一、typeof

1. typeof [] 的执行结果是 (\_\_)
2. typeof 123 的执行结果是 (\_\_)
3. typeof null 的执行结果是 (\_\_)
4. typeof undefined 的执行结果是 (\_\_)
5. typeof function(){} 的执行结果是 (\_\_)
6. typeof运算符运算后有一个返回结果，这个结果它自身的数据类型是 (\_\_)

二、数据类型转换

1、!!{} 的执行结果是 (\_\_)

2、!!0 的执行结果是 (\_\_)

3、!!null 的执行结果是 (\_\_)

4、!!undefined 的执行结果是 (\_\_)

5、!!NaN 的执行结果是 (\_\_)

6、!![] 的执行结果是 (\_\_)

7、!false 的执行结果是 (\_\_)

8、!200 的执行结果是 (\_\_)

三、相等判断

1. 1 == true 的执行结果为 (\_\_)
2. 0 == false 的执行结果为 (\_\_)
3. 6 == [6] 的执行结果为 (\_\_)
4. 0 == [] 的执行结果为 (\_\_)

5、0 == null 的执行结果为 (\_\_)

6、false == undefined 的执行结果为 (\_\_)

7、null == undefined 的执行结果为 (\_\_)

8、{} == {} 的执行结果为 (\_\_)

9、“[object Object]”== {} 的执行结果为 (\_\_)

10、NaN == NaN 的执行结果为 (\_\_)

11、NaN == flase 的执行结果为 (\_\_)

12、[] == false 的执行结果为 (\_\_)

四、运算符

1、var val = []? 1 : 2; val的值为 (\_\_)。

2、var a = 1 – 1? 10 : 20; a的值为 (\_\_)。

3、null && {} && 2的执行结果为 (\_\_)

4、123 && 321 && 0 的执行结果为 (\_\_)

5、[] && 0 && false 的执行结果为 (\_\_)

6、1 || false || null 的执行结果为 (\_\_)

7、0 || undefined || NaN || 3 的执行结果为 (\_\_)

五、arguments

1、arguments是代表实参的对象，还是形参的对象：(\_\_)

2、arguments的(\_\_)属性可以用来获取实参的个数

3、如何通过arguments获取实参：(\_\_)

4、arguments是对象还是数组：(\_\_)

5、实参和形参的关系：(\_\_)用来接收(\_\_)传递过来的数据