Type Conversion

Overview

JavaScript is a loosely typed language, and most of the time, operators automatically convert a value to the right type. Still, there are also cases when we need to do **type conversions explicitly.**

- Converting Strings to Numbers
- Converting Numbers to Strings
- Converting Booleans to Numbers
- Converting Numbers to Booleans

Converting Strings to Numbers

Valid String, which can be converted to a number is converted using **Number( )**
Empty String -> Converts to 0
Invalid String -> Converts to NaN

```
Example :   Number(“22.3”) ;  // Returns 22.3
            Number(“ “) ;       //Returns 0
            Number(“Coding Ninjas”) ; // Returns NaN
```

Methods similar to Number( ) :

- **parseInt( ) :** Parses a string and returns an integer
- **parseFloat( ) :** Parses a string and returns a floating point number

```
Example :   parseInt("22.3") ;  // Returns 22
            parseFloat("22.3") ;  // Returns 22.3
            parseFloat("22") ;  // Returns 22
```

Converting Numbers to Strings

The **String ( ) and toString( )** method can convert numbers to strings.

```
Example :   String(22.3) ;  // Returns "22.3"
            String(20+11) ;  // Returns "31"
            (15.3).toString( ) ;  // Reuturns "15.3"
```

Converting Booleans to Numbers

The **number( )** method is used to convert booleans to numbers.

```
Example :   Number(false) ;    // returns 0
            Number(true);     // returns 1
```

Converting Booleans to Strings

The **String ( ) and toString( )** method can convert Boolean to String.

```
Example :   String(false)     // returns "false"
            String(true)      // returns "true"
            false.toString()  // returns "false"
            true.toString()   // returns "true"
```

Conversions done by + operator

```
"10" + 5 ;        // returns "15" as a String
"10" - 5 ;        // returns 5 as a Number
10 + null ;       // returns 10 because null is converted to 0
"10" + null ;     // returns "10null"  as a String
“10” + undefined ; // returns “10undefined” as a String
"10" * "5"  ;     // returns 50 as a Number
```