



# JavaScript Expression

Any unit of code that can be evaluated to a value is an expression. Since expressions produce values, they can appear anywhere in a program where JavaScript expects a value such as the arguments of a function invocation. As per the MDN documentation, JavaScript has the following expression categories. .



# JavaScript Arithmetic Expressions:

Arithmetic expressions evaluate to a numeric value. Examples include the following

```
10;    // Here 10 is an expression that is evaluated to the numeric  
        value 10 by the JS interpreter  
10+13; // This is another expression that is evaluated to produce  
        the numeric value 23
```



# JavaScript String Expressions:

String expressions are expressions that evaluate to a string. Examples include the following

```
'hello';  
'hello' + 'world'; // evaluates to the string 'hello world'
```



## JavaScript Logical Expressions:

Expressions that evaluate to the boolean value true or false are considered to be logical expressions. This set of expressions often involve the usage of logical operators && (AND), ||(OR) and !(NOT). Examples include

```
10 > 9; // evaluates to boolean value true
10 < 20; // evaluates to boolean value false
true; //evaluates to boolean value true
a===20 && b===30; // evaluates to true or false based on
the values of a and b
```



# JavaScript Primary Expressions:

Primary expressions refer to stand alone expressions such as literal values, certain keywords and variable values. Examples include the following

```
'hello world'; // A string literal
23;           // A numeric literal
true;         // Boolean value true
sum;          // Value of variable sum
this;         // A keyword that evaluates to the current object
```