

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федерально автономное образовательное учреждение высшего образования  
«Севастопольский государственный университет»  
кафедра Информационных систем

Куркчи Ариф Эрнестович

Институт информационных технологий и управления в технических системах  
курс 3 группа ИС/б-31-о  
09.03.02 Информационные системы и технологии (уровень бакалавриата)

ОТЧЕТ  
по лабораторной работе №2  
по дисциплине «Веб-технологии»  
на тему «Исследование возможностей программирования на стороне клиента.  
Основы языка JavaScript.»

Отметка о зачете \_\_\_\_\_ (дата)

Руководитель практикума

ст. преподаватель  
(должность)

\_\_\_\_\_  
(подпись)

А. Л. Овчинников  
(инициалы, фамилия)

## 1. ЦЕЛЬ РАБОТЫ

Изучить основы языка JavaScript и объектной модели браузера, приобрести практические навыки проверки HTML-форм с использованием JavaScript.

## 2. ПОСТАНОВКА ЗАДАЧИ

### Вариант №10

Модифицировать страницу «Фотоальбом» (использовать HTML- страницы, разработанные при выполнении предыдущей лабораторной работы), реализовав вывод таблицы, содержащей фото, с использованием операторов циклов. Значения имен файлов фото и подписей к фото предварительно разместить в массивах *fotos* и *titles*.

Модифицировать страницу «Мои интересы», реализовав вывод списков с использованием JavaScript-функции с переменным числом аргументов.

Добавить на страницах «Контакт» и «Тест по дисциплине “...”» функции проверки заполненности форм. В случае если какое-либо из полей формы осталось незаполненным при нажатии на кнопку отправить, вывести сообщение об ошибке и установить фокус на незаполненный элемент.

Добавить на странице «Контакт» текстовое поле «Телефон». И для полей “Фамилия Имя Отчество” и “Телефон” добавить функции специфической проверки значений. В случае если какое-либо из полей формы заполнено не верно, при нажатии на кнопку отправить, вывести сообщение об ошибке и установить фокус на неверно заполненный элемент. Формат правильных значений полей:

- Фамилия Имя Отчество – введено три слова, разделенные одним пробелом.
- Телефон – строка может состоять только из цифр; начинаться только с последовательности «+7» или «+3»; не содержит пробелов; количество цифр в строке от 9 до 11.

Добавить на странице «Тест по дисциплине “...”» функции специфической проверки значений полей в соответствии с вариантом задания, представленном в таблице 1. В случае невыполнения условия формировать сообщение об ошибке и установить фокус на неверно заполненный элемент.

10	1	Установлены менее трех переключателей
----	---	---------------------------------------

## 3. ИСХОДНЫЙ КОД

## main.js

```

1 function validate_form(form, field) {
2   for (var i = 0, len = form.elements.length; i < len; i++) {
3     var element = form.elements[i],
4         p = parent(element, 'form-group');
5     if (p) {
6       p.classList.remove('has-success', 'has-error');
7       if (!element.validity.valid) {
8         p.classList.add('has-error');
9         if (field == undefined || element.name == field) {
10            element.focus();
11          }
12        } else {
13          p.classList.add('has-success');
14        }
15      }
16    }
17  }
18
19  function parent(el, cls) {
20    while ((el = el.parentElement) && !el.classList.contains(cls));
21    return el;
22  }
23
24  function validate_field(field, regex) {
25    if(field) {
26      var value = field.value,
27          p = parent(field, 'form-group');
28      if(!regex.test(value)) {
29        p.classList.remove('has-success');
30        p.classList.add('has-error');
31      }
32    }
33  }

```

## contact.js

```

1 function validate(field) {
2   validate_form(document.form_contact, field);
3   validate_contacts(document.form_contact);
4 }
5
6 function validate_contacts(form) {
7   validate_field(form.name, /^[^\wа-яё]+\s+[\wа-яё]+\s*[\wа-яё]*$/i);
8   validate_field(form.tel, /^[\+]?(\?:380(\d){9}|7\s*(\d{3})?\s*(?:-\d){7})$/i);
9 }

```

## gallery.js

```

1 var gallery_elements = [];
2 for(var i=1;i<=15;i++) {
3   var fi = i <= 9 ? '0'+i : i;
4   gallery_elements.push({
5     id: i,
6     legend: 'Photo '+i,
7     alt: 'Photo '+i,
8     title: 'Photo '+i,
9     src: 'img/gallery/photo'+fi+'.jpeg'
10  });
11 }
12
13 function load_gallery() {
14   var cont = document.getElementById('gallery-cont'),
15       html = '';
16   for(var i=0;i<gallery_elements.length;i++) {
17     var el = gallery_elements[i];
18     html += '<div class="item"><span
19 class="item_title">'+el.legend+'</span></div>';
20   }
21   cont.innerHTML = html;

```

## interests.js

```

1 var list = {
2   type: 'ul',
3   elements: [
4     {
5       href: '#loved',
6       text: 'Любимое',
7       children: {
8         type: 'ol',
9         elements: [
10           {
11             href: '#loved__lists',
12             text: 'Списки'
13           },
14           {
15             href: '#loved__blocks',
16             text: 'Блоки'
17           }
18         ]
19       }
20     }
21   ]

```

```

19     }
20   },
21   {
22     href: '#hated',
23     text: 'Ненави́стное',
24     children: {
25       type: 'ol',
26       elements: [
27         {
28           href: '#hated__requirements',
29           text: 'Требования'
30         },
31         {
32           href: '#hated__tasks',
33           text: 'Задания'
34         }
35       ]
36     }
37   }
38 ];
39
40
41 function load_list() {
42   var cont = document.getElementById('links-cont'),
43       html = build_list(list);
44
45   cont.innerHTML = html;
46 }
47
48 function build_list(list) {
49   var elements = '';
50   for (var i = 0; i < list.elements.length; i++) {
51     var el = list.elements[i];
52     elements += '<li><a href="' + el.href + '">' + el.text + '</a></li>';
53     if (el.children) {
54       elements += build_list(el.children);
55     }
56   }
57   return '<' + list.type + '>' + elements + '</' + list.type + '>';
58 }

```

## test.js

```

1 function validate(field) {
2   validate_form(document.form_test, field);
3   validate_tests(document.form_test);
4 }
5
6 function validate_tests(form) {
7   validate_field(form.name, /^[\wa-яё]+\s+[\wa-яё]+\s*[\wa-яё]*$/i);
8
9   var count = 0,
10       max = 3,
11       list = [
12         form.interpolation_spline,
13         form.interpolation_lagrange,
14         form.interpolation_poly,
15         form.interpolation_dot
16       ],
17       p = parent(list[0], 'form-group');
18   for(var i = 0; i < list.length; i++) {
19     if(list[i].checked) {
20       count++;
21     }
22   }
23
24   if (count > max) {
25     p.classList.remove('has-success');
26     p.classList.add('has-error');
27   }
28 }

```

## ВЫВОДЫ

В ходе лабораторной работы были повторены основы языка JavaScript и объектной модели браузера. Написана простая валидация форм на страницах, созданных в первой лабораторной работе.