* Какое будет выведено значение: let x = 5; alert( x++ ); ?

Ответ: 5 («x» увеличится после вывода)

* Чему равно такое выражение: [ ] + false - null + true ?

Ответ: NaN т.к массив приводит к типу «string» , а «- null» к типу «number», а число не получается, из такой то каши)))

* Что выведет этот код: let y = 1; let x = y = 2; alert(x); ?

Ответ: 2 (присваивание происходит справа на лево, т.е. 2=y=x)

* Чему равна сумма [ ] + 1 + 2?

Ответ: 12 (числа ведут себя как строка в уравнении из-за [ ])

* Что выведет этот код: alert( "1"[0] )?

Ответ: 1 (при этом если поставить [1] то будеи undefined ) ПОЧЕМУ?

* Чему равно 2 && 1 &&null&& 0 &&undefined ?

Ответ: null (при сравнении «&&» доходит до первого false

* Есть ли разница между выражениями? !!( a&& b ) и (a && b)?

Ответ: Да , раздница в том, что «!!» - это двойное отрицание , может применяться для приведения к булиевому типу данных. Например если:   
a=1; b=2; (a && b) выведет 2  
 !!(a && b) выведет true

* Что выведет этот код: alert( null || 2 && 3 || 4 ); ?

Ответ: 3 (приоритет больше у && , вытикающий порядок:  
 (2&&3)= 3  
null || 3 = 3  
3 || 4 = 3

* a = [1, 2, 3]; b = [1, 2, 3]; Правда ли что a == b?

Ответ: false это разные массивы

* Что выведет этот код: alert( +"Infinity" ); ?

Ответ: числовое значение Infinity

* Верно ли сравнение: "ёжик" > "яблоко"?

Ответ: true кирилица преобразуется в…. другую кодировку и сравнивается в ней

* Чему равно 0 || "" || 2 || undefined || true || falsе ?  
  Ответ число 2   
  0 || "" = “”  
  "" || 2 = 2  
  2 || undefined = 2 || true = 2 || falsе = 2