

ПРАКТИЧЕСКАЯ РАБОТА «ОСНОВЫ JAVASCRIPT - 5»

Цель: научиться использовать функции в JavaScript, работать с локальными и внешними переменными.

Вначале убедитесь, что на вашем компьютере есть в папке **D:\Язык HTML\Занятие 10\Выполненные задания** папка под вашей фамилией (например, такая): **Иванов**.

Использование объявления функции

1. Наберите в **VSCode** код скрипта, который вызывает объявленную функцию 2 раза (когда мы указываем в скрипте ее имя), сохраните полученный результат под именем **index32.html** в папке **Язык HTML\Занятие 10** и откройте (запустите) его в любом браузере:

```
<!DOCTYPE HTML>
<html>
  <head>
    <script>
      function showMessage() {
        alert('Привет всем присутствующим от Виталия Валентиновича!');
      }
    </script>
  </head>
  <body>
    <script>
      showMessage();
      showMessage();
    </script>
  </body>
</html>
```

Использование локальных и внешних переменных в функциях

1. Наберите в **VSCode** код скрипта, в котором мы объявим внутри функции локальные переменные и попробуем использовать их, сохраните полученный результат под именем **index33.html** в папке **Язык HTML\Занятие 10** и откройте (запустите) его в любом браузере:

```
<!DOCTYPE HTML>
<html>
  <head> </head>
  <body>
    <script>
      function showMessage() {
        let message = 'Привет, я - Путин Владимир Владимирович!'; // локальная переменная
        alert(message);
      }
      showMessage(); // 'Привет, я - Путин Владимир Владимирович!'
      alert(message); // <-- не выведет значение, т.к. переменная видна только внутри
    </script>
  </body>
</html>
```

2. Наберите в **VSCode** код странички с использованием оператора цикла **for** внутри которого объявляются переменные **i** и **j**, сохраните полученный результат под именем **index34.html** и откройте (запустите) его в любом браузере:

```
<!DOCTYPE HTML>
<html>
  <head> </head>
  <body>
    <script>
      function count() {
        for (let i=0; i<3; i++) {
          let j = i * 2;

        }
        alert('i='+i); // i=3, на этом значении цикл остановился
        alert('j='+j); // j=4, последнее значение, на котором цикл сработал, было i=2
      }
      count();
    </script>
  </body>
</html>
```

3. Наберите в **VSCode** код странички и проверьте как изменится работа функции, если объявить локальные переменные внутри функции не в теле цикла, а вне его! Сохраните полученный результат под именем **index35.html** и откройте

(запустите) его в любом браузере:

```
<!DOCTYPE HTML>
<html>
  <head> </head>
  <body>
    <script>
      function count() {
        let i, j; // Мы передвинули объявления переменных let из цикла в начало
        for (i=0; i<3; i++) {
          j = i * 2;
        }
        alert('i='+i);
        alert('j='+j);
      }
      count();
    </script>
  </body>
</html>
```

4. Наберите в **VSCode** код странички, в которой функция внутри себя будет получать доступ и перезаписывать внешнюю переменную `userName`. Сохраните полученный результат под именем **index36.html** и откройте (запустите) его в любом браузере:

```
<!DOCTYPE HTML>
<html>
  <head> </head>
  <body>
    <script>
      let userName = 'Путин';
      function showMessage() {
        userName = 'Меркель'; // (1) присвоение во внешнюю переменную
        let message = 'Привет, я ' + userName;
        alert(message);
      }
      showMessage();
      alert(userName); // Меркель, значение внешней переменной изменено функцией
    </script>
  </body>
</html>
```

5. Наберите в **VSCode** код странички переданные в функцию значения для локальных переменных не повлияют на внешнюю переменную с таким же именем. Сохраните полученный результат под именем **index37.html** и откройте (запустите) его в любом браузере:

```
<!DOCTYPE HTML>
<html>
  <head> </head>
  <body>
    <script>
      function showMessage(from, text) {
        from = '***' + from + '***'; // (1), красиво оформили from
        alert(from + '\n\n' + text);
      }
      let from = 'Дмитрий Анатольевич Медведев', msg = 'Привет!'; // (2)
      showMessage(from, msg); // значения будут скопированы в параметры
      alert(from); // перезапись в строке (1) не повлияет на внешнюю переменную
    </script>
  </body>
</html>
```

6. Наберите в **VSCode** код странички в котором мы попробуем 2 варианта передачи параметров внутрь функции полный и неполный (не все параметры переданы!!!). Сохраните полученный результат под именем **index38.html** и откройте (запустите) его в любом браузере:

```
<!DOCTYPE HTML>
<html>
```

```

<head> </head>
<body>
  <script>
    function showMessage(from, text) {
      if (text == undefined) {
        text = 'текст не передан';
      }
      alert(from + ": " + text);
    }
    showMessage("Виталий Валентинович", "Привет!"); // Виталий Валентинович: Привет!
    showMessage("Виталий Валентинович"); // Виталий Валентинович: текст не передан
  </script>
</body>
</html>

```

Самостоятельная работа

Создайте в **VSCode** документ HTML, в котором средствами JavaScript:

Задание 1: написать скрипт, в котором вы создадите функцию `calcD`, которая будет возвращать дискриминант квадратного уравнения по формуле $b^2 - 4ac$.

```

<!DOCTYPE HTML>
<html>
  <head> </head>
  <body>
    <script>
      function calcD(a, b, c) {
        return b * b - 4 * a * c;
      }
      let discriminant = calcD(-4, 2, 1);
      alert('D='+discriminant);
    </script>
  </body>
</html>

```

Измените код так, чтобы он запрашивал три значения для переменных `a`, `b`, `c`. А потом выводил результат. Реализовать код с использованием стрелочной функции. Сохраните полученный результат под именем **index39.html**.

Задание 2: Заказчик попросил Вас помочь организовать лотерею для его клиентов! Причем клиентам очень важно, чтобы выбираемые числа были полностью случайные. Клиенты заказчика (участники лотереи) выбирают 3 числа, каждое из которых может быть одно- или двузначным в диапазоне от 0 до 99 (т.е. любое целое число меньше 100). Все билеты выглядят так: "18-7-92", или "46-60-4", или "25-81-98" и так далее.

Вы должны создать стрелочную функцию, которая бы возвращала 3 случайным образом отобранных числа (отделённых друг от друга дефисами и `log`-сообщениями с результатом в консоли). Сохраните полученный результат под именем **index40.html**.