|  |  |  |
| --- | --- | --- |
| American University of SharjahSchool of Engineering Department of Computer Engineering  P. O. Box 26666  Sharjah, UAE  **Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **ID : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |  | **Instructor:** Imran A. Zualkernan **Office**: ESB-2066  **Phone**: 971-6-515 2953  **Fax**: 971-6-515 2979  **e-mail**: izualkernan@aus.edu  **Semester**: Fall 2020 |

COE457 – Final Exam

Fall 2020

Total Time: 2 Hours

/100

**The quiz open book and notes. Attempt both questions.**

**Please note that questions have been designed so that it is not possible for two students to come up with code that is structurally similar. Two solutions where the names of the variables have been changed or similar cosmetic changes done will be considered a violation of AUS academic integrity code and will result in an automatic F in the course for BOTH parties.**

**You are required to upload:**

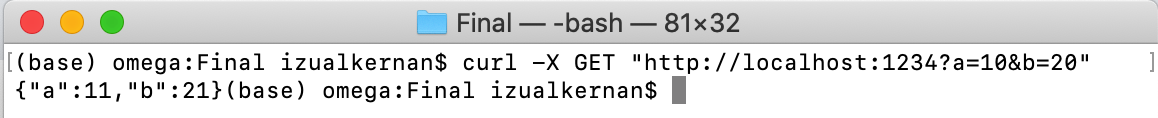
1. This file with all the code pasted into it.

2. Please only submit a WORD file. Any other format will be rejected.

**Q1.** (50 points)

Q1 a. (10 points) Write a Node.js program using Express that takes two numbers and using GET returns a JSON structure containing an increment to each number. The input numbers are labelled **a** and **b**.

**Expected Output:**



**Grading Rubric:**

|  |  |  |  |
| --- | --- | --- | --- |
| **0-3** | **4-6** | **7-8** | **9-10** |
| * Node.js and Express are not used. | * Node.js and Express are used but the output is not in the right JSON format. | * Node.js and Express are used and the output is in JSON format and almost correct. | * Node.js and Express are used and the output is in JSON format and correct. |

**Solution (Past your code here):**

**Screenshot (Paste your screenshot here)**

**Leaving this space empty will result in automatic deduction of 5 points.**

Q1 b. (30 points) Write a proxy server (running on port 9999) that will be used to access the Node.js program written in part (a). The proxy server does the following:

1. The proxy should send back a cookie called “message” that has the product of the two numbers a, and b.

2. The proxy should also save the product of the two incremented numbers in a MongoDB database with a timestamp.

3. The proxy should show a smiley on top of the tab in the browser.

You cannot use the node packages http, http-proxy, http-proxy-middleware or similar to implement the proxy server. The proxy has to be written by-hand on top of Node.js and Express.

**Expected Output** (Please show similar for your program):

Graphical user interface, application

Description automatically generated

Table

Description automatically generated

**Grading Rubric:**

|  |  |  |  |
| --- | --- | --- | --- |
| **0-3** | **4-6** | **7-8** | **9-10** |
| * Node.js and Express are not used. OR * Any of the packages not allowed are used OR * Proxy is not working at all. | * Node.js and Express are used AND * Proxy is working AND * One of the three is working perfectly.   + Cookies are set correctly.   + MongoDB is saving the message   + Smiley is being shown on the top. | * Node.js and Express are used AND * Proxy is working AND * Two of the three are working perfectly.   + Cookies are set correctly.   + MongoDB is saving the message   + Smiley is being shown on the top. | * Node.js and Express are used AND * Proxy is working AND * Cookies are set correctly. AND * MongoDB is saving the message AND * Smiley is being shown on the top. |

**Solution (Past your code here):**

**Screenshot (Paste your screenshot here)**

**Leaving this space empty will result in automatic deduction of 20 points.**

Q1 c. (10 points) Write an html file (q1.html) that uses jQuery with the original server (part a) and the proxy (part b) to automatically update the addition of the two numbers in a web-browser. The file q1.html must be launched from the local file system and not from localhost:1234.

Please note that the Smiley or the Cookie do not have to be set for this part.

**Program Expected Output:**

Graphical user interface, text, application, email

Description automatically generated

**Grading Rubric:**

|  |  |  |  |
| --- | --- | --- | --- |
| **0-3** | **4-6** | **7-8** | **9-10** |
| * jQuery is not used. | * jQuery is used AND * Both original and proxy servers are used but the output is now correct. | * jQuery is used AND * Both original and proxy servers are used. * The output is correct. * The output works only with localhost and not with the local file system. | * jQuery is used AND * Both original and proxy servers are used. * The output is correct. * The output works   with localhost and the local file system. |

**Solution (Past your code here):**

**Screenshot (Paste your screenshot here)**

**Leaving this space empty will result in automatic deduction of 10 points.**

**Q2.** (50 points)

Q2 a (40 points) Write a service using the following two APIs that upon hitting a button suggests a new activity and shows a nice picture with a text overlay of the activity as well. You can use a single picture of your choice as the background picture.

The two APIs are:

<https://textoverimage.moesif.com/#documentation>

<https://www.boredapi.com/>

You must use only html and jQuery for this program and cannot use your own back-end server (e.g., Node.js + Express).

**Expected Output (use any background picture of your choice)**

Graphical user interface, application

Description automatically generated

**Grading Rubric**

|  |  |  |  |
| --- | --- | --- | --- |
| **0-3** | **4-6** | **7-8** | **9-10** |
| * No output is shown, and the APIs are not used OR * Own server-side is used OR * jQuery is not used. | * jQuery is used AND * Text information is extracted from the one API and this information is shown in text. | * jQuery is used AND * Text information is correctly extracted from one API AND * Some of the image overlay API seems to be working or in the right direction. | * jQuery is used AND * Text information is correctly extracted from one API AND * The text overlay works perfectly as shown. |

**Solution (Past your code here):**

**Screenshot (Paste your screenshot here)**

**Leaving this space empty will result in automatic deduction of 20 points.**

Q2 b (10 points). Modify the program in Part (a) so that is also sends every activity to an MQTT topic bored/todo. Show your output in MQTTBox or similar. You must use a locally hosted MQTT broker (e.g., Mosquitto or Ponte).

**Expected Output:**

Graphical user interface, text, application, email

Description automatically generated

**Grading Rubric:**

|  |  |  |  |
| --- | --- | --- | --- |
| **0-3** | **4-6** | **7-8** | **9-10** |
| * The MQTT does not publish at all. | * MQTT publishes but a locally hosted MQTT broker is not being used. | * MQTT publishes. * A locally hosted MQTT broker is being used. * The output is almost correct. | * MQTT publishes. * A locally hosted MQTT broker is being used. * The output is always correct. |

**Solution (Past your code here):**

**Screenshot (Paste your screenshot here)**

**Leaving this space empty will result in automatic deduction of 5 points.**