



DOUGLAS COLLEGE

COMMERCE AND BUSINESS ADMINISTRATION

CSIS 3380-002: ADVANCED WEB PROGRAMMING WITH JAVASCRIPT & AJAX

Quiz 2 (November 16, 2021)

Total Duration: 90 minutes

Total marks: 100

Hands-on Problems (40 + 60 = 100 marks).

Instructions

- Download the Quiz2 folder and rename it to include your name as: lastName_studentID
- Please use the html file **question1.html** in the Quiz2 folder to answer the question 1.
- Write your name and ID in the <body> part of the html document where specified

Note: Points will be deducted if you do not include this identifying information as instructed above.

- For question 2, read the instructions and follow along.
- **Submission:** Once you are done, upload your work to Blackboard under Quiz2.

Question 1: Use Fast FOREX Currency Converter API (<https://www.fastforex.io/>) to convert amount from one currency to another and display the result on the webpage. You can either use **jQuery ajax** or **Fetch API** to fetch data from the third-party API. **(40 Marks)**

There are two steps that you need to accomplish:

Step I: Make an AJAX request at the following API endpoint:

https://api.fastforex.io/fetch-all?api_key=e780bf65c0-11aee9d708-r2aisy

Check the response, get all the currency names and use these currency names to populate the select boxes with id="from_curr" and id="to_curr".

Hint: Use the following code for appending an option to select

```
jQuery:      $("select").append('<option>My option</option>');
JavaScript:  select = document.getElementById('selectElementId');
              opt = document.createElement('option'); ..., select.appendChild(opt);
```

Step II: Next, the user enters the amount, selects a *from currency value* from select box with id="from_curr" and then, selects a *to currency value* from select box with id="to_curr" and hits the "Convert Currency" button. The result is calculated and is displayed in the input field with id = "answer".

On button click, another AJAX request is made at the following endpoint:

https://api.fastforex.io/convert?api_key=e780bf65c0-11aee9d708-r2aisy&amount={amount}&from={curr}&to={curr}

You should replace the values for amount, from currency and to currency with the ones entered by the user.

Example: The following API displays results when amount is 2, for_curr is CAD, to_curr is USD:

https://api.fastforex.io/convert?api_key=e780bf65c0-11aee9d708-r2aisy&amount=2&from=CAD&to=USD

Note: You can quickly generate your own API key by visiting the site <https://www.fastforex.io> and clicking the button "Get your Free Trial API Key" and entering the credentials. Or, you can use the one provided in the example.

Quiz 2

--- Replace with Student Name and Student Id ---

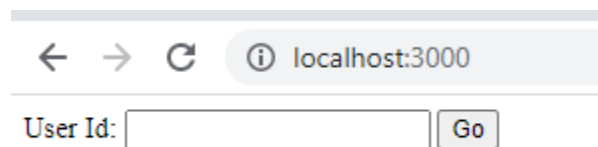
Question 1 Currency Convertor

2	CAD	USD	Convert Currency	1.59764
---	-----	-----	------------------	---------

Question 2.

JSONPlaceholder is a free REST API that provides fake data which we can use to test our application. Using the posts resource at endpoint <https://jsonplaceholder.typicode.com/posts?userId=somevalue>, build a Node/Express server and render the website with the API data. Complete the following tasks: **(60 Marks)**

- Handle an HTTP GET request from the client at the root route localhost:3000/
- Send back the following HTML file to the client as the response (**use question2.html**):



The screenshot shows a web browser window with the address bar displaying 'localhost:3000'. Below the address bar, there is a form with the label 'User Id:' followed by an input field and a 'Go' button.

- After the client populates the `userId` field, parse the `requestBody` to retrieve `userID`
- Validate the `userID` (it should take a value between 1 and 10)
 - If not valid – send a response back to the user that the `userID` is not valid
 - Otherwise, structure the URL to fetch the posts from the external service for that `userID`For example, Append the `userID` as a query string parameter, such as “`?userId=2`”

Note: The response argument in the `http.get()` callback is not the body, but rather an [http.ClientResponse](#) object. You need to assemble the body:

```
http.get(url, function(res){
  let responseBody = "";

  res.on('data', d =>{
    responseBody += d;
  });

  res.on('end', function(){
    responseBody = JSON.parse(responseBody);
    //do stuff with responseBody
  });
});
```

- Send the results back to the client in the form of a table:

2 11 et ea vero quia laudantium autem	delectus reiciendis molestiae occaecati non minima eveniet qui voluptatibus accusamus in eum beatae sit vel qui neque voluptates ut commodi qui incidunt ut animi commodi
2 12 in quibusdam tempore odit est dolorem	itaque id aut magnam praesentium quia et ea odit et ea voluptas et sapiente quia nihil amet occaecati quia id voluptatem incidunt ea est distinctio odio
2 13 dolorum ut in voluptas mollitia et saepe quo animi	aut dicta possimus sint mollitia voluptas commodi quo doloremque iste corrupti reiciendis voluptatem eius rerum sit cumque quod eligendi laborum minima perferendis recusandae assumenda consectetur porro architecto ipsum ipsam
2 14 voluptatem eligendi optio	fuga et accusamus dolorum perferendis illo voluptas non doloremque neque facere ad qui dolorum molestiae beatae sed aut voluptas totam sit illum
2 15 eveniet quod temporibus	reprehenderit quos placeat velit minima officia dolores impedit repudiandae molestiae nam voluptas recusandae quis delectus officii harum fugiat vitae
sint suscipit perspiciatis velit	suscipit nam nisi quo aperiam aut asperiores eos fugit maiores voluptatibus quia voluptatem quis ullam qui in alias quia est
2 16 dolorum rerum ipsa laboriosam odio	consequatur magni mollitia accusamus ea nisi voluptate dicta
2 17 fugit voluptas sed molestias voluptatem provident	eos voluptas et aut odit natus earum aspernatur fuga molestiae ullam deserunt ratione qui eos qui nihil ratione nemo velit ut aut id quo
2 18 voluptate et itaque vero tempora molestiae	eveniet quo quis laborum totam consequatur non dolor ut et est repudiandae est voluptatem vel debitis et magnam
2 19 adipisci placeat illum aut reiciendis qui	illum quis cupiditate provident sit magnam ea sed aut omnis veniam maiores ullam consequatur atque adipisci quo iste expedita sit quos voluptas
2 20 doloribus ad provident suscipit at	qui consequuntur ducimus possimus quisquam amet similique suscipit porro ipsam amet eos veritatis officii exercitationem vel fugit aut necessitatibus totam omnis rerum consequatur expedita quidem cumque explicabo

Grading Rubrics:

For each question, if all the requirements are met, you get full marks. You lose 5 marks for each of the missing requirement.