

# Лабораторная работа №3.

## РАБОТА С МАССИВАМИ. ОСНОВЫ ДИНАМИЧЕСКОЙ ВЁРСТКИ HTML-СТРАНИЦ

### Цели работы.

Изучить основы базового языка JavaScript. Научиться создавать сценарии JavaScript и внедрять их на web-страницы. Получить основные навыки работы с массивами и с данными разных типов, а также навыки динамической вёрстки статических веб-страниц.

### Задание.

В файле *countries.js* приведён массив *countries* с информацией о государствах мира и массив *about* с заголовочной информацией к массиву *countries*.

Необходимо разработать Web-страницу, содержащую сценарий JavaScript, позволяющий отобразить информацию о каждом государстве в приведённом ниже виде. Последовательность отображения информации о государстве задана массивом *about*.

Информация о каждом государстве должна выводиться в виде отдельной таблицы, для формирования которой необходимо разработать универсальную функцию *makeTableFromCountry* такую, что информацию обо всех государствах выведет код:

```
function makeTableFromCountry (country) {  
    tab("<table width=100%>");  
    /*  
        формирование таблицы, отображающей  
        информацию о городе country  
    */  
    tab("</table>");  
    return tab;  
}  
свояПеременная=countries.map(makeTableFromCountry);  
свояПеременная.forEach(function(info, i, arr) {  
    ЭлементДляВывода.innerHTML+=info+"<br>";  
});
```

Функция *makeTableFromCountry* должна осуществлять проверку типа элементов *country* с помощью стандартных функций и операторов JavaScript. Работоспособность этой функции должна сохраняться (т.е. информация о государствах должна корректно выводиться) в случае изменения порядка следования информации в любом из элементов *countries[i]*. Заданные массивы с информацией о государствах должны подключаться как содержимое внешнего файла *countries.js*.

Стилевое оформление – произвольное.

Способ отображения информации об одном государстве		Указания к отображению данных
<b>Италия - Рим</b>		Заголовок h1
Континент	Евразия	Вывод в виде таблицы шириной 75% от доступного пространства для вывода данных.
Часть света	Европа	
Разница во времени	-2	
Денежная единица	итальянская лира	
Географические объекты:	① Альпы	Вывод в виде таблицы в два столбца одинаковой ширины. Использовать функцию <i>map</i> в виде:

Литературные произведения:	<p>❷ Пальмария</p> <p>❸ Апеннинский полуостров</p> <p>❹ Комо</p> <p>❺ Тальяменто</p>	<pre>gObj=country[i].map(outArrayData);</pre> <p>Функция <i>outArrayData</i> должна «преобразовывать» элементы массива с названиями географических объектов <i>country[i]</i> в строки таблицы. Например, для <i>country[6][0]</i> для Италии это будет строка вида:</p> <pre>&lt;td&gt;❶ Альпы &lt;/td&gt;</pre> <p>Дальнейшая «сборка» строк в таблицу должна также осуществляться с помощью <i>map</i>:</p> <pre>таблица="&lt;table&gt;"; gObj.map(   function(item){таблица+=item;}); таблица="&lt;/table&gt;";</pre>
	<p>❶ Джованни Верга «Жизнь среди полей» (1880);</p> <p>❷ Анна Виванти «Цирцея» (1912);</p> <p>❸ Джузеппе Понтиджа «Рождённые дважды» (2000);</p>	<p>Вывод в виде аналогичном для «Географических объектов». Дополнить функцию <i>outArrayData</i> алгоритмом определения типа передаваемых ей данных: строка – для вывода географических объектов, или массивы – для вывода литературных произведений.</p>
	Исторические события	Обычный текст, полужирный
	<p><b>1882</b> - избирательная реформа</p> <p><b>1945</b> - антифашистское восстание, 18-28 апреля</p>	<p>Вывод в виде таблицы, количество столбцов которой зависит от количества исторических дат – полей объекта, содержащего исторические события в <i>country</i>.</p>

Результат работы разрабатываемого сценария для Италии и Японии:

## Италия - Рим

Континент	Евразия
Часть света	Европа
Разница во времени	-2
Денежная единица	итальянская лира

Географические объекты:

- ❶ Альпы
- ❷ Пальмария
- ❸ Апеннинский полуостров
- ❹ Комо
- ❺ Тальяменто

Литературные произведения:

- ❶ Джованни Верга «Жизнь среди полей» (1880);
- ❷ Анна Виванти «Цирцея» (1912);
- ❸ Джузеппе Понтиджа «Рождённые дважды» (2000);

### Исторические события

1882 - избирательная реформа	1945 - антифашистское восстание, 18-28 апреля
------------------------------	---

# Япония - Токио

Континент	Евразия
Часть света	Азия
Разница во времени	-6
Денежная единица	японская иена

Географические объекты: ❶Фудзияма  
❷Сакурадзима  
❸Абукума

Литературные произведения: ❶Сюсаку Эндо «Море и яд» (1958);  
❷Юдзо Ямамото «Профессор Цумура» (1919);  
❸Марико Хасиока «Аут» (1997);

## Исторические события

1872 - отменена сложного и строгого сословного деления	1904 - японский флот атаковал русскую эскадру в Порт-Артуре	1941 - атака на Пёрл-Харбор
--	---	-----------------------------

## Порядок выполнения.

1. Изучить теоретические сведения по массивам JavaScript. В частности изучить документацию на функции *map* и *foreach*.
2. Изучить теоретические сведения по типам данным JavaScript.
3. В соответствии с поставленной задачей разработать алгоритм её решения (см. «Указания к отображению данных» в таблице).
4. Написать сценарии JavaScript по разработанному в предыдущем пункте алгоритму.
5. Проверить правильность работы разработанных сценариев в различных браузерах.

## Содержание отчета.

1. Титульный лист.
2. Цель работы и задание.
3. Описание процесса вывода информации по каждому её виду (общая информация, географические объекты, литературные произведения, исторические события) и соответствующие алгоритмы.
4. ~~JavaScript-код решения.~~
5. Выводы по проделанной работе.

## Приложение.

### Справочная информация по используемым в работе функциям.

Функция	Интернет-ресурс
Справка по работе с массивами	<a href="https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Array">developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Array</a>
Определение типа значения	<a href="https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Operators/typeof">developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Operators/typeof</a> <a href="https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Array/isArray">developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Array/isArray</a> <a href="https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Number/isInteger">developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Number/isInteger</a>
Функции для доступа к элементам html-документа:	
Доступ по идентификатору:	<a href="https://developer.mozilla.org/ru/docs/Web/API/Document/getElementById">developer.mozilla.org/ru/docs/Web/API/Document/getElementById</a>

<code>document.getElementById()</code>	<code>ById</code>
Доступ по названию дескриптора: <code>document.getElementsByTagName()</code>	<a href="https://developer.mozilla.org/ru/docs/Web/API/Element/getElementsByTagName">developer.mozilla.org/ru/docs/Web/API/Element/getElementsByTagName</a> и <a href="https://developer.mozilla.org/ru/docs/Web/API/Document/getElementsByTagName">developer.mozilla.org/ru/docs/Web/API/Document/getElementsByTagName</a>
Доступ к элементу по его имени: <code>document.getElementsByName()</code>	<a href="https://developer.mozilla.org/en-US/docs/Web/API/Document/getElementsByTagName">developer.mozilla.org/en-US/docs/Web/API/Document/getElementsByTagName</a>
Доступ к элементам определённого класса: <code>document.getElementsByClassName()</code>	<a href="https://developer.mozilla.org/ru/docs/Web/API/Document/getElementsByTagName">developer.mozilla.org/ru/docs/Web/API/Document/getElementsByTagName</a>