# Лабораторная работа № 1 Введение в JavaScript

**Цель работы:** изучить основные типы данных и операторы языка JavaScript.

**Краткие теоретические сведения:**

Для написания сценариев на языке JavaScript достаточно любого текстового редактора, но лучше использовать специализированные, такие как Sublime Text, Notepad++, или интегрированные среды разработки Visual Studio Code или WebStorm.

Программы на JavaScript могут быть вставлены в любое место HTML-документа с помощью тега <script>. Для вставки сценария или библиотеки из внешнего файла используется атрибут src, в котором указывается либо абсолютный путь до скрипта от корня сайта, либо полный URL-адрес. Например:

<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>

Сценарий JavaScript выполняется с использованием инструкций, которые представляют собой синтаксические конструкции и команды, которые выполняют действия. Обычно каждую инструкцию пишут на новой строке, чтобы код было легче читать, и заканчивают точкой с запятой.

alert('Hello');

alert('World');

Для пояснения программного кода используются комментарии. В JavaScript комментарии могут находиться в любом месте скрипта и могут быть однострочными и многострочными.

// Этот комментарий занимает всю строку

/\*

Это - многострочный комментарий.

\*/

Любое веб-приложение обычно обрабатывает какую-либо информацию. Для хранения информации в JavaScript используются переменные и константы.

Для создания переменной в JavaScript используйте ключевое слово let:

let message;

message = 'Hello';

На практике для краткости часто совмещают объявление переменной и запись данных в одну строку.

В старых скриптах вы также можете найти другое ключевое слово: var вместо let. Ключевое слово var – почти то же самое, что и let. Оно объявляет переменную, но немного по-другому, «устаревшим» способом.

Для объявления константной, то есть, неизменяемой переменной, используется ключевое слово const:

const PI = 3.1415;

При названии переменных надо учитывать, что JavaScript является регистрозависимым языком, то есть в следующем коде объявлены две разные переменные:

var myIncome;

var MyIncome;

Значения, которые заносятся в переменную с использованием оператора присваивания ( = ), могут быть одного из 8 основных типов:

* **number** для любых чисел: целочисленных или чисел с плавающей точкой, целочисленные значения ограничены диапазоном ±253.
* **bigint** для целых чисел произвольной длины.
* **string** для строк. Строка может содержать один или больше символов, нет отдельного символьного типа.
* **boolean** для true/false.
* **null** для неизвестных значений – отдельный тип, имеющий одно значение null.
* **undefined** для неприсвоенных значений – отдельный тип, имеющий одно значение undefined.
* **object** для более сложных структур данных.
* **symbol** для уникальных идентификаторов.

Оператор typeof позволяет нам увидеть, какой тип данных сохранён в переменной:

* Имеет две формы: typeof x или typeof(x).
* Возвращает строку с именем типа. Например, "string".
* Для null возвращается "object" – это ошибка в языке, на самом деле это не объект.

Часто значения, которые заносятся в переменную, являются результатом вычисления выражения. Выражение представляет из себя набор операндов и операторов, выполняющих действия над операндами.

JavaScript является наследником языка С. Поэтому набор операторов и управляющих конструкций в языке точно такой же как и в С, с которым вы уже знакомы.

Следует, однако, отметить, что язык JavaScript является интерпретируемым и слаботипизированным, в отличии от компилируемых. Тип переменной определяется по введенному значнению или результату вычиления выражения, поэтому в процессе выполнения сценария тип переменной может меняться. Следует внимательно следить за преобразованием типов.

Для преобразования строк в числа в JavaScript предусмотрены встроенные функции parseInt() и parseFloat(). Функция parseInt(строка, основание) преобразует указанную в параметре строку в целое число в системе счисления по указанному основанию (8, 10 или 16). Если основание не указано, то предполагается 10, то есть десятеричная система счисления. Функция parseFloat(строка) преобразует указанную строку в число с плавающей разделительной (десятичной, основание) точкой:

parseInt("3.14") // результат= 3

parseInt("Вася") // результат = NaN, то есть не является числом

parseInt("0xFF", 16) // результат = 255

parseFloat("3.l4") // результат= 3.14

parseFloat("-7.875") // результат = -7.875

B JavaScript предусмотрены довольно скудные средства для ввода и вывода данных. Это вполне оправданно, поскольку JavaScript создавался в первую очередь как язык сценариев для веб-страниц. Можно воспользоваться тремя стандартными методами глобального объекта window для ввода и вывода данных: alert(), prompt(), confirm().

Метод alert() позволяет выводить диалоговое окно с заданным сообщением и кнопкой ОК:

alert(“сообщение”);

Cообщение представляет собой данные любого типа: последовательность символов, заключенную в кавычки, число (в кавычках или без них), переменную или выражение.

Метод confirm() позволяет вывести диалоговое окно с сообщением и двумя кнопками – ОК и Отмена (Cancel). В отличие от метода alert этот метод возвращает логическую величину, значение которой зависит от того, на какой из двух кнопок щелкнул пользователь. Если он щелкнул на кнопке ОК то возвращается значение true (истина), если же он щелкнул на кнопке Отмена, то возвращается значение false (ложь). Синтаксис применения метода confirm имеет следующий вид:

let answer = confirm(“сообщение”);

Метод prompt позволяет вывести на экран диалоговое окно с сообщением, а также с текстовым полем, в которое пользователь может ввести данные. Кроме того, в этом окне предусмотрены две кнопки: ОК и Отмена (Cancel). Данный метод принимает два параметра: сообщение и значение, которое должно появиться в текстовом поле ввода данных по умолчанию. Если пользователь щелкнет на кнопке ОК, то метод вернет содержимое поля ввода данных, если он щелкнет на кнопке Отмена, то возвращается логическое значение false (ложь).

let input = prompt(“сообщение”,”начальное\_значение”);

**Практическая часть:**

**Задание 1.** Работа с диалоговыми окнами.

1. В редакторе создайте файл z1\_1.html и поместите в него следующий код:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Первая программа на JavaScript</title>

</head>

<body>

<script>

"use strict";

alert("Мое первое диалоговое окно");

const answer = confirm(

"Хотите продолжить выполнения сценария на JavaScript"

);

if (answer) {

const name = prompt("Введите ваше имя", "");

alert(name + ", y вас уже начинает получаться!!!");

} else alert("Жаль, можно было бы еще поработать!");

</script>

</body>

</html>

2. Откройте этот файл в браузере и просмотрите результат.

**Задание 2.** Вычисление арифметических выражений.

1. Создайте файл z1\_2.html и поместите в него следующий код:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Вычисление арифметических выражений</title>

</head>

<body>

<script>

"use strict";

const x = parseInt(prompt("Введите значение х", ""));

let a = x \* x - 7 \* x + 10;

let b = x \* x - 8 \* x + 12;

let c = a / b;

alert("c = " + c);

</script>

</body>

</html>

2. Просмотрите в браузере результаты работы скрипта.

**Задание 3.** Вычисление площади и периметра правильного n-угольника, описанного около окружности радиуса R.

1. Cоздайте файл z1\_3.html и поместите в него следующий код:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Линейные алгоритмы</title>

</head>

<body>

<script>

"use strict";

let n = parseInt(prompt("Введите количество углов n", ""));

let r = parseInt(prompt("Введите радиус r", ""));

let a = 2 \* r \* Math.tan(Math.PI / n);

let p = a \* n;

let s = (1 / 2) \* n \* a \* r;

alert("Площадь = " + s.toFixed(2));

alert("Периметр = " + p.toFixed(2));

</script>

</body>

</html>

2. Просмотрите в браузере результаты работы скрипта.

**Задание 4.** Формирование динамических страниц с использованием метода write объекта document.

1. В текстовом редакторе создайте файл z1\_4.html и поместите в него следующий код:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Динамически сформированная страница</title>

</head>

<body>

<script>

"use strict";

document.write("Начало сформированной страницы");

document.write("<h1> Заголовок первого уровня</h1>");

document.write(

'<p style="text-align:center;font-size:18px;color:red;">Изменение размера и цвета шрифта'

);

document.write("<p>Конец формирования страницы, содержащей сценарий");

</script>

</body>

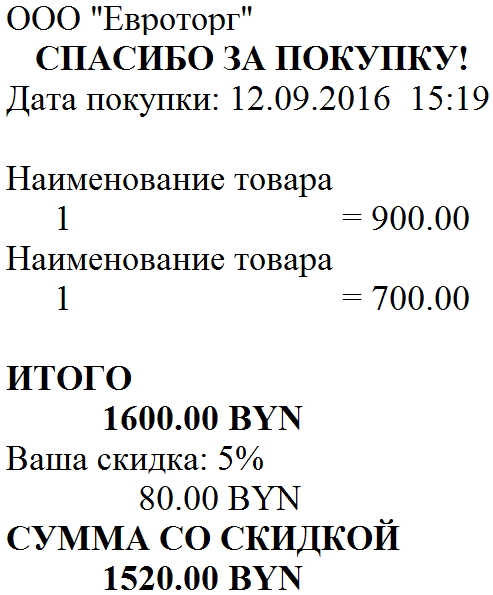
</html>

2. Просмотрите в браузере результаты работы скрипта.

**Индивидуальные задания**

**Во всех скриптах, в заголовке окна браузера должны быть ваши фамилия и имя!!!**

**Задание 1.** Вы совершаете покупку в магазинах евроторга. Необходимо написать сценарий, в котором с использованием метода prompt вводится общая сумма покупки, вводится скидка в размере от 1 до 5 %. В окне документа должны быть отображены: общая стоимость покупки, скидка в рублях и стоимость покупки со скидкой. Вывод оформить как кассовый чек.



**Задание 2.** Помогите школьнику выполнить домашнее задание: решить задачу (таблица 1.1) и посчитать значение переменной b (таблица 2.1).

Исходные данные вводятся с использованием метода prompt. При выводе информации предусмотреть форматирование документа, вывод текста задания, включая рисунок исходного выражения, и вывод информации о разработчике скрипта.

Таблица 1

|  |  |  |
| --- | --- | --- |
| Вариант | Вычислить | Расчетные формулы |
| 1 | Площадь круга и длину окружности радиуса *r* |  |
| 2 | Площадь и угол при основании равнобедренного треугольника  с основанием *a* и высотой *h* |  |
| 3 | Площадь и периметр прямоугольника  со сторонами *a, b* | *S = ab P =* 2(*a+b*) |
| 4 | Скорость в конце пути и путь, пройденный за время *t* с ускорением *a*  при *v*0 *=* 0 | *v = at* |
| 5 | Площадь и периметр квадрата со стороной *а* | *S = a*2 *P =* 4*a* |
| 6 | Объем и площадь боковой поверхности параллелепипеда со сторонами *а, b, c* | *V = abc S =* 2(*a+b*)*с* |
| 7 | Площадь кольца с внешним радиусом *R*  и внутренним *r* | *S =* π(*R*2*– r*2) |
| 8 | Площадь боковой поверхности и объем цилиндра с радиусом основания *r* и высотой *h* | *S =* 2π*r h V =* π*r* 2*h* |
| 9 | Площадь и периметр прямоугольного треугольника с катетами *a, b*игипотенузой*с* | *P = a + b + c* |
| 10 | Объем и площадь поверхности куба  со стороной *а* | *V = a*3 *S =* 6*a*2 |
| 11 | Периметр и площадь треугольника  со сторонами *a, b, c* | *P = a + b + c =* 2*p* |
| 12 | Площадь основания и объем цилиндра  с радиусом основания *r* и высотой *h* | *S =* π*r 2 V = Sh* |
| 13 | Объем и площадь основания параллелепипеда со сторонами *а, b, c* | *V = abc S = ab* |
| 14 | Площадь основания и объем конуса  с радиусом основания *r* и высотой *h* | *S =* π*r* 2 |
| 15 | Гипотенузу и площадь прямоугольного треугольника с катетами *a, b* |  |
| 16 | Высоту и площадь равнобедренной трапеции с основаниями *a, b* (*b > a*)  и углом при большем основании *α* |  |
| 17 | Площадь поверхности и объем шара радиуса *R* |  |
| 18 | Скорость в конце пути и путь, пройденный телом за время *t*  с ускорением a и начальной скорости *v0* | *v = v*0 *+ at* |
| 19 | Радиус круга, вписанного в треугольник со сторонами *a, b, c* |  |
| 20 | Кинетическую энергию тела массой m, движущегося со скоростью *v* |  |
| 21 | Площадь прямоугольного треугольника  с гипотенузой *с* и одним из катетов *a* |  |
| 22 | Периметр и площадь прямоугольного треугольника с катетами *a, b* |  |
| 23 | Высоту и площадь равнобедренного треугольника с основанием *a* и углом  при основании |  |
| 24 | Радиус круга, описанного вокруг треугольника со сторонами *a, b, c* |  |
| 25 | Периметр и площадь параллелограмма  со сторонами *a, b* и острым углом *α* | *P =* 2(*a+b*) |
| 26 | Площадь прямоугольной трапеции  с основаниями *a*, *b* (*b > a*) и углом при большем основании α |  |
| 27 | Сопротивление проводника длиной l, площадью поперечного сечения *S* и удельным сопротивлением ρ |  |
| 28 | Расстояние между точками с координатами *x*1*, y*1 и *x*2*, y*2 |  |
| 29 | Путь, пройденный за время *t* со скоростью *v* | *S = vt* |
| 30 | Емкость плоского конденсатора C  с площадью поверхности одной пластины *S*, расстоянием между пластинами *d* и диэлектрической проницаемостью материала ε |  |

Таблица 2

| Вариант | Вид функции | Вариант | Вид функции |
| --- | --- | --- | --- |
| 1 |  | 16 |  |
| 2 |  | 17 |  |
| 3 |  | 18 |  |
| 4 |  | 19 |  |
| 5 |  | 20 |  |
| 6 |  | 21 |  |
| 7 |  | 22 |  |
| 8 |  | 23 |  |
| 9 |  | 24 |  |
| 10 |  | 25 |  |
| 11 |  | 26 |  |
| 12 |  | 27 |  |
| 13 |  | 28 |  |
| 14 |  | 29 |  |
| 15 |  | 30 |  |

Для вычисления арифметического выражения используются методы объекта **Math**:

|  |  |
| --- | --- |
| **Методы** | **Описание** |
| abs(x) | Возвращает абсолютное значения (модуль) числа x. |
| acos(x) | Возвращает арккосинус числа x в радианах. |
| asin(x) | Возвращает арксинус числа x в радианах. |
| atan(x) | Возвращает арктангенс числа x как численное значение между -PI/2 и PI/2 . |
| ceil(x) | Округляет значение x до первого большего целого числа. |
| cos(x) | Возвращает косинус числа x (число x задается в радианах). |
| exp(x) | Возвращает значение E в степени x. |
| floor(x) | Округляет значение x до первого меньшего целого числа. |
| log(x) | Возвращает натуральный логарифм (с основанием E) х. |
| max(x1,x2,...xn) | Возвращает большее из чисел x1,x2,...xn. |
| min(x1,x2,...xn) | Возвращает меньшее из чисел x1,x2,...xn. |
| pow(x,y) | Возводит x в степень y и возвращает результат. |
| random() | Возвращает случайное число между 0 и 1 (например 0.6230522912910803). |
| round(x) | Округляет значение x до ближайшего целого числа. |
| sin(x) | Возвращает синус числа x (число x задается в радианах). |
| sqrt(x) | Возвращает квадратный корень x. |
| tan(x) | Возвращает тангенс угла. |

**Требование по содержанию отчета:**

В отчете должны быть отображены следующие пункты:

1. Титульный лист.
2. Цель работы.
3. Задания.

5. Листинг кода сценария с комментариями.

6. Результат выполнения скрипта.

8. Выводы.

**Контрольные вопросы для защиты:**

1. Какие виды диалоговых окон вы знаете?
2. Как ввести данные пользователя?
3. Как преобразовать строку в число?
4. Какие типы переменных используются в JavaScript?
5. Что означает значение **undefined**?
6. Как вывести сообщение в диалоговое окно?
7. Для чего используется ключевое слово **var**?
8. Для чего используется ключевое слово **let**?
9. Для чего используется ключевое слово **const**?
10. Для чего используется метод **write** объекта **document**?
11. Для чего используется встроенный объект **Math**?
12. Как записать сложное арифметическое выражение?