* Какое будет выведено значение: let x = 5; alert( x++ ); ?  
  ~~6~~

5 // сначала код возвращает 5, а на следующий цикл уже возвращает 6

* Чему равно такое выражение: [ ] + false - null + true ?  
  ~~Underfind~~1) [ ] + false  
  когда мы работаем с пустым массивом в таких операциях, он будет приведен к строковому типу данных, т.е. к строке. По динамической типизации, когда мы конкатенирует строку с другим типом данных, мы получаем строку  
  2) [ ] + false – null – получим NaN

3) [ ] + false - null + true - получим NaN

* Что выведет этот код: let y = 1; let x = y = 2; alert(x); ?  
  2 // correct
* Чему равна сумма [ ] + 1 + 2?  
  ~~3~~  
  12 // сложим строки
* Что выведет этот код: alert( "1"[0] )?  
  ~~"1"[0]~~  
  1 – это число, [0] – порядковый номер массива  
  ответ: 1.
* Чему равно 2 && 1 && null && 0 && undefined ?  
  ~~false~~

//оператор && всегда запинается на лжи  
т.е 2 – правда, 1 – правда, null – ложь и код дальше не идет  
ответ: null

* Есть ли разница между выражениями? !!( a && b ) и (a && b)?  
  да, ~~первое это все варианты кроме а и b~~  
  в первом случае вернется булиновое значение, что точно не равняется второму выражению
* Что выведет этот код: alert( null || 2 && 3 || 4 ); ?  
  ~~false~~3

//таблица приоритетов сравнения   
логическое И && стоит выше, чем логическое ИЛИ ||

&& возвращает последнее true значение

* a = [1, 2, 3]; b = [1, 2, 3]; Правда ли что a == b ?  
  нет //correct
* Что выведет этот код: alert( +"Infinity" ); ?  
  тип данных number  
  выведет Infinity //correct
* Верно ли сравнение: "Ёжик" > "яблоко"?  
  no //correct   
  смотреть кодировку по символам
* Чему равно 0 || "" || 2 || undefined || true || falsе ?

2 // correct