```
//The sendEmail.js file was created utilising the following links to assist with the
2
         https://dashboard.emailjs.com/admin
3
         https://dashboard.emailjs.com/admin/integration/browser
     // https://www.emailjs.com/docs/sdk/send/
5
6
7
     //Creating a variable to associate the button
     let btn = document.getElementById('send-modal-button');
8
     //This section of code listens to the events happening within the contact form and \ell //the text inside the button from 'Send' to 'Sending...'
9
10
11
     document.getElementById('contactForm')
          .addEventListener('submit', function (event) {
12
13
          event.preventDefault();
14
15
          //Informing the user that the email is sending.
16
         btn.textContent = 'Sending...';
17
          // Variables which stores the details which connect the service and template as:
18
19
          let serviceID = 'personal-gmail';
20
          let templateID = 'middle_east';
21
22
23
          /* The two variables mentioned above are used within the template parameters. TI
          are used in the email template that has been set up within EmailJS along with s
24
         of the contact form within the HTML file.*/
25
26
          emailjs.send(serviceID, templateID, {
              "from_fullname": this.from_fullname.value,
27
              "from_useremail": this.from_useremail.value,
28
29
              "country_requested": this.country_requested.value,
              "question_asked": this.question_asked.value,
30
         })
31
32
         /*The object which would contain the text which would be changed if successful (
33
         within the console log within the browser and the status in a text format as we
34
         would notify the user once the message has been sent. */
35
              .then((response) => {
btn.textContent = 'Sent!'
36
37
              console.log("Message Sent!", response.statusText);
38
              alert('Message sent. Someone will reach out to you shortly.');
39
40
          /* The object which would contain the text which would be changed if unsuccessfu
41
         within the console log within the browser and the status in a text format as we
42
         would notify the user if the message was not sent. */
43
          }, (error) => {
              btn.textContent = 'Error!';
44
45
              console.log("Message did not send!", response.statusText);
46
              alert(`Unfortunately, the message was not sent. Please trying again later.``
47
         });
48
     });
```

49



version 2.12.0
There are 3 functions in this file (https://github.com/jshint/jFunction with the largest signate

median is 1, About (/about)

Largest function has 5 statemer Documentation (/docs) The most complex function has of 1 while the median is 1.

Contribute (/contribute)

Six^BWarmhings

- 7 'let' is available in ES6 (use extensions (use moz).
- 19 'let' is available in ES6 (use extensions (use moz).
- 20 'let' is available in ES6 (use extensions (use moz).
- 35 'arrow function syntax (=>) 'esversion: 6').
- 43 'arrow function syntax (=>) 'esversion: 6').
- 46 'template literal syntax' is o 'esversion: 6').

Two undefined variables

- 25 emailis
- 45 response