# IS216 Term 2022/23 T1

## Mini Lab Test [15 marks]

## General Instructions:

1. This is a time-bound (1 hour), open-book, open-Internet, and **individual** test.

2. You must test your web pages using **Google Chrome Web Browser** (Version 98.0.x or later). Your graders will be using only **Google Chrome Web Browser** (Version 98.0.x or later) to test your web pages.

3. No questions will be entertained by the IS216 teaching team (faculty/instructor/Teaching Assistants) during the test period. If necessary, make your own assumptions and complete test questions.

4. You must use only standard HTML5, CSS, Bootstrap (Version 5.2), JavaScript, and Axios in your solutions unless the question specifies otherwise. Do NOT use any other third-party libraries (e.g. Angular, React, or others).

5. Use meaningful names for HTML class/id and JavaScript variables/functions. You must indent your code (HTML/CSS/JavaScript) properly. Failure to do so will attract a penalty of up to **20%** of your score for the corresponding question.

6. You **MUST** include your name and SMU email ID in the comments of all your submitted source files. Failure to do so will attract a penalty of up to **20%** of your score for the corresponding question.
   For example, if your registered name is "LEE Oppa" and email ID is "lee.oppa.2020", include the following comment at the beginning of each source file you write.

| **HTML files** | **CSS, JavaScript files** |
|---|---|
| `<!--`<br>`    Name:  LEE Oppa`<br>`    Email: lee.oppa.2020`<br>`-->` | `/*`<br>`    Name:  LEE Oppa`<br>`    Email: lee.oppa.2020`<br>`*/` |

7. You may wish to comment out the parts in your code which cause errors. Commented code will not be marked.

## Academic Integrity

All student submissions will be thoroughly checked by an SMU-approved source code plagiarism checker software program AND an additional external software program. The source code checking will be conducted across all submissions.

1. Suspected plagiarism cases will be reported immediately to the IS216 faculty in charge and SCIS Dean's Office for further investigation.
2. Students in the suspected cases will be informed accordingly by their section faculty, and the incident will be escalated to the SMU University Council of Student Conduct.
3. More information about the SMU Student Code of Conduct can be found at this link: https://smu.sharepoint.com/sites/ucsc/Documents/Code%20of%20Academic%20Integrity.pdf

## Submission Instructions

1. **Due Date**
   **4th Oct 2022 (Tuesday) 11 PM Singapore Time** (or a later time as announced by the IS216 faculty during the lab test).
   Late submission policy is as follows:

| | |
|---|---|
| Submit **within 5 minutes** of set deadline | 10% penalty<br>(of your score for the entire test) |
| Submit **within 10 minutes** of set deadline | 25% penalty<br>(of your score for the entire test) |
| Submit **within 15 minutes** of set deadline | 50% penalty<br>(of your score for the entire test) |
| **Beyond 15 minutes**, 0 mark<br>(submission will NOT be accepted) | |

2. **Zip** up all files in **Q1/Q2/Q3 folders** into **<YOUR_SMU_ID>.zip**
   - For example, **lee.oppa.2020.zip**
   - Verify by unzipping this zip file – check the content inside
   - Incorrect submission file name WILL attract a penalty of up to 20% of your score for the entire test.
   - Only **zip** format is accepted. **.7z, rar** or other **compression formats** are **NOT** accepted.
   - Until the correct **zip** format is submitted again by the student, it will be assumed that the student has NOT made the submission and late submission policy will apply.
   - Submit the **zip** file to the following location:
     - **IS216-Web Application Development II-Merged** eLearn page: https://elearn.smu.edu.sg/d2l/home/333446
   - Go to Assignments → Mini Lab Test → Submit your ZIP file
   - It is your (student's) individual responsibility to ensure that the zip file submission was successful. Your section faculty and Teaching Assistants will NOT verify the submission for you.

**Legend**

🚫 Do not edit this given resource

✏ Your answer/code goes here into this given resource file.

# Q1: Basic CSS (5 Marks)

## Instructions

Given resources in folder `Q1`

- 🚫 `q1.html`
- ✏️ `q1.css`

1. Open `q1.html` in a browser and examine how it looks.
2. You are NOT allowed to edit `q1.html`. Make sure that your new `q1.css` works with the original `q1.html` file.

Here is what you need to achieve: [5 marks]

1. Edit only `q1.css` so that `q1.html` looks close to "After" shown in Figure 1.

2. You will not be penalized for slight variations between your answer and what is shown here (e.g. it's OK if the space between each red-bordered quotation box is 10 or 20px, as long as there is a visible space. The corners of the red quotation boxes need to be visibly slightly rounded, but we do not require a specific curvature.)

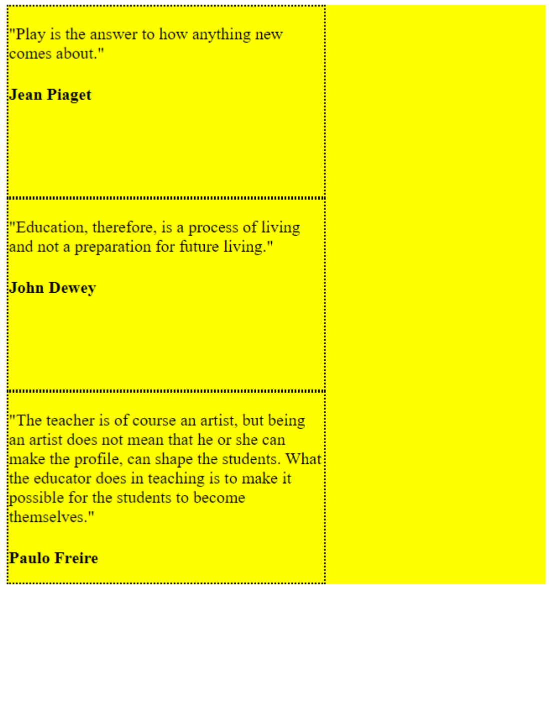
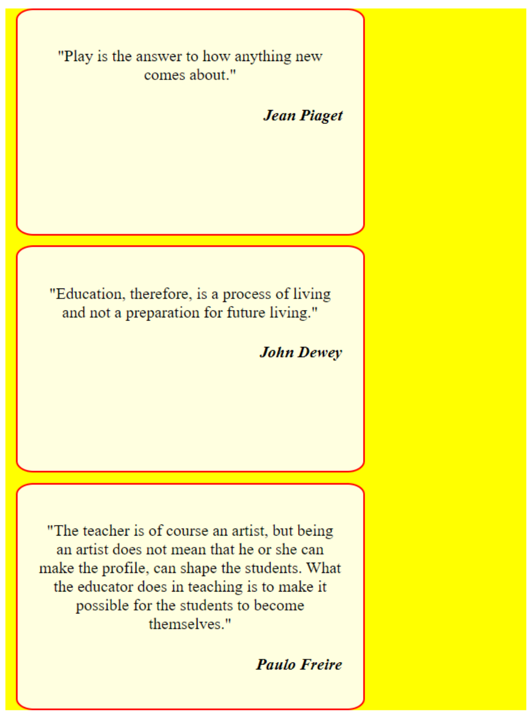
| Before | After |
|---|---|
| <br>"Play is the answer to how anything new comes about."<br>**Jean Piaget**<br>"Education, therefore, is a process of living and not a preparation for future living."<br>**John Dewey**<br>"The teacher is of course an artist, but being an artist does not mean that he or she can make the profile, can shape the students. What the educator does in teaching is to make it possible for the students to become themselves."<br>**Paulo Freire** | <br>"Play is the answer to how anything new comes about."<br>***Jean Piaget***<br>"Education, therefore, is a process of living and not a preparation for future living."<br>***John Dewey***<br>"The teacher is of course an artist, but being an artist does not mean that he or she can make the profile, can shape the students. What the educator does in teaching is to make it possible for the students to become themselves."<br>***Paulo Freire*** |

*Figure 1: How `q1.html` looks like currently (left) and how `q1.html` should look like (right)*

# Q2: Bootstrap for Responsiveness (5 Marks)

**Instructions**

Given resources in folder `Q2`

- 🚫 `img/[several JPEGs]`
- 🚫 `q2_requirements.mp4`
- ✏ `q2.html`

1. Open `q2.html` in a browser and examine how it looks when you change the width of your browser.

Here is what you need to achieve:

1. Watch `q2_requirements.mp4` (it is a captioned video with no sound). You need to edit `q2.html` so that the page behaves in the same way:

   - When the width is >=1200px (XL breakpoint), all 6 pictures appear in a single row.

   - When the width is <1200px but >=992px (LG breakpoint), the pictures are broken into 2 rows. The logo picture is twice the width of the photos.

   - When the width is <992px but >=768px (MD breakpoint), the pictures are broken into 3 rows of 2 pictures per row.

   - When the width is <768px, the pictures appear in 6 rows of 1 picture per row.

2. You are only allowed to edit & submit `q2.html`. Make sure it works with the JPEGs in the img folder. [5 marks]

# Q3: JS/DOM/Async: Jokes (5 Marks)

**Instructions**

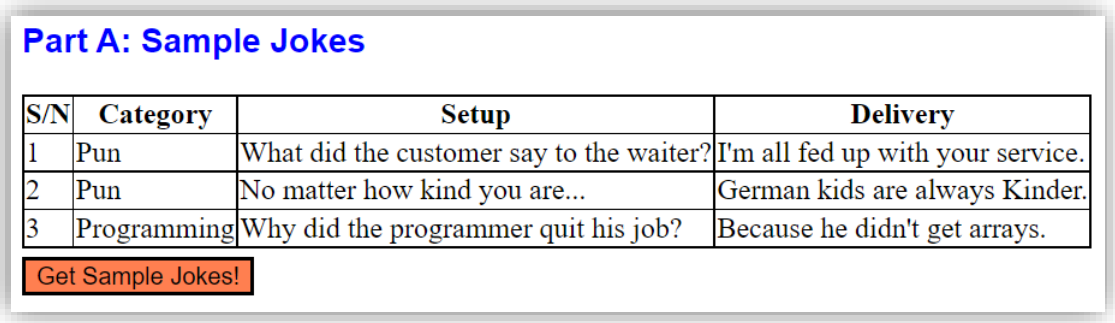
Given resources in folder `Q3`

- 🚫 `q3_getJoke.php`
- 🚫 `q3_jokes.html`
- ✏️ `q3.js`

1. The variable `sampleJokes` in the script section of `q3_jokes.html` is provided, which is to be used in solving part A below.

Here is what you need to achieve:

A. Complete the code in the function `getSampleJokes()` in `q3.js` such that when the user clicks on the "Get Sample Jokes!" button, the table in the HTML will be populated with data in `sampleJokes`. Only jokes that have the value "twopart" under the key "type" in the JavaScript object should be displayed. The desired display is shown in Fig. A below. [3 marks]



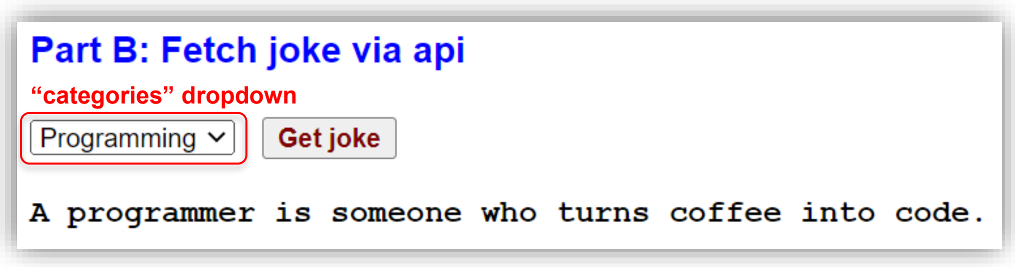
Part A: Sample Jokes

| S/N | Category | Setup | Delivery |
|---|---|---|---|
| 1 | Pun | What did the customer say to the waiter? | I'm all fed up with your service. |
| 2 | Pun | No matter how kind you are... | German kids are always Kinder. |
| 3 | Programming | Why did the programmer quit his job? | Because he didn't get arrays. |

*Figure 3A: Output of displaying "twopart" sampleJokes*

Note: DO NOT hard code the table above. You are to use the data given in the variable `sampleJokes` to populate the table. We will be testing your code with different data.

B. Complete the code in the function `getJoke()` such that when the user clicks on the "Get Joke" button, the value of the "categories" dropdown list will be sent to `q3_getJoke.php` using axios as the "query" parameter, and the result returned will be displayed in the paragraph with id "`jokeText`". E.g. Sending request using "`q3_getJoke.php?query=Programming`" will yield the output shown in Fig. B below. [2 marks]



Part B: Fetch joke via api

```
A programmer is someone who turns coffee into code.
```

*Figure 3B: Sample output of displaying a joke retrieved using the API when "Programming" option is selected*

You are only allowed to edit & submit `q3.js`.

END