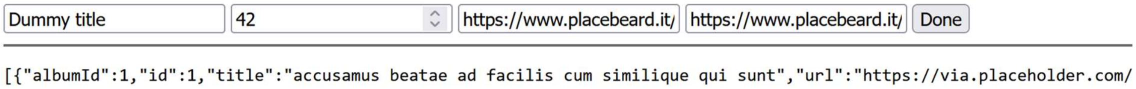
Web Front End Lab 3 (ECMAScript with 'fetch')

This lab gets you to write ECMAScript to fetch some JSON data and then show it on the page

1. Return to your 'lab2.html' or open the solution file 'lab2.html'
2. Save it as 'lab3.html'
3. Change the <title> to say 'Fetching API Data'
4. Also check the link to the stylesheet is ok (might need changing to 'lab3.css')
5. Create a file called 'lab3.js'
6. Return to 'lab3.html' and include a <script> tag, with the 'src' attribute pointing to 'lab3.js'
7. Also add a <pre> tag wherever you would like the raw data to appear
8. Give the <pre> tag an id
9. Find the <button> tag (inside the <form>) and give it an id
10. Return to 'lab3.js' and set a constant that references the id of the <button>
11. If you made your form fields ‘required’, then set each field to a default value to save you having to type every time you try it out
12. Now we need an event handler function and a listener:
    a. Write a function called 'getPhotos'
    b. Inside the function, prevent the <form> from submitting:
       ```
       e.preventDefault()
       ```
       Alternatively, just remove the <form tag>
    c. Write code that will fetch JSON from https://jsonplaceholder.typicode.com/photos
    d. Use .then to handle the returned data, then show the returned data in the <pre> tag, something like this:



Dummy title | 42 | https://www.placebeard.it/ | https://www.placebeard.it/ | Done

```
[{"albumId":1,"id":1,"title":"accusamus beatae ad facilis cum similique qui sunt","url":"https://via.placeholder.com/
```

13. Remember to add your function as a 'click' event listener on the button
14. If you have time, explore ways to show the returned data on the page
    a. Maybe show a list of photo titles
    b. Or show the returned images (use the Url in <img> tags with alt text)
    c. It might get complex, but you could try to add rows to your table for each returned image