# IS216 Term 2022/23 T1
# Sample Mini Lab Test Set 2 [30 marks]

*This lab test was an actual lab test used in 2021/22 T2 & is now released as a practice paper.*

General Instructions:

- Time given: 1 hour. Format: open-book, open-Internet, and to be completed individually.
- Your graders will be using only **Google Chrome Web Browser** to test your web pages.
- 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.
- 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).
- 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.
- 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>`*/` |

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

Submission Instructions:

- Zip up all files in Q1/Q2 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.

There are 2 questions for this test:

1. [css] World peace webpage (10 marks)
2. [bootstrap, Javascript, Axios] World military power (20 marks)

**Legend**

🚫 Do NOT edit this given resource file.

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

## Q1. [CSS] World Peace Webpage [10 marks]

Given resources:

➔ Q1\

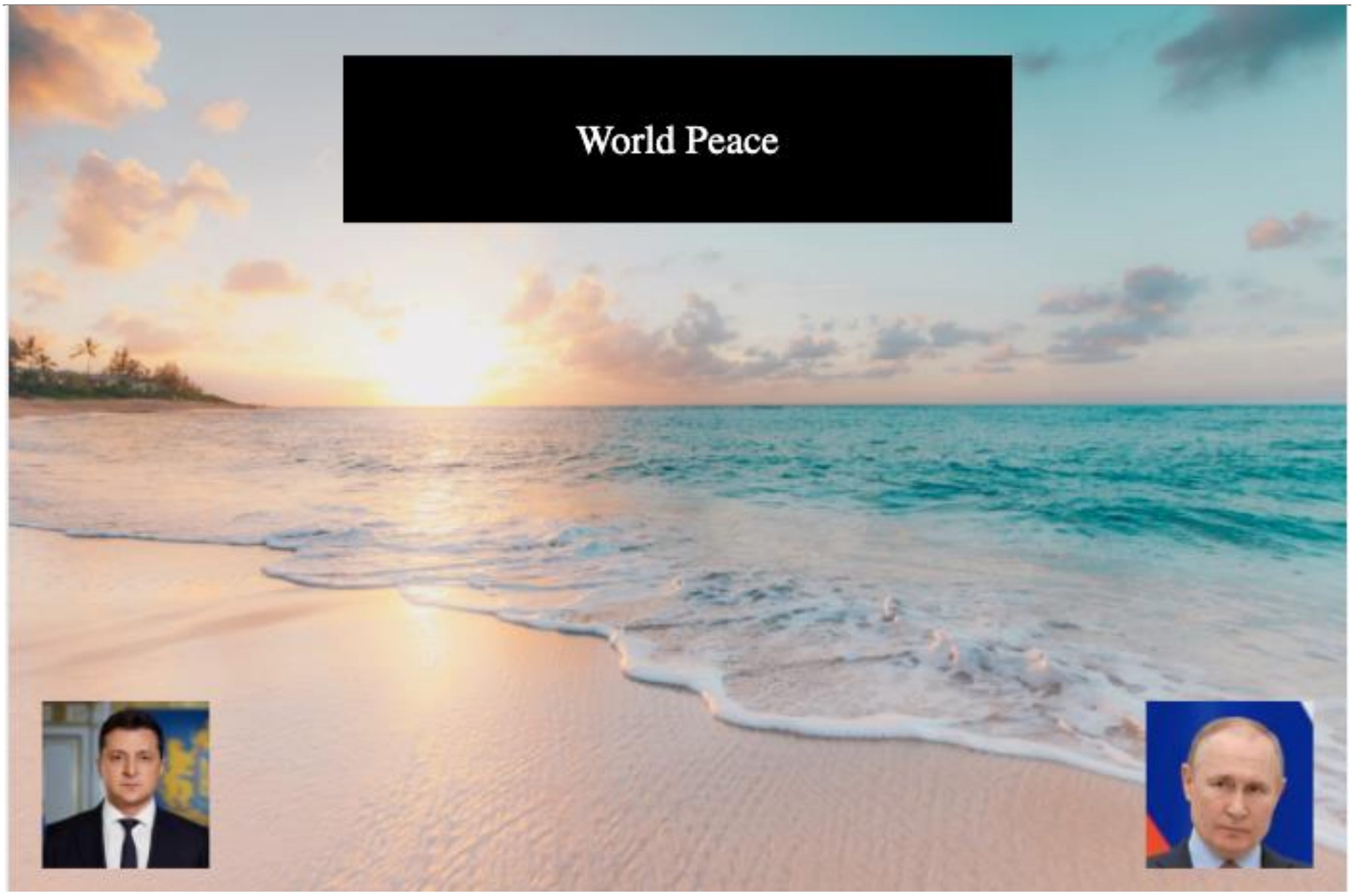
- ✏ home.html
- ✏ home.css
- 🚫 images/* *(There are 3 JPG image files. Do NOT move them.)*

In this question, you will use HTML and CSS to decorate and build a responsive web page **WITHOUT** using Bootstrap. You are strictly **NOT** allowed to use Bootstrap for this question.

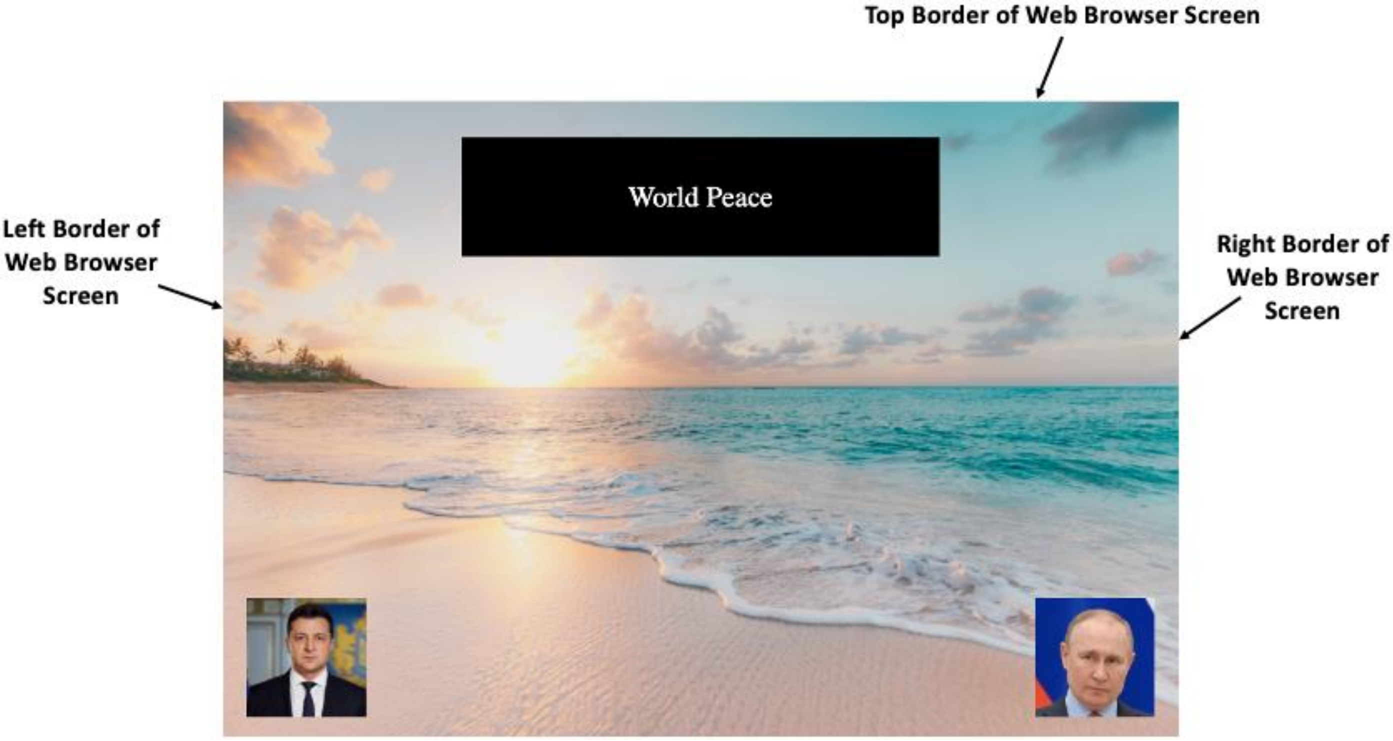
**IMPORTANT**

You are to **edit home.html** and **home.css only**. Do **NOT** edit other files. Do **NOT** add additional files (it will **NOT** be included for grading purposes).

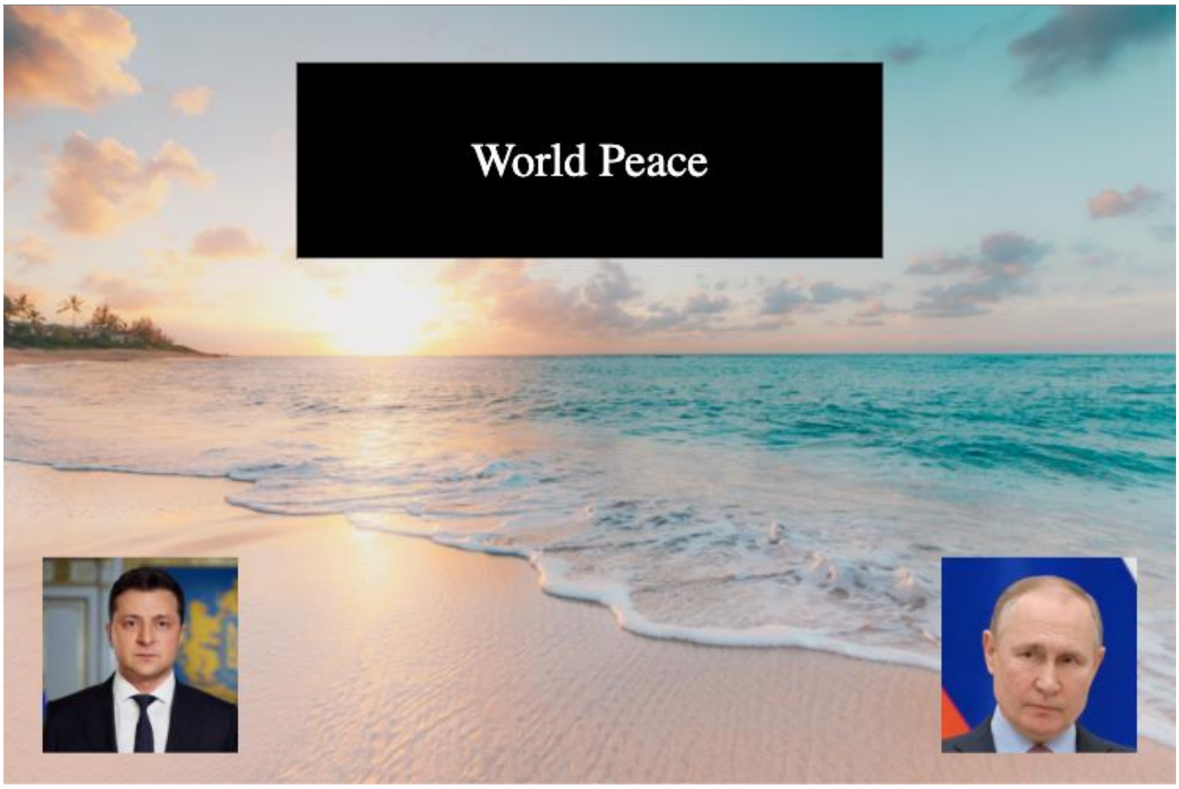
When done correctly, the **Web Browser** will display **home.html** as shown below:



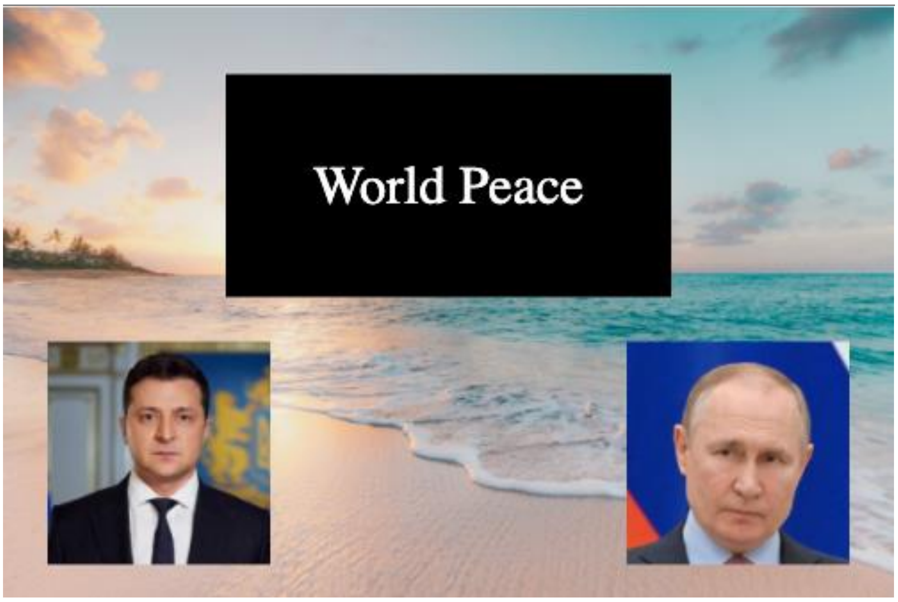
***Test Case #1***
**Web Browser Screen Width 800 pixels**


Top Border of Web Browser Screen
Left Border of Web Browser Screen
Right Border of Web Browser Screen
World Peace

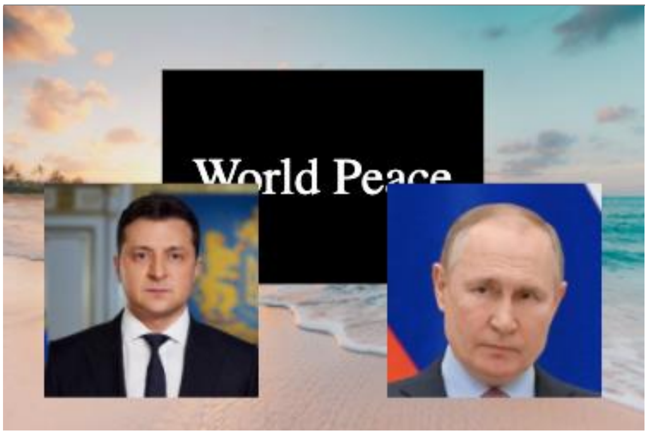
***Test Case #1***
**Web Browser Screen Width 800 pixels**



***Test Case #2***
**Web Browser Screen Width 600 pixels**



***Test Case #3***
**Web Browser Screen Width 400 pixels**



***Test Case #4***
**Web Browser Screen Width 300 pixels**

Please ensure that the following are implemented correctly:

**NOTE: "At all times"** below implies **Web Browser Screen width** between **300 pixels** and **800 pixels**, both inclusive *(as shown in the previous pages)*.

1. At all times, the "beach" image (**beach.jpg**) spans the **full width of the Web Browser Screen**.
   a. You are **NOT** allowed to edit **beach.jpg** file.
   b. You are **NOT** allowed to modify **beach.jpg** image's original width-to-height ratio. For example, if the width shrinks by 30%, the height must also shrink by 30% and vice versa.

2. At all times, the **black rectangle** (containing the text "World Peace" in white font color) will fulfill all of the following conditions:
   a. Be situated **30 pixels** below the **Top Border of the Web Browser Screen**
   b. Be situated **70 pixels** away from both:
      i. The **Left Border of the Web Browser Screen**, AND
      ii. The **Right Border of the Web Browser Screen**
   c. Span **50 percent** of the **Web Browser Screen width**
   d. Has a **fixed height** of **100 pixels**
   e. Be horizontally centered in the **Web Browser Screen**

3. At all times, the text "World Peace" will fulfill all of the following conditions:
   a. Be **horizontally centered** inside the **black rectangle**
   b. Be **vertically centered** inside the **black rectangle**
   c. Font color is **white**
   d. Font size is **24 pixels**

4. At all times, **russia.jpg** and **ukraine.jpg** BOTH will always span **100 pixels width** AND **100 pixels height**.
   a. You are **NOT** allowed to edit both image files.
   b. You are **NOT** allowed to modify images' original width-to-height ratios. For example, if the width increases by 25%, the height must also increase by 25% and vice versa.

5. At all times, **ukraine.jpg** will be situated:
   a. **20 pixels** away from the **Left Border of the Web Browser Screen**, AND
   b. **20 pixels** above the **Bottom Border of the "beach" image**

6. At all times, **russia.jpg** will be situated:
   a. **20 pixels** away from the **Right Border of the Web Browser Screen**, AND
   b. **20 pixels** above the **Bottom Border of the "beach" image**

7. As shown in **Test Case #4** where the **Web Browser Screen width** is **300 pixels**:
   a. BOTH **ukraine.jpg** and **russia.jpg** must appear **on TOP** of the **black rectangle** (and the white "World Peace" text).

## Q2. [Bootstrap, JavaScript, Axios] World Military Power [20 marks]

Given resources:

➔ Q2\

- ✏ slideshow.html
- ✏ slideshow.js
- ✏ table.html
- ✏ table.js
- 🚫 flags/* *(There are 5 JPG image files. Do NOT move them.)*
- 🚫 api/info.php *(API endpoint file. Do NOT move it.)*
- 🚫 axios/info.php *(Axios library file. Do NOT move it.)*
- 🚫 bootstrap/* *(Bootstrap CSS & JavaScript files. Do NOT move them.)*

In this question, you will use Bootstrap, JavaScript, and Axios to build a responsive and dynamic web page.

**IMPORTANT**

You are to **edit:**

1. **[Part A] `slideshow.html`** and **`slideshow.js` only**
2. **[Part B] `table.html`** and **`table.js` only**

Do **NOT** edit other files. Do **NOT** add additional files (it will **NOT** be included for grading purposes).

---

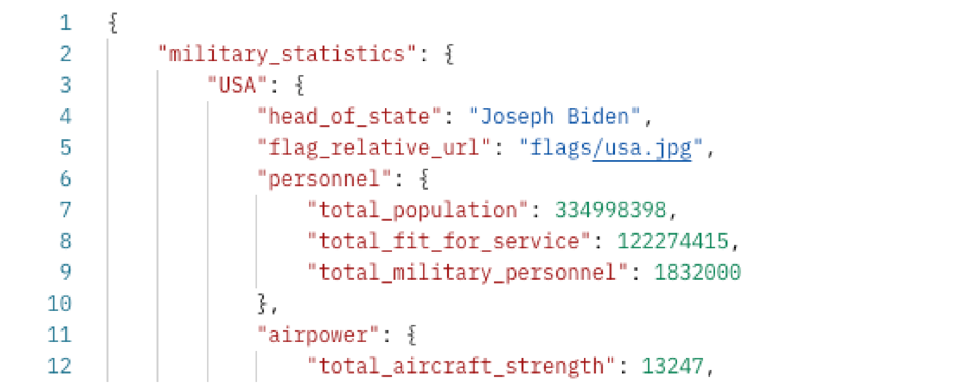
**API**

The **World Fire Power API** is **locally available**. Suppose that the location of your **JavaScript** file is **`http://localhost/MLT/Q2/slideshow.js`**. There are **2 ways** to access the **API**'s only **endpoint**:

| | |
|---|---|
| **Relative URL** | `api/info.php` |
| **Absolute URL** | `http://localhost/MLT/Q2/api/info.php` |

Using the **Absolute URL** (as shown above), please **TEST** the API's endpoint using **Postman**.

1. Open **Postman** and click **+** sign to open a **new request** tab.
2. Use **GET** method
3. Copy or type in the **Absolute URL**
4. Click **Send** button
5. Below, see under **Body → Pretty**. You should be able to see the following:



```
{
    "military_statistics": {
        "USA": {
            "head_of_state": "Joseph Biden",
            "flag_relative_url": "flags/usa.jpg",
            "personnel": {
                "total_population": 334998398,
                "total_fit_for_service": 122274415,
                "total_military_personnel": 1832000
            },
            "airpower": {
                "total_aircraft_strength": 13247,
```

*… **more data below…***

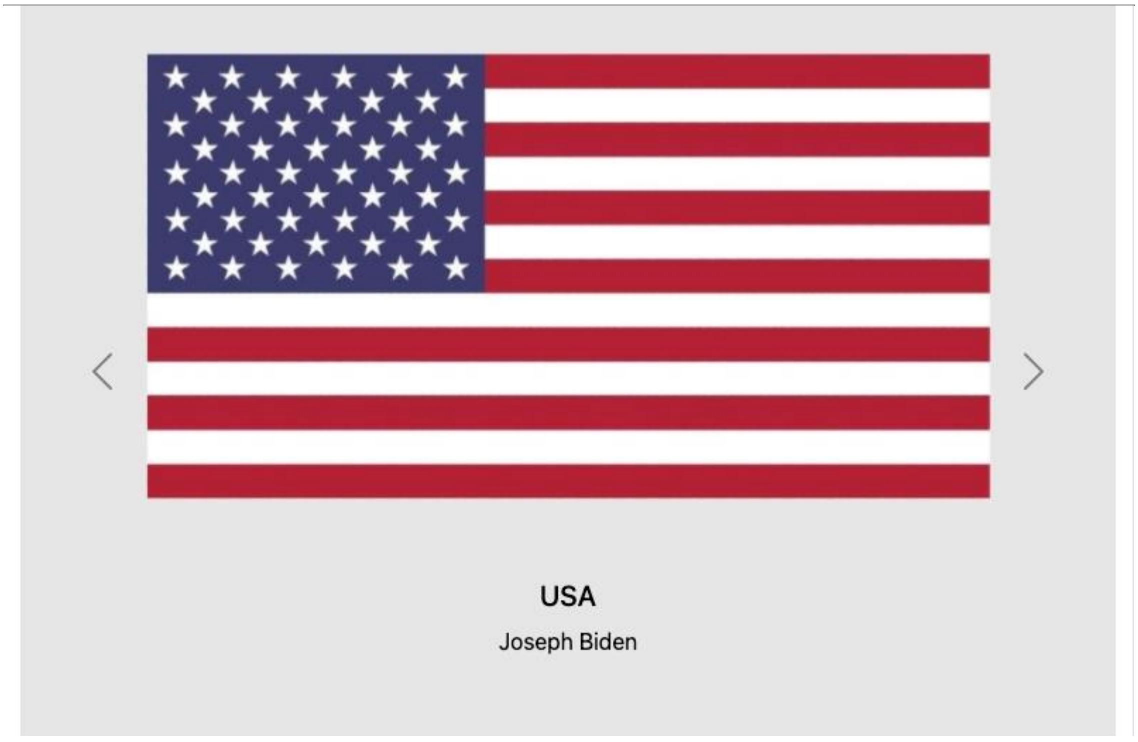
### Part A: Slide Show of Countries [10 marks]

Edit `slideshow.html` and `slideshow.js` to display **countries'** names, flags, and heads of state using **Bootstrap Carousel**.

Your **JavaScript code** must call the **API** *(see the previous page for more information)* to retrieve necessary information to populate `slideshow.html`.

1. Your **JavaScript code** must perform **Document Object Model (DOM) manipulation** to dynamically render the retrieved data inside `slideshow.html`.
2. If you **hardcode** (e.g. manually type/key in the data) into `slideshow.html` and/or `slideshow.js`, your solution will be marked as **INCORRECT** and you will receive ZERO (0) marks for **Part A**.
3. Your graders will use the same API but with **other countries' data** returned by the API.
    a. Your solution must correctly display the data **dynamically**.

When done correctly, the **Web Browser** will display `slideshow.html` as shown below:

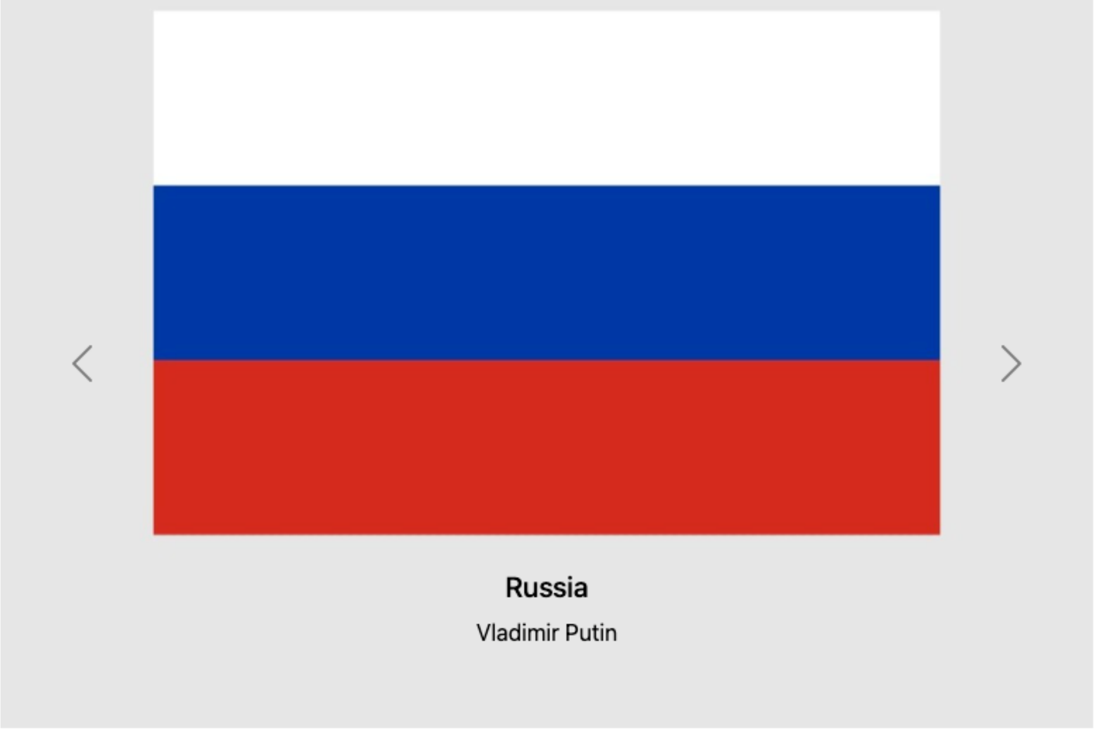


USA

Joseph Biden

**Web Browser Screen Width 800 pixels**

When `slideshow.html` loads for the first time in a web browser, the above is the **1st slide** shown *(before the slide show proceeds to the next slide "Russia")*.

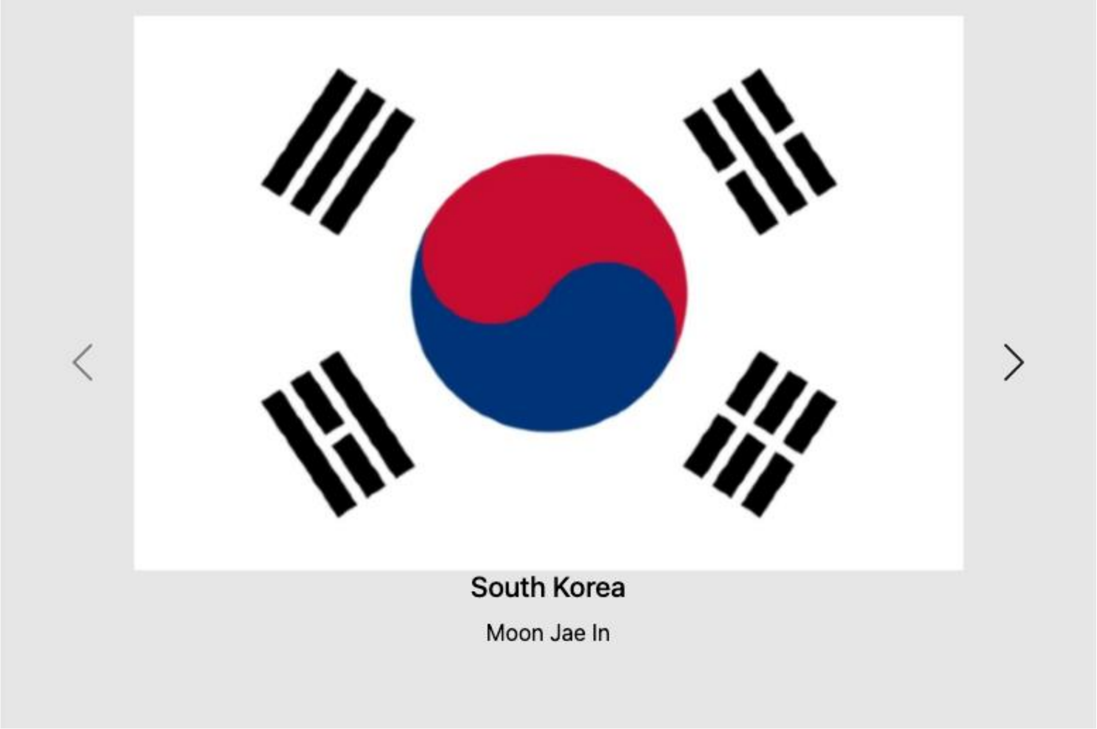


Russia

Vladimir Putin

**Web Browser Screen Width 800 pixels**

This is the **2nd slide** shown *(right after the "USA" slide)*.

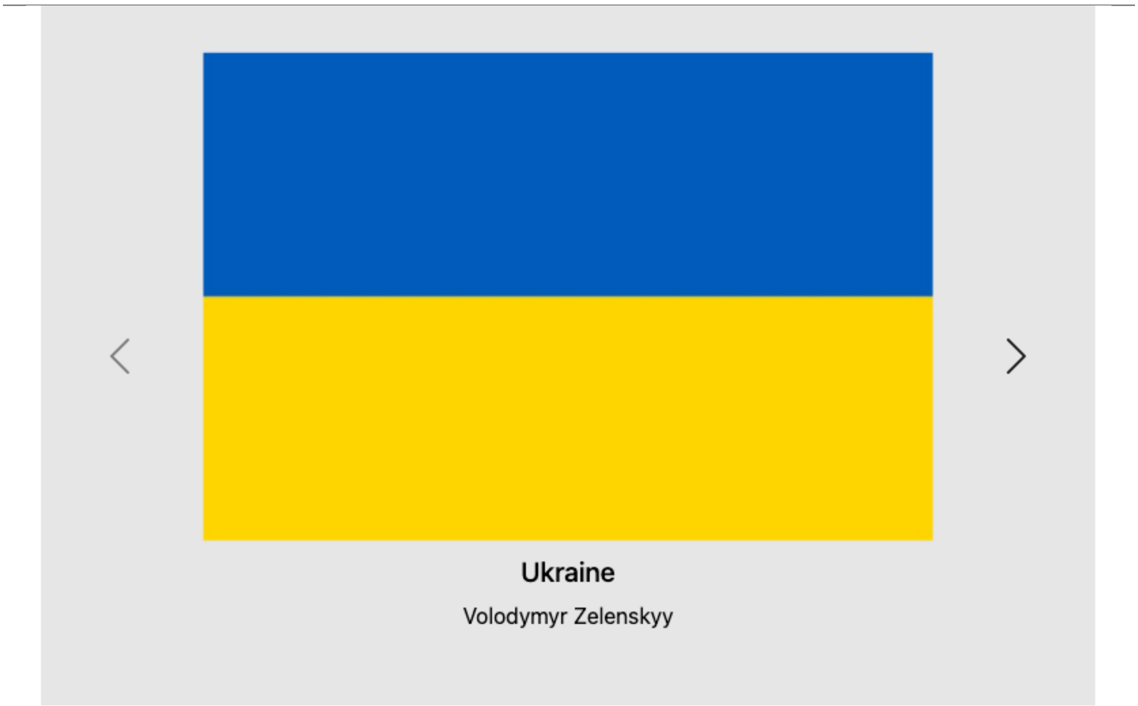


South Korea

Moon Jae In

**Web Browser Screen Width 800 pixels**

This is the **3rd slide** shown *(right after the "Russia" slide)*.

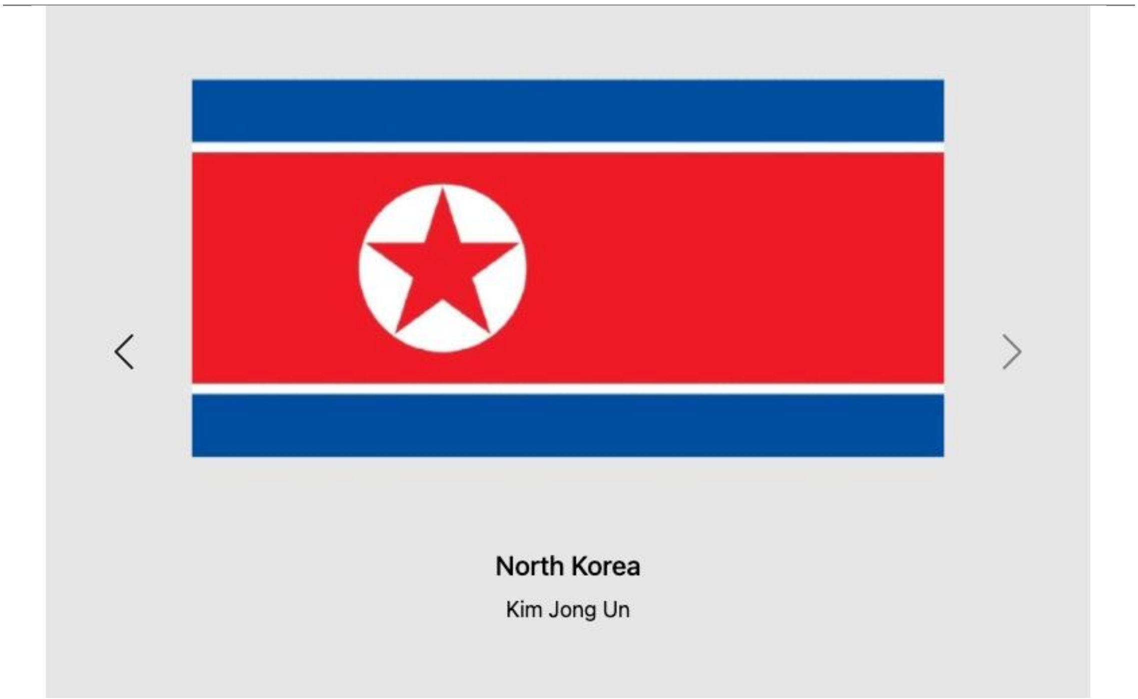


Ukraine

Volodymyr Zelenskyy

**Web Browser Screen Width 800 pixels**

This is the **4th slide** shown *(right after the "South Korea" slide)*.



North Korea

Kim Jong Un

**Web Browser Screen Width 800 pixels**

This is the **5th slide** shown *(right after the "Ukraine" slide)*.

Please ensure that the following are implemented correctly:

**NOTE:** For this question, please only use **Web Browser Screen width 800 pixels**.

1. The **images** of **countries' flags** are found in **Q2 → flags** *(folder)*.
    a. You are **NOT** allowed to edit the image files.
    b. You are **NOT** allowed to modify the images' original width-to-height ratios. For example, if the width shrinks by 30%, the height must also shrink by 30% and vice versa.

2. Between **slides**, there must be **exactly 2 seconds** of **delay**.

3. The **slides** must **automatically cycle** without the user's manual intervention.

4. In addition to the requirement #3, should the user choose to do so, he/she should be able to **manually** move to the **previous** or **next** slide using the **left/right** arrows *(the HTML code for this is already provided in the HTML file)*.

---

## Part B: Table of Countries & Military Statistics [10 marks]

Edit `table.html` and `table.js` to display **countries'** statistics using **Bootstrap Table**.

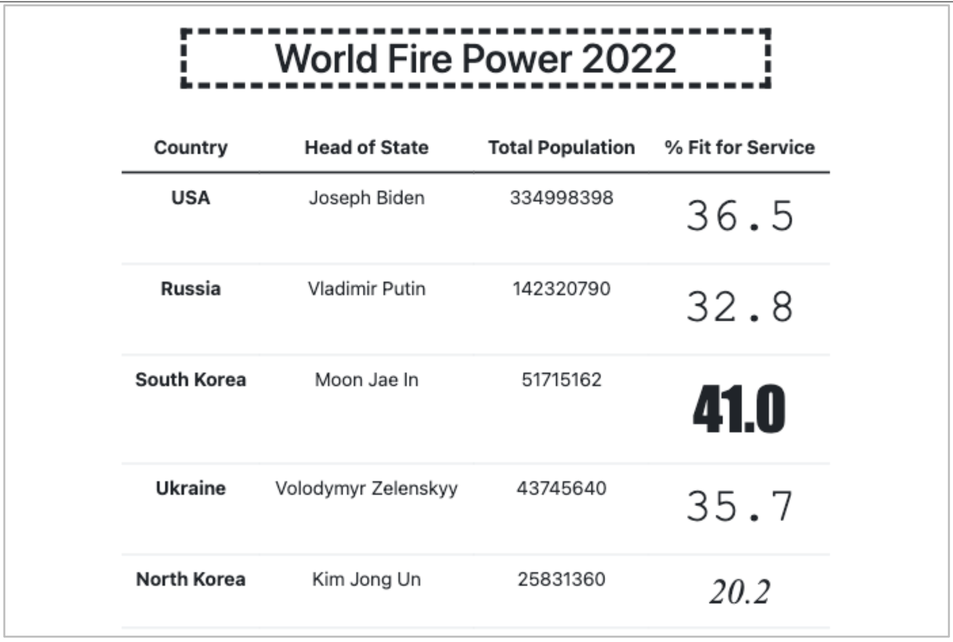
For this question, please **write Internal CSS code** inside `table.html` file.

- Do **NOT** create an external CSS file. If you do so, it will **NOT** be considered for grading purposes.

Your **JavaScript code** must call the **API** *(see Page 8 for more information)* to retrieve necessary information.

1. Your **JavaScript code** must perform **Document Object Model (DOM) manipulation** to dynamically render the retrieved data inside `table.html`.
2. If you **hardcode** (e.g. manually type/key in the data) into `table.html` and/or `table.js`, your solution will be marked as **INCORRECT** and you will receive ZERO (0) marks for **Part B**.
3. Your graders will use the same API but with **other countries' data** returned by the API.
    a. Your solution must correctly display the data **dynamically**.

When done correctly, the **Web Browser** will display `table.html` as shown below: *(see in the next page)*



World Fire Power 2022

| Country | Head of State | Total Population | % Fit for Service |
|---|---|---|---|
| USA | Joseph Biden | 334998398 | 36.5 |
| Russia | Vladimir Putin | 142320790 | 32.8 |
| South Korea | Moon Jae In | 51715162 | 41.0 |
| Ukraine | Volodymyr Zelenskyy | 43745640 | 35.7 |
| North Korea | Kim Jong Un | 25831360 | 20.2 |

**Web Browser Screen Width 800 pixels**

Please ensure that the following are implemented correctly:

**NOTE:** For this question, please use **Web Browser Screen width between 700 pixels** and **900 pixels**. Vary the **Web Browser Screen width** to test whether the **H1 Heading** and the **Table** are **horizontally centered** *(see below for instructions)*.

1. The **H1 Heading** (containing the text “World Fire Power 2022”) will fulfill all of the following conditions:
    a. Be situated **20 pixels** below the **Top Border of the Web Browser Screen**
    b. There is a **30 pixels** vertical distance between this **H1 Heading** and the **table below**
    c. Has a **fixed width** of **500 pixels**
    d. Be **horizontally centered** in the **Web Browser Screen**
    e. Has a **dashed border** whose **width** is **5 pixels**

2. The **Table** will fulfill all of the following conditions:
    a. Has a **fixed width** of **600 pixels**
    b. The **entire table** will be **horizontally centered** in the **Web Browser Screen**
    c. All the **cells** inside the **Table** will be **horizontally centered**
    d. The last table column **“% Fit for Service**” for **each country** is a **Percentage value** derived from: **total_fit_for_service** divided by **total_population** *(retrieved from the API response)*.
        i. The displayed value is in **Percentage** (rounded to the 1st decimal place).
        ii. Apply the **corresponding Internal CSS *(class selector)*** to this table column based on the following rules. Note that the **Internal CSS** code is available inside **`table.html`** file.

| % Fit for Service | Class Selector Name |
|---|---|
| `X < 30 (Below 30)` | `.font-okay` |
| `30 <= X < 40` | `.font-omg` |
| `40 and Above` | `.font-holy-moly` |