#### Переменные и типы данных

Javascript во многом наследовал синтаксис языка C, а это значит, что строчные и заглавные буквы в идентификаторах различаются. Нет необходимости специально объявлять переменные, хотя это является хорошей программистской практикой. Первое появление переменной автоматически является ее объявлением. Имя переменной должно начинаться с буквы или символа подчеркивания (\_), последующими символами могут быть буквы, цифры и символ подчеркивания (\_). Нельзя использовать в качестве имен переменных зарезервированные слова, такие как if, for, function, new. Для объявления переменной может использоваться необязательное слово var. Его использование обязательно только в том случае, если переменная объявляется внутри функции и пределом видимости этой переменной должна быть эта функция. Переменные, объявленные без использования var, являются глобальными. Примеры правильного объявления:

 var test; //объявление без присвоения. Значение переменной undefined

 a = 2;

 var b2 = 3;

 sum\_a\_b2 = a + b2;

Примеры недопустимых имен:

var 3piglets // начинается с цифры

var Smith&Wesson // символ & не допускается

var for // for является зарезервированным словом

В Javascript есть три **простых типа данных**: строка (String), число (Number) и логический тип (Boolean), два **составных типа**: объект (Object) и массив (Array) и два **специальных типа**: пустое значение (Null) и неопределенное значение (Undefined). Тип данных ассоциирован не с именем переменной, а с ее значением.

Строковые данные – любая цепочка символов, заключенная в одиночные (') или двойные кавычки ("). Одиночные кавычки могут содержаться в строках, заключенных в двойные кавычки и наоборот.

"простая строка"

'еще одна строка'

"don't worry"

Числовые данные подразделяются на целые и числа с плавающей запятой (фактически, точкой), но внутренним представлением данных всегда является число с плавающей запятой. Целые числа могут быть представлены в десятеричной, восьмеричной и шестнадцатеричной форме. Числа с плавающей запятой могут записываться в степенной форме.

|  |  |  |
| --- | --- | --- |
| **Число** | **Десятичное представление** | **Описание** |
| 0.0001, .0001, 1e-4, 1.0e-4 | 0.0001 | Разные представления числа с плавающей запятой |
| 125 | 125 | Целое |
| 0377 | 255 | Восьмеричное представление целого |
| 0378 | 378 | Десятичное целое. Восьмеричное представление допускает лишь цифры от 0 до 7 |
| 0xff | 255 | Шестнадцатеричное представление |
| 0x3e7 | 999 | Шестнадцатеричное представление |
| 00.001 0x0.001 | - | Ошибка компиляции. Восьмеричные и шестнадцатеричные числа могут быть только целыми |

Данные логического (Boolean) типа принимают лишь одно из двух значений: true (истина) или false (ложь).

 y = (x == 10); //истина, если x равно 10

ЗАДАНИЕ

Реализовать во внешнем файле ряд сценариев. Оперировать стандартным набором типов данных и операторов для произвольных значений переменных. Организовать вывод результатов в консоль. (ссылка на справочник <https://learn.javascript.ru/operators> )

Пример:

//сложение

var x=5;

var y=1;

var c;

c=x+y;

console.log(c);