一、typeof

1. typeof [] 的执行结果是 (\_object\_)
2. typeof 123 的执行结果是 (\_number\_)
3. typeof null 的执行结果是 (\_object\_)
4. typeof undefined 的执行结果是 (\_undefined\_)
5. typeof function(){} 的执行结果是 (\_function\_)
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 (\_string\_)

二、数据类型转换

1、!!{} 的执行结果是 (\_true\_)

2、!!0 的执行结果是 (\_false\_)

3、!!null 的执行结果是 (\_false\_)

4、!!undefined 的执行结果是 (\_false\_)

5、!!NaN 的执行结果是 (\_false\_)

6、!![] 的执行结果是 (\_true\_)

7、!false 的执行结果是 (\_true\_)

8、!200 的执行结果是 (\_false\_)

三、相等判断

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

5、0 == null 的执行结果为 (\_false\_)

6、false == undefined 的执行结果为 (\_false\_)

7、null == undefined 的执行结果为 (\_true\_)

8、{} == {} 的执行结果为 (\_false\_)

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

10、NaN == NaN 的执行结果为 (\_false\_)

11、NaN == flase 的执行结果为 (\_false\_)

12、[] == false 的执行结果为 (\_true\_)

13、[1,2] ==“1,2 ”的执行结果为 (\_true\_)

四、运算符

1、var val = []? 1 : 2; val的值为 (\_1\_)。

2、var a = 1 - 1? 10 : 20; a的值为 (\_20\_)。

3、null && {} && 2的执行结果为 (\_null\_)

4、123 && 321 && 0 的执行结果为 (\_0\_)

5、[] && 0 && false 的执行结果为 (\_0\_)

6、1 || false || null 的执行结果为 (\_1\_)

7、0 || undefined || NaN || 3 的执行结果为 (\_3\_)

五、arguments

1、arguments是代表实参的对象，还是形参的对象：(\_实参\_)

2、arguments的(\_length\_)属性可以用来获取实参的个数

3、如何通过arguments获取实参：(\_下标,索引\_)

4、arguments是对象还是数组：(\_对象\_)

5、实参和形参的关系：(\_形参\_)用来接收(\_实参\_)传递过来的数据