

WEB アプリケーションの処理の**構成要素は大きく分けて3つ**あります。その一つは**画面表示**で、二つ目は**データベースの更新**です。しかし、この二つの構成要素は比較的単純な事が多く、残りの要素が最も負荷の高い となります。

当然この内容はアプリケーション毎に違ったものなり、画面の要素によってもいろいろな注意が必要となります。そして、画面表示の一部ではありますが、ユーザの入力によって起きる不具合をユーザに知らせるための の表示が必要となります。

この処理は基本的にはサーバ側で行われますが、WEBアプリのテクノロジーを総動員するとブラウザ側での処理として実装する事も最近は少なくありません。

最も簡単な方法としては、HTML 要素に直接記述する属性によって、**簡易的な制限を加える**事ができます。その代表的な属性が という属性で、**未入力しないようにする**方法です。ただ、この方法は完全では無く、スペースや漢字スペースを受け入れてしまうので実際は**正規表現**を用いた 属性のほうが現実的です。

しかし、この方法ではユーザに対するメッセージが具体的なものでは無く、製品に利用するのは避けるべきでしょう。せめて、 を使用した入力単位の検査を作成してリアルタイムに問題の提示を迅速に行えるようにするべきです。さらに言えば、jQuery 等のライブラリを使用して、プログラマの負担の少ない**標準化**を行うべきです。

PHP で未入力がどうかを検査する場合、**空白(漢字の空白も含む)**を未入力としてチェックする為に、 関数を使用して正規表現で入力文字列の両側の空白を排除して空文字であればエラーとします。また、**PHP は画面単位で送信**を行うので、サーバへ送られたデータに対して**全て検査**を行い、問題のある結果もまた全てブラウザへ返して表示すべきです。

具体的には、文字列追加連結演算子の を使用して検査の結果を変数にまとめて保存して画面部分に埋め込みます。そして、可能ならば問題の出た最初の画面上のコントロールに をセットしましょう。

しかし、ブラウザ側でチェックするのであればコントロール毎に問題を迅速にユーザに知らせるほうが良い場合もあります。

```
var name = $("#name").val();
name = name.replace(/^[¥$]+/, "");
name = name.replace(/[¥$]+$/, "");
if ( name == "" ) {
    $("#message").text("名前が入力されていません");
    $("#name").val( "" );
    $("#name").focus();
    $("#name").select();

    event.preventDefault();
    return;
}
```

いずれにしても、画面に投入されるデータの整合性をプログラミングでチェックする場合、多くの場面で**正規表現**が有効です。簡単な例で言えば、**0～9 の文字の入力のみ許すのであれば** HTML であれば正規表現を記述する属性の値として と書きます。

また、処理として PHP で記述するのであれば 関数を使用し、JavaScriptであれば文字列に対して メソッドを使用しますが、その場合最初から最後まで 0～9 である事を確かめる為に、^を最初に、\$を最後に加えます。

そして、PHP の場合は、**"/正規表現/"**を引数に与え、JavaScript では **/正規表現/**を引数に与えます。