

使用頻度はあまり高くありませんが、金額の処理をするのにいくつか使用が必要なのが数学的な定数とメソッドを持つのが オブジェクトです。このオブジェクトのメソッドは、クラスメソッドとして常にオブジェクト名.メソッド名(引数)という形で使用します。引数の数値に対して、以下のメソッドが使用できます。

<input type="text"/>	: 小数点以下を切り上げる
<input type="text"/>	: 小数点以下を切り捨てる
<input type="text"/>	: 小数点以下を四捨五入する
<input type="text"/>	: 0以上1未満の疑似ランダムな数値を返す(引数なし)

日付や時間を扱う **Date オブジェクト**は少し扱いがやっかいですが、**new Date()** で現在のものを取得し、**引数に(年,月,日)** を指定する事によって任意のものを取得できます。但し、月のみ ~ で指定する必要があり、取得した場合同様の範囲で取得されます。オブジェクトに対して以下のメソッドが使用できます。

<input type="text"/>	: 年を4桁の整数で取得する
<input type="text"/>	: 月を取得する
<input type="text"/>	: 日を1から31の整数で取得する
<input type="text"/>	: 1970年からの経過ミリ秒数を取得する

最後のメソッドを使ってユニークな数値を取得する事ができるので、JavaScript で呼び出す に付加して**キャッシュが効かないようにする**事ができます。

JavaScript のグローバルな関数で、文字列を整数に変換する が存在します。引数の文字列が整数に変換しなかった場合は、特殊な値である を返します。逆に、数値の入った変数を文字列に変換するメソッドは変数名、 ()ですが、変数内が整数の場合、メソッドの引数に をセットすると、16進数文字列になります。なので、"0x" + 1 6 進数文字列 として整数に変換する関数を実行すると元の整数に戻ります。また、文字列としての連結演算子は なので、数値の後ろから空文字を連結すると文字列になります。

JavaScript 内の単純オブジェクトを外部とのやりとりに使用する為に、 オブジェクトが存在し、**二つのメソッド**があります。

<input type="text"/>	: 文字列をオブジェクトとして解析し、JavaScriptの値やオブジェクトを作成。
<input type="text"/>	: JavaScript のオブジェクトを文字列に変換します。

文字列に変換する際、第2引数は を指定し、第3引数に値を指定するとその数のスペースでインデントして整形されます。

function 内でのみ利用できる オブジェクトがあり、プロパティとして を持っています。これを使用して、function の仮引数を定義せずに、渡された引数の内容を知る事ができます。また、渡されていない部分を参照した場合は、 となります。また、function もオブジェクトであり、既存の String オブジェクトに追加のインスタンスメソッドを追加作成する事ができます。そのメソッド名を **myfunc** とすると、以下の定義になります。

String. .myfunc = function(){ 処理 };

処理の中で、本体の文字列を取得するには this. () を使用します。引数の指定も可能ですし、前述のオブジェクトで動的に引数をチェックする事もできます。

使用方法は複雑ですが、正規表現を利用する為のオブジェクトとして が存在します。new を使用して新しい正規表現オブジェクトを作成する時に、引数に文字列のパターンを指定できます。String オブジェクトのメソッドの引数として利用する事が出来、そのメソッドとしては や や があります。