1. Дан код:

var a = 1, b = 1, c, d;

c = ++a; alert(c); // 2

d = b++; alert(d); // 1

В постфиксной форме сначала происходит возвращение значения, а потом выполняется инкрементирование. В префиксной форме инкрементирование производится сразу, а возврат — уже с обновленным значением.

Поэтому в первом случае 2, во втором 1

c = (2+ ++a); alert(c); // 5

Новое значение а=2, ++а = 3, 2+ ++а = 5

d = (2+ b++); alert(d); // 4

Новое значение b=1+1=2, b++=2, 2+ ++b = 4

alert(a); // 3

новое значение a=3, взято прошлое значение

alert(b); // 3

Новое значение b=3, взято прошлое значение и на данной итерации инкрементировано

Почему код даёт именно такие результаты?

2. Чему будет равен x в примере ниже?

var a = 2;

var x = 1 + (a \*= 2);

x=5

3. Объявить две целочисленные переменные a и b и задать им произвольные начальные значения. Затем написать скрипт, который работает по следующему принципу:

* если a и b положительные, вывести их разность;
* если а и b отрицательные, вывести их произведение;
* если а и b разных знаков, вывести их сумму; *ноль можно считать положительным числом*.

Честно говоря, подсмотрела в 3 лекции, что перед prompt надо ставить +, чтобы получалось число, потому что без этого результат с + конкатенировался, хотя – и \* отрабатывали корректно

4. Присвоить переменной а значение в промежутке [0..15]. С помощью оператора switch организовать вывод чисел от a до 15.

+

5. Реализовать основные 4 арифметические операции в виде функций с двумя параметрами. Обязательно использовать оператор return.

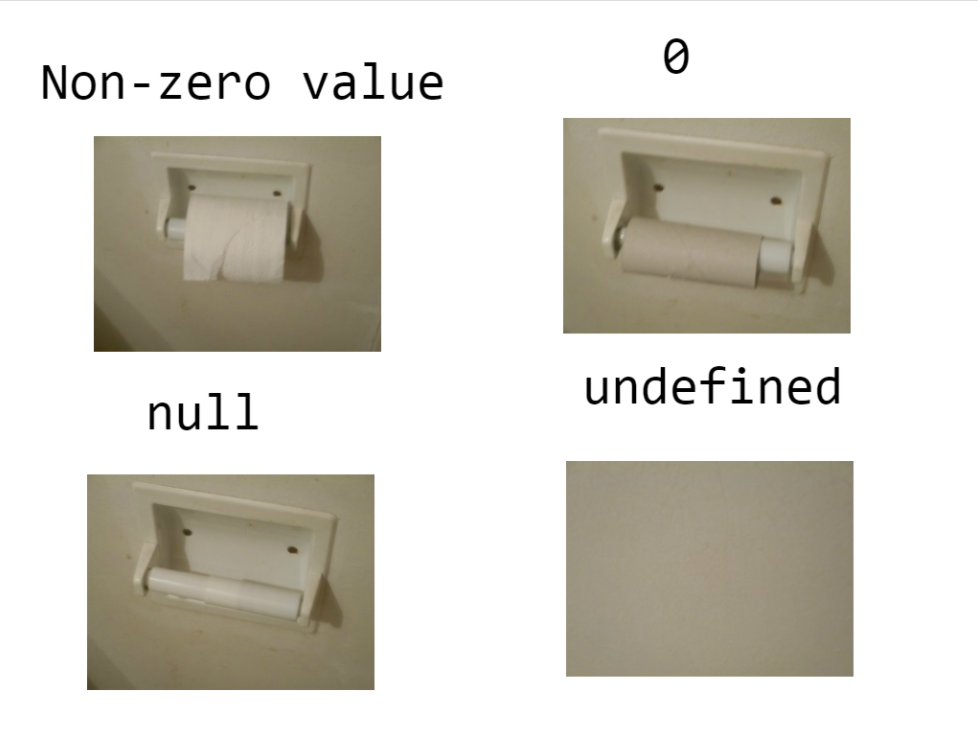
+

6. Реализовать функцию с тремя параметрами: function mathOperation(arg1, arg2, operation), где arg1, arg2 – значения аргументов, operation – строка с названием операции. В зависимости от переданного значения операции выполнить одну из арифметических операций (использовать функции из пункта 5) и вернуть полученное значение (использовать switch).

7. \*Сравнить null и 0. Попробуйте объяснить результат.

Насколько я правильно поняла, 0 – это когда объект есть, но он =0, а null – когда объект должен быть, но его нет

Ну вот мне кажется картинка оч в тему)))



8. \*С помощью рекурсии организовать функцию возведения числа в степень. Формат: function power(val, pow), где val – заданное число, pow – степень.

6 и 8 не успела