Hang Biao Li

Project 2: Second Art Gray Paper

1.
```
writeText();

function writeText() {
    textEl.innerText = text.slice(0, idx);

    idx++;

    if (idx > text.length) {
        //pause loop  for 5 second after full text show
        idx = 0;
        setTimeout(() => {
            writeText();
        }, 5000);
        return;
    }

    setTimeout(writeText, 215);
}
```

From the Udemy course. The code is to be able to loop each letter by letter until the full sentence is loaded into the screen. Then after 5 seconds, it will start loop again.

There was originally a problem I encountered where I originally used set timeout twice in the wrong way, if it gets to full text it will stop for 5 seconds. But instead, it results in looping faster and faster as it goes.

2.
```
//pass value to another page
const buttonbtn = document.querySelectorAll('.submit-button');

buttonbtn.forEach(button => {
    button.addEventListener('click', function () {
        const imgSrc = this.closest('.flex-container').querySelector('.art-image').src;
        const imgName = this.closest('.flex-container').querySelector('.art-image').alt;

        // Encode data
        const encodedImgSrc = encodeURIComponent(imgSrc);
        const encodedImgName = encodeURIComponent(imgName);

        // Redirect to ArtView.html with the encoded parameters
        window.location.href = `ArtView.html?imgSrc=${encodedImgSrc}&imageName=${encodedImgName}`;
    });
});
```

Sadly my original plan was to pass the value through the node or save using JSON, but there was so many bug and error that I can't figure out how to fix them. So instead of using node, I pass the value instead which has a similar effect to what I originally planned to do.

Sadly I wasn't able to use node or a way to save data if user comments due to errors which I originally planned for ArtView.html

3.
```
document.addEventListener('DOMContentLoaded', function () {
    const urlParams = new URLSearchParams(window.location.search);
    const imgSrc = urlParams.get('imgSrc');
    const imageName = urlParams.get('imageName');

    // Set the image source and image name
    document.getElementById('img').src = decodeURIComponent(imgSrc);
    document.getElementById('artDetail').innerText = `Image Name: ${decodeURIComponent(imageName)}`;
});
```

It gets the data and loads the image into the ArtView

4.
```
const title = document.getElementById('title')
const excerpt = document.getElementById('excerpt')
const detail = document.getElementById('presenter')
const name = document.getElementById('date')

const animated_bgs = document.querySelectorAll('.animated-bg')
const animated_bg_texts = document.querySelectorAll('.animated-bg-text')

setTimeout(getData, 1000)

function getData() {
  title.innerHTML = 'Art Gallery Project'
  excerpt.innerHTML =
    'The idea of this project is to allow user to comment and view on my art work.'
  detail.innerHTML = 'Project present to you by:'
  name.innerHTML = 'Hang Biao Li'

  animated_bgs.forEach((bg) => bg.classList.remove('animated-bg'))
  animated_bg_texts.forEach((bg) => bg.classList.remove('animated-bg-text'))
}
```

From the Udemy course idea. Able to present a card in contract to showcase a sort of profile.

5.
```
//udemy course for button
const buttons = document.querySelectorAll('.submit-button');

buttons.forEach(button => {
    button.addEventListener('mouseenter', function (e) {
        const x = e.pageX;
        const y = e.pageY;

        const buttonTop = e.target.offsetTop;
        const buttonLeft = e.target.offsetLeft;

        const xInside = x - buttonLeft;
        const yInside = y - buttonTop;

        const circle = document.createElement('span');
        circle.classList.add('circle');
        circle.style.top = yInside + 'px';
        circle.style.left = xInside + 'px';

        this.appendChild(circle);

        setTimeout(() => circle.remove(), 500);
    });
});
```

This is another Udemy course where if I hover over my mouse to the button it will have a cool circle effect which I think is very cool.

6.
```
<div class ="arts">
  <div id="art1" class="flex-container">
    <img id = "img" src="cloud.png" alt="cloud" class="art-image">
    <div class="flex-bottom">
      <input type="text" placeholder="cloud" class="image-name">
      <button id = "submitBtn" class="submit-button">Submit</button>
    </div>
  </div>

  <div id="art2" class="flex-container">
    <img id = "img" src="person.png" alt="person" class="art-image">
    <div class="flex-bottom">
      <input type="text" placeholder="person" class="image-name">
      <button id = "submitBtn" class="submit-button">Submit</button>
    </div>
  </div>

  <div id="art3" class="flex-container">
    <img id = "img" src="rock.png" alt="rock" class="art-image">
    <div class="flex-bottom">
      <input type="text" placeholder="rock" class="image-name">
      <button id = "submitBtn" class="submit-button">Submit</button>
    </div>
  </div>
```

Those are the div classes for each image placed. Originally I messed up by horrible ordering which resulted in unorganized photo placement as well as the placeholder not showing up.

7.
```css
.flex-container {
  display: flex;
  flex-direction: column;
  align-items: center;
  text-align: center;
  width: 30%;
  margin: 10px;
  box-sizing: border-box;
  position: relative;
  background-color: #f2f2f2;
  padding: 15px;
}
```

The best way I feel to organize the photo was to use a flex box. The previous way I how order ended up creating an uneven mess.

8.
```html
<div class="FlexContainer">
    <div class="ImgLoad">
      <img  id = "img" src="" alt="Art Image" class="art-image">
    </div>

    <div class="Details">
      <p id = "artDetail">Art Details go here...</p>
    </div>
```

There was originally another div class for loading save comments as well as a div class for submitting comments. However due to multiple errors and mistakes in trying to get node.js and json work. I ended up deleting it.

9.
```html
<div class="container">
  <div class="panel active" style="background-image: url('rock.png')">
    <h3>rock/h3>
  </div>
  <div class="panel active" style="background-image: url('person.png')">
    <h3>person</h3>
  </div>
  <div class="panel active" style="background-image: url('cloud.png')">
    <h3>cloud</h3>
  </div>
```

One of the biggest changes to this project is what I originally planned to do before calling it “Second Art” which was to upload free images from other websites. But due to unsure if that will cause copy right issue, I ended up deciding on drawing on my own.

10. 

```
button .circle {
  position: absolute;
  background-color: #fff;
  width: 100px;
  height: 100px;
  border-radius: 50%;
  transform: translate(-100%, -100%) scale(2);
  animation: scale 0.8s ease-out;
}
```

The idea is from an Udemy project that gives a cycle effect if I click the button. I found it very cool so I used it in this project by modifying the CSS effect, instead of the effect only on the button. It will now have an effect outside of the button. Also, the original one requires a click to show the effect which I changed to hover over by mouse sensor.