**Chapter 8: Operators**

Theoperators used in javascript are:

* Arithmetic
* Assignment
* Comparison
* Logical
* Bitwise
* Typeof
* Ternary
* in

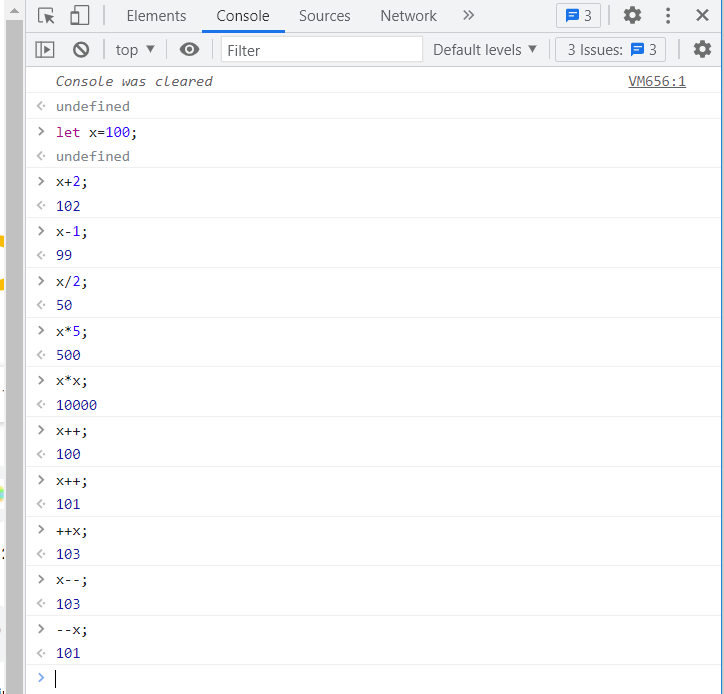
**Arithmetic operators:**

They are the basic operators which do addition(+), subtraction(-), multiplication (\*), division (/), modulus (%), increment (++), decrement (--).



These operations can be performed directly on variables also:

For example:

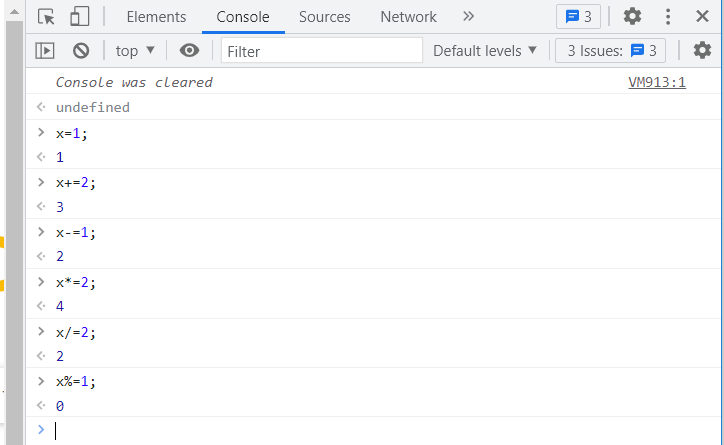


The arithmetic operator performs from left to right whereas the assignment operator performs from right to left.

**Assignment:**

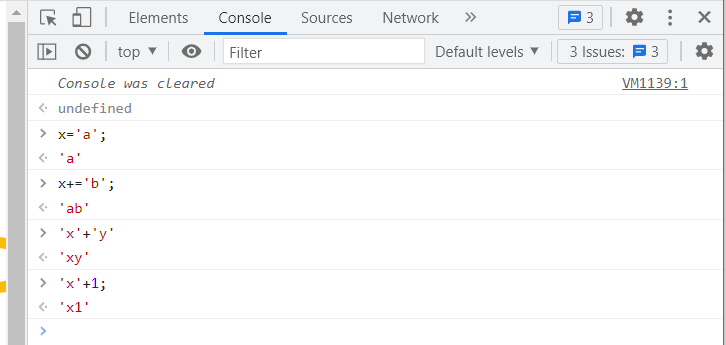
x=1 means 1 is assigned to x

x+=1 means x=x+1; here value of x is added to 1 and assigned to x. Now x=2.



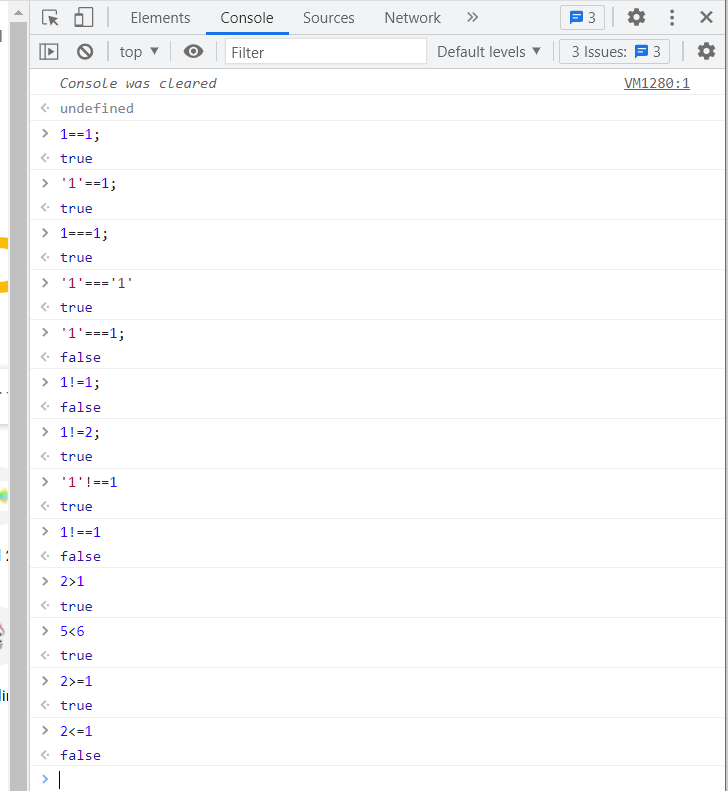
**String:**

When + operator is used with strings it will concatenate rather than performing addition. When any one of the operand is string and the other is number it will consider both as strings and it will concatenate finally.



**Comparison operator:**

This == operator equates the value and this === operator equates the value and its type.

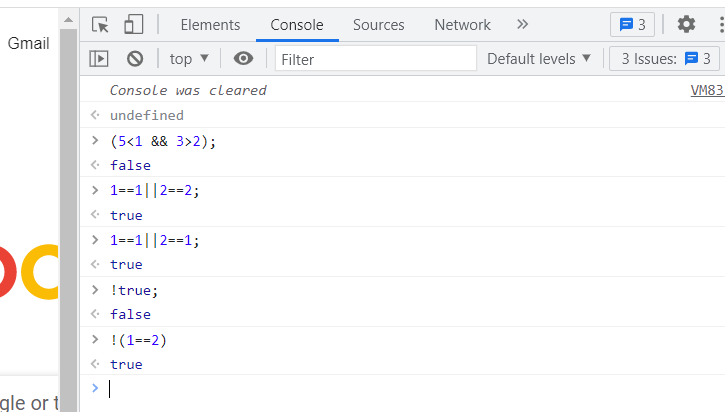


**Logical operator:**

&&- Logical AND- the output is true if both the condition is true else false

||- logical OR- the output is true if any one of the condition is true.

!- logical NOT- it will reverse the output.



**Bitwise operator:**

& -AND- sets each bit to 1 if both the bits are 1.

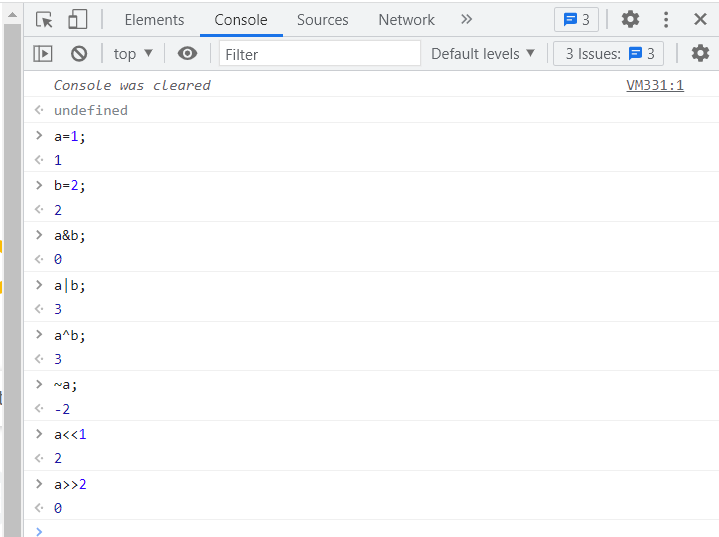
| - OR - sets each bit to 1 if one of the two bits is 1.

^- XOR- sets each bit to 1 if only one of two bits is 1.

~ - NOT- inverts all the bits

<<- shifts left by pushing zeros in from the right.

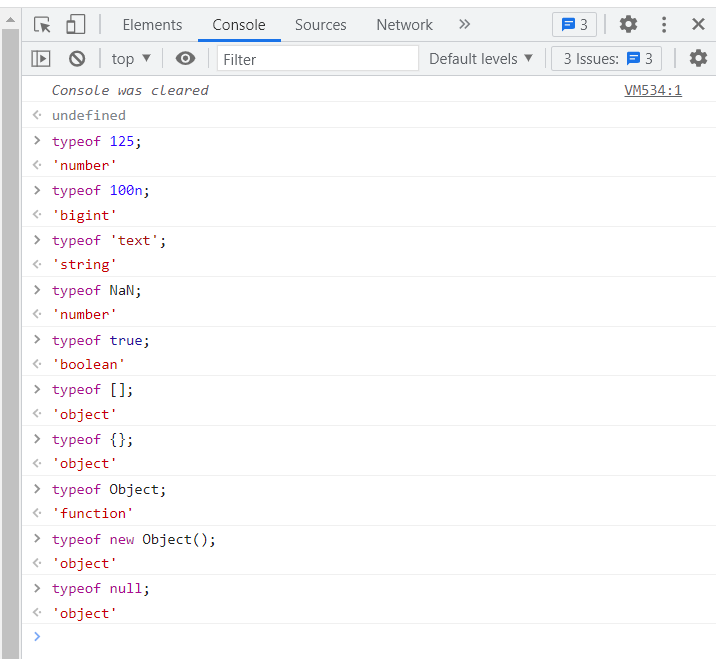
>> - shifts right by pushing zeros in from the left.



**Typeof:**

The typeof operator is used to check the type of a value.

There is one exception, NaN (Not-a-Number) is always treated as Number.

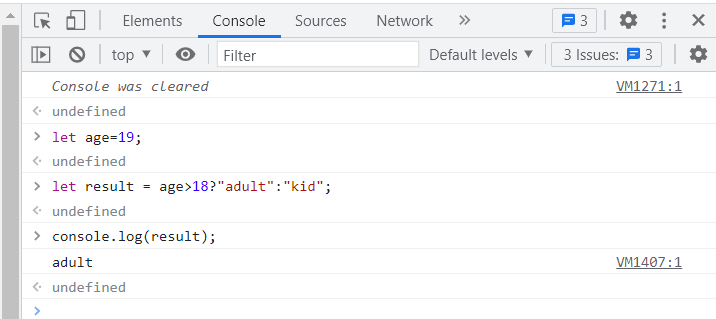


**Ternary operator(?:)**

Ternary operator has the form of:

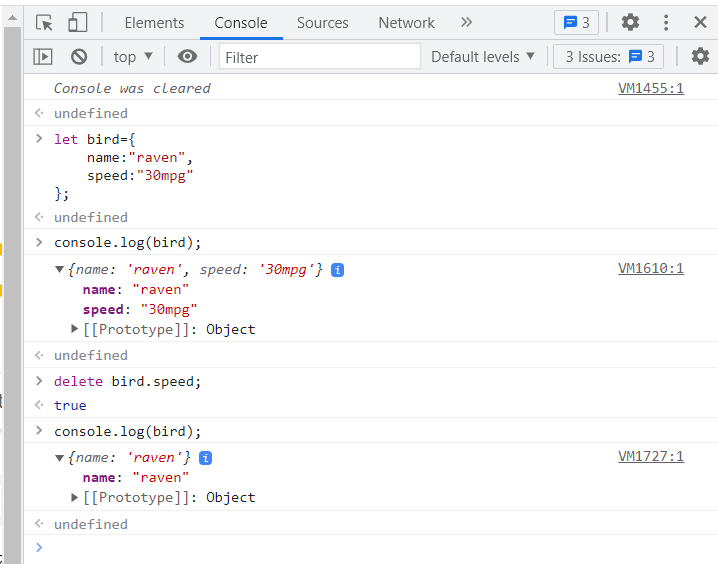
**Statement?statement:statement;**

It is the short form of **if-statement**.



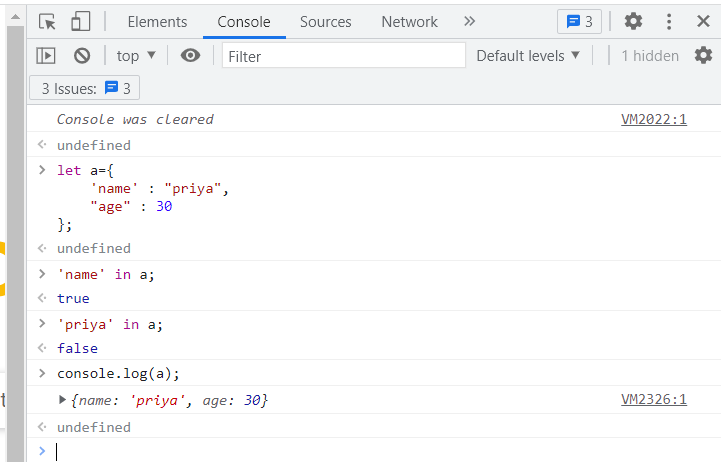
**Delete**:

The delete keyword is used to delete an object property.The delete keyword doesnot allow to delete any variables.

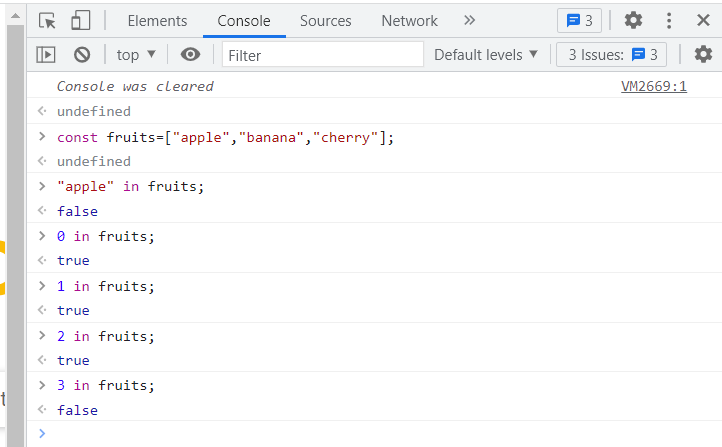


**In** operator:

The **in** operator is used to check if a **property** **name** exists in an object and it doesnot apply for property value.



Similarly the in operator is used to check if the index value exists in the array and not for the values in the array.



The “length” property is native to all arrays, and does not exist natively on an object unless it is added explicitly.

