

## ПРИМЕР РАЗМЕЩЕНИЯ СПЕЦИАЛЬНЫХ ФРАГМЕНТОВ КОДА СЧЕТЧИКА ОБРАЩЕНИЙ

Ниже приведен пример кода, управляющего функциональностью электронной формы, до размещения специальных фрагментов кода счетчика обращений.

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <title>Страница с формой</title>
7 </head>
8
9 <body>
10  <button id="button_show_form">Обратная связь</button>
11  <div class="popup-inline-text-content" id="feedback">
12    <h3>Обратная связь</h3>
13    <form enctype="multipart/form-data" method="post" action="/ajax/feedback/">
14      <p class="">Сообщение:</p>
15      <textarea id="feedback-text" name="user_f_1"></textarea>
16      <button class="icon arr-r" type="submit">Отправить</button>
17    </form>
18  </div>
19 </body>
20
21 </html>
22 |
```

Далее приведен пример размещения специальных фрагментов кода счетчика обращений внутри кода (выделено красным), управляющего функциональностью электронной формы, при помощи JavaScript кода.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>Страница с формой</title>
6 </head>
7 <body>
8 <button id="button_show_form">Обратная связь</button>
9 <div class="popup-inline-text-content" id="feedback">
10 <h3>Обратная связь</h3>
11 <form enctype="multipart/form-data" method="post" action="/ajax/feedback/">
12 <p>Сообщение:</p>
13 <textarea id="feedback-text" name="user_f_1"></textarea>
14 <br>
15 <button class="icon arr-r" type="submit"><span>Отправить</span></button>
16 </form>
17 </div>
18 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js" crossorigin="anonymous"></script>
19 <script>
20 window.sputnikCounter = window.sputnikCounter || {};
21 window.sputnikCounter.events = window.sputnikCounter.events || [];
22 var $feedback = $('#feedback');
23 var $button = $('#button_show_form');
24 $feedback.hide(); // Скрываем форму по умолчанию
25
26 $button.on('click', function() {
27 // Показываем форму пользователю
28 window.sputnikCounter.events.push({
29 form: "open"
30 });
31 $feedback.show();
32 })
33 $feedback.find('form').on('submit', function(event) {
34 // Добавляем обработчик события submit
35 event.preventDefault();
36 window.sputnikCounter.events.push({
37 form: "submit"
38 });
39 $.ajax({
40 url: $(this).attr('action'),
41 success: function() {
42 // Обработчик успешного запроса
43 window.sputnikCounter.events.push({
44 form: "success"
45 });
46 otherYourSuccessCode.apply(this, arguments);
47 },
48 error: function() {
49 // Обработчик неуспешного запроса
50 window.sputnikCounter.events.push({
51 form: "error"
52 });
53 otherYourFailureCode.apply(this, arguments);
54 }
55 });
56 })
57 </script>
58
59 <script type="text/javascript">
60 (function(d, t, p) {
61 var j = d.createElement(t);j.async = true;j.type = "text/javascript";
62 j.src = ("https:" == p ? "https:" : "http:") + "//stat.sputnik.ru/cnt.js";
63 var s = d.getElementsByTagName(t)[0];s.parentNode.insertBefore(j, s);
64 })(document, "script", document.location.protocol);
65 </script>
66 </body>
67 </html>
```

Обязательно, перед закрывающим тегом `</body>` должен быть добавлен код счетчика обращений, как показано на рисунке выше (строки 59-65).

```
<script>
  (function(d, t, p) {
    var j = d.createElement(t); j.async = true; j.type = "text/javascript";
    j.src = ("https:" == p ? "https:" : "http:") + "//stat.sputnik.ru/cnt.js";
    var s = d.getElementsByTagName(t)[0]; s.parentNode.insertBefore(j, s);
  })
  (document, "script", document.location.protocol);
</script>
```

Перед первым использованием метода `window.sputnikCounter.events.push` требуется провести инициализацию переменных, как показано на рисунке выше (строки 20-21).

```
window.sputnikCounter = window.sputnikCounter || { };
window.sputnikCounter.events = window.sputnikCounter.events || [ ];
```