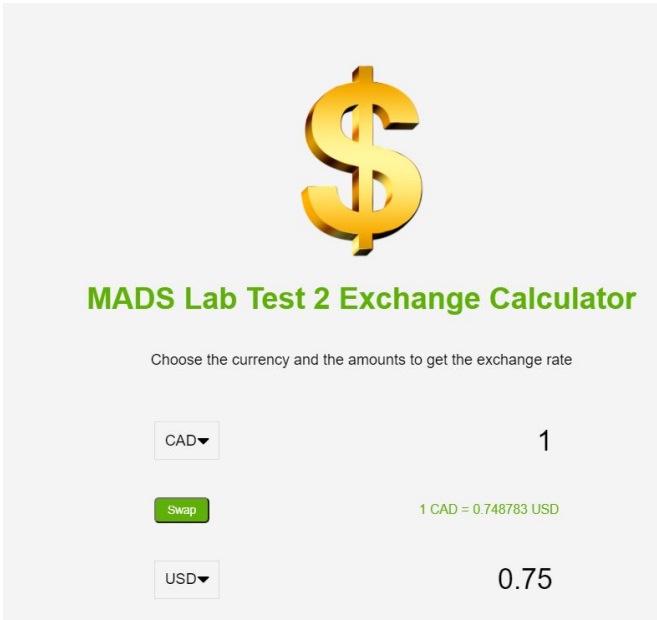
You have been contracted as a mobile developer to develop a simple one screen Money Exchange Calculator in the form of a cross platform app using React Native.

Requirement 1 (Look and feel):

[10 marks]

Please note, your app **MUST** be laid out and look similar to Figure 1.1. Additionally, your app must contain all the pertinent React Native components and styles so that it looks similar to Figure 1.1 (Some React Native Components include : Image, Picker, Text, Textinput, Button,etc)

Figure 1.1



The screenshot shows a mobile app interface for a money exchange calculator. At the top center is a large, 3D gold dollar sign icon. Below the icon, the title "MADS Lab Test 2 Exchange Calculator" is displayed in green. Underneath the title is a subtitle "Choose the currency and the amounts to get the exchange rate". The interface features two currency pickers. The first picker is labeled "CAD" with a dropdown arrow and has the value "1" next to it. Below this is a green "Swap" button. The second picker is labeled "USD" with a dropdown arrow and has the value "0.75" next to it. In the center, between the two pickers, the text "1 CAD = 0.748783 USD" is displayed in green.

Picker 1, and Picker 2 should list the following country codes CAD, USD, EUR, GBP

Requirement 2 (Money Exchange Rate Calculation):

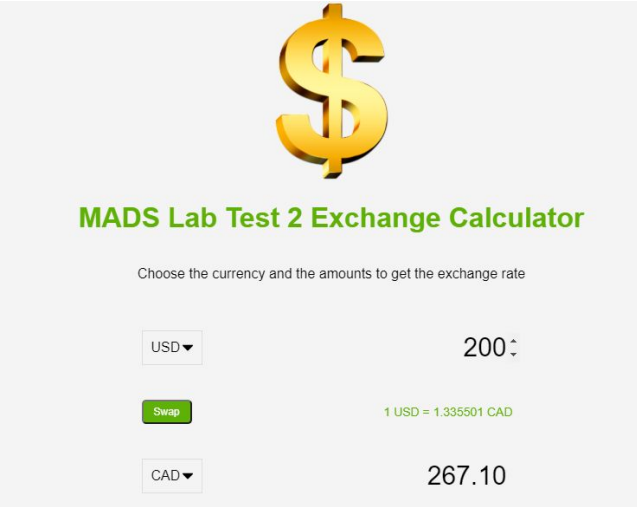
[10 marks]

Your app must implement the Exchange Rate Calculation functionality. This is done with the help of a REST API.

When the user changes the number value in the Text Input field, your app must calculate the exchange rate between the country code selected from the 1st picker against the country code selected in the 2nd picker.

For example, If the user selects USD as the first country code with the first picker and CAD as the second country code with the second picker and 200 is entered in the Text Input field then your app MUST send an HTTP request to the API, get the relevant data, calculate and display what is the exchange rate of \$200.00 USD in CAD. Also, your app must show what is the exchange rate of 1USD in CAD, as show in the Figure 1.2

Figure 1.2



The screenshot shows a web interface for an exchange rate calculator. At the top is a large gold dollar sign icon. Below it, the title "MADS Lab Test 2 Exchange Calculator" is displayed in green. Underneath the title is the instruction "Choose the currency and the amounts to get the exchange rate". The interface features two currency pickers: "USD" and "CAD", each with a dropdown arrow. To the right of the "USD" picker is a text input field containing the number "200" and a small up/down arrow icon. Below the "USD" picker is a green "Swap" button. To the right of the "Swap" button, the text "1 USD = 1.335501 CAD" is displayed in green. Below the "CAD" picker, the calculated result "267.10" is shown.

See API's endpoint :

https://api.exchangerate-api.com/v4/latest/Country_Code_of_FirstPicker

If the user selects USD as the first country code picker then the app should send an HTTP request to the following endpoint :

<https://api.exchangerate-api.com/v4/latest/USD>

Figure 1.3

The below is a snapshot of the JSON response that will be sent back from the API.

```
{
  "base": "USD",
  "date": "2020-07-30",
  "time_last_updated": 1596067451,
  "rates": {
    "USD": 1,
    "AED": 3.672023,
    "ARS": 72.163534,
    "AUD": 1.394064,
    "BGN": 1.666869,
    "BRL": 5.141434,
    "BSD": 1,
    "CAD": 1.335501,
    "CHF": 0.91596,
    "CLP": 766.257003,
    "CNY": 7.001237,
    "COP": 3656.285714,
    "CZK": 22.39891,
    "DKK": 6.34448,
    "DOP": 58.256449,
    "EGP": 15.919967,
    "EUR": 0.850915,
    "FJD": 2.129875,
    "GBP": 0.770985,
    "GTQ": 7.695901,
    "HKD": 7.749961,
    "HRK": 6.386935,
    "HUF": 296.246273,
    "IDR": 14679.000638,
    "ILS": 3.40872,
    "INR": 74.802881,
    "ISK": 135.347785,
    "JPY": 105.068861
  }
}
```

Requirement 3 (Swap button functionality):

[5 marks]

When the user clicks on the swap button, the country code value of the first picker and second picker must be swapped with each other. Also, the app must re-calculate and display the exchange rate of the two countries based on the new swapped data.

Requirement 4 (Create a customized Component):

[5 marks]

Your app must have the default App component and you **MUST** create an additional customized component and use it in an intuitive way.

Note, you are free to use either Functional Components or Class Based Components. The option is yours to make.

THE END