

Лабораторная работа №3
Базовый JS веб-приложений.

Часть 1 (теоретическая):

Базовый синтаксис JS.

1. Изучить базовые типы данных JS. Понять отличие идентификаторов от зарезервированных слов. Изучить назначение литералов JS. Выражения и операторы, особенности применения и отличия. Иметь базовое понятие об инструкциях JS. В чем разница между использованием «var», «let» и «const»? Список примитивных типов в JavaScript. Что такое преобразование типов?

a. <https://kalapasa.ru/stati/javascript/identifikatory-i-zarezervirovannye-slova/>

b. https://developer.mozilla.org/ru/docs/Web/JavaScript/Guide/Grammar_and_types

c. <https://habr.com/ru/company/ruvds/blog/414377/>

d. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Statements>

2. Изучить понятие функции, что такое определение и вызов функции? Что такое аргументы и параметры функций, какие значения они могут принимать? Понять использование функции как объекта. Что такое замыкания? В каком случае и виде используются замыкания? Зачем оборачивать содержимое исходного JS-файла в блок функции? Какова причина помещения всего содержимого исходного файла JavaScript в книгу функций?

a. <https://learn.javascript.ru/function-basics>

b. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Guide/Functions>

c. <https://metanit.com/web/javascript/4.8.php>

d. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Closures>

3. Изучить особенности и принципы работы массивов в JS. Создание и инициализация массивов. Как можно организовать доступ к элементам массива? Что представляют и для чего нужны индексы массива? Понять основные методы массивов и варианты их применения. Что такое цикл событий? Как можно добавить элемент в начало и в конец массива? Как проверить, является ли объект массивом? В чем разница между null, undefined и undeclared? В чем разница между Array.prototype.forEach и Array.prototype.map?

a. <https://learn.javascript.ru/array>

b. <https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/>

[Global_Objects/Array](#)

ООП JS.

4. Изучить основные понятия объектно ориентированного программирования JS. Понять основное строение и назначение объектов. Овладеть навыками создания объектов. Понять назначение и отличия атрибутов свойств и объектов. Получить навыки моделирования классического ООП. Изучить типы конструкторов. Получить навыки использования конструкторов в JS. Понять что такое прототипы, для чего применяются, как и когда используются? Как динамически добавлять и удалять свойства в JavaScript? Объясните различие между Object.freeze() и const? Как можно клонировать объект?

a. <https://developer.mozilla.org/ru/docs/Learn/JavaScript/Objects>

b. <https://learn.javascript.ru/dom-attributes-and-properties>

c. <https://javascript.ru/tutorial/dom/attributes>

d. <https://developer.mozilla.org/ru/docs/Learn/JavaScript/Objects/>

[Object_prototypes](#)

Часть 2 (практическая)

1 Задан бесконечный цикл. С помощью двух отдельных вызовов функции `prompt()` вводит два числа. Числа сравниваются между собой, после чего пользователю отображается одна из трёх фраз: «числа равны», «первое число меньше», «второе число меньше». При введении пользователем не числовых значений, в зависимости от введённых данных пользователь может увидеть фразы «первый или второй ввод – не число». Выполнение скрипта прекращается при вводе пользователем в первом или втором вводе значения `q` без учёта регистра.

2. Дан многоквартирный жилой дом. У ждома присутствуют три характеристики: количество этажей (1-20), число подъездов (1-15), количество квартир на лестничной площадке (1-5). Пользовательский скрипт запрашивает характеристики, а также квартиры. Необходимо рассчитать и вывести на экран номер подъезда, в котором находится запрашиваемая квартира. Если пользователь вводит некорректные значения предусмотреть генерацию исключения, с выводом информации об ошибке на экран.

3. 22 марта 2022 был вторник. Написать скрипт который будет спрашивать у пользователя номер месяца в интервале "1..12", а так же число в этом месяце "1..31". Результатом работы скрипта является вывод на экран названия дня недели (например, «среда») для введённого пользователем числа и месяца.

4 Вам необходимо создать функцию, которая будет получать на вход два любых числа. При условии если оба введённых числа чётные - необходимо вернуть их произведение. При нечётных числах возвращаем сумму. В варианте при котором первое число чётное, а второе нечётное - возвращаем нечётное число, в обратном случае возвращаем чётное.

5 Необходимо создать функцию, которая будет возвращать сумму любого введённого количества чисел. Обратите внимание на то как должна вызываться данная функция:

```
result = sum(3)(1)(1); // результат 5
```

6 Реализуйте функцию `range()`, которая будет принимать на вход два аргумента: “старт” и “финиш” диапазона. Функция возвращать массив, содержащий все числа введённого диапазона, включая начальное и конечное. При этом необходимо предусмотреть необязательный аргумент – шаг для построения массива который может быть как положительным так и отрицательным. Если шаг явно не задан – он равен 1. Результатом работы `range(7, 4, -2)` будет `[7, 5]`.

7 Необходимо реализовать функцию создания матрицы, принимающую в качестве аргументов количество строк и количество столбцов. Эта функция должна отображать на экране матрицу, заполненную случайными числами в диапазоне от 0 до 100.

Так же необходимо написать другую функцию, которая будет выполнять суммирование двух различных по значению “первых матриц”.

8 Создайте функцию, которая будет объединять уникальные элементы всех одномерных массивов, переданных ей в качестве параметра. Если вводимый пользователем массив многомерный, то необходимо выдать пользователю сообщение о том что функция работает только с одномерными массивами и указать размерность введённого массива.

`MyUnionFunc([1, 5, 5, 3], [10, 5, 1, 10], [5, 1])` вернёт `[1, 3, 5, 10]`.

9 Создайте функцию, которая в качестве аргумента может принимать массив с уровнями вложенности элементов любой глубины, выполнит приведение его к “плоскому” виду. Если элементы массива не имеют вложенности, то функция просто выводит его на экран без изменений. `MyArrayFlattenFunc([1, [2], [[5], [3, [4]]]])` вернёт `[1, 2, 3, 4, 5]`

10 Реализовать функцию которая принимает на вход в качестве параметра массив или объект, подобный массиву, а так же параметр который отвечает за количество повторений. Возвращаемое функцией значение это массив, в котором удалены повторяющиеся значения.

```
var firstResult= MyFunctionUnique([1, 2, 1, 5, 1, 5, 1, 4], 3);  
// firstResult = [2, 4, 5]  
var secondResult = MyFunctionUnique("miru mir", 2);  
// secondResult = [" ", "u"];
```

Результатом выполнения практической части, должна быть сверстанная страничка на которой с применением форм и графического интерфейса будет реализовано решение всех задач лабораторной работы.

Варианты в данной лабораторной работе отсутствуют, но верстка и внешний вид лабораторной работы должен быть уникальным для каждого исполнителя.

Использование **фреймворков JS** не допускается, ограничений на использование CSS и HTML фреймворков нет.

Отчёт должен содержать титульный лист, текст разработанных программ на JS и внешний вид сверстанной странички.