

# 必ずできる!

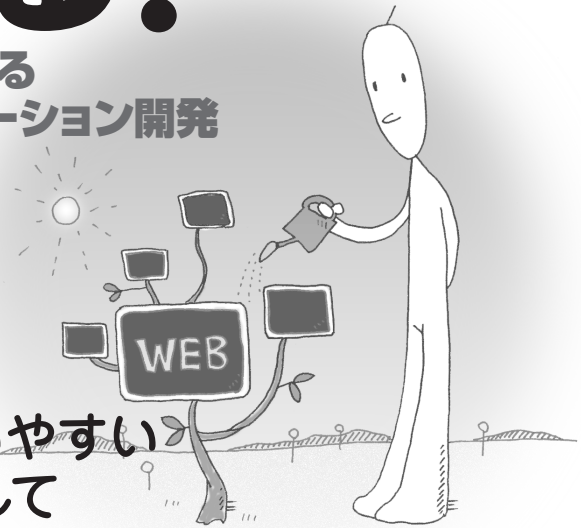
## 基礎固め ゼロからはじめる Webアプリケーション開発

# ASP.NET

第4回

### スケジュール登録画面を改良する～堅牢かつ使いやすいアプリケーションを目指して

山田 祥寛 YAMADA, Yoshitiro  
<http://www.wings.msn.to/>



#### Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2002
- ASP.NET
- Internet Information Services
- Other:  
MSDE

#### Level



#### Samples

この記事で取り上げたソースコードおよびサンプルプログラムは、付録CD-ROMの¥DOTNET¥ASPディレクトリに収録しています。

¥BEGINASPNET  
「グループスケジュール管理」アプリケーション

#### はじめに

本連載では「グループスケジュール管理」アプリケーションの構築を通じて、ASP.NETアプリケーション開発のための基本的な手法を学んでゆきます。

前回までの学習で、ようやくスケジュール登録画面の機能を一通り完成させました。ただ、実際に使用するアプリケーションとしてはいささか貧弱ですので、使い勝手やデータの信頼性、保守性を高めるために、今回はいくつかの改善を施してみたいと思います。

改善点は、「ユーザーが入力したデータの検証」「日付データをスクリーン上のカレンダーから簡易入力する」「コンボボックスのオプションをデータベースのテーブルから動的に生成する」の3つです。それぞれ独立したトピックスではありますが、いずれも今後の応用に向けた重要な内容ばかりですので、基本的な構文からひとつひとつきっちりとおさえてゆきましょう。

#### 不正データの 入力チェック

ユーザーにデータを入力させる場合、その入力データはさまざまです。たとえば、桁数の多い数値であれば、桁区切りのカンマを入力するユーザーがいるかもしれません。また、日付データの表記もさまざまです。「2003/12/25」と入力する人もいれば、「25/12/2003」「20031225」と入力する人もいます。

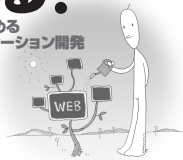
みなさんご存知のように、データベースはあらかじめ登録されるデータの型を厳密に定義しています。つまり、開発当初に予想もしなかったような形式でデータが入力された場合、正しくデータを保存することができないのです。入力されたデータ内容によっては、アプリケーション自体が止まってしまうことさえあります。

そこで、フォーム上でユーザーにデータを入力させる場合、アプリケーションで入力データをチェックする必要があります。これが入力データの「妥

# 必ずできる!

基礎固め ゼロからはじめる  
Webアプリケーション開発

## ASP.NET



当性検証」と呼ばれるものです。

旧来の技術でこの妥当性を検証しようとした場合、なかなか手間がかかりました。「データがきちんと入力されているか」「文字列の長さ、数値は指定範囲にきちんと含まれているか」「E-Mailアドレスは正しい形式で記述されているか」など……。それこそ膨大な量のチェック項目を、いちいち「If…Then」命令などで記述していたのです（「もしも」データが入力されて「いなかったら」エラーを表示する、というようにです）。しかし、この方法ではチェックの内容が変わったときに変更するのが大変ですし、なによりも、アプリケーションの本来のロジック（流れ）がチェックロジックに埋もれてしまってコードが読みにくくなる恐れがあります。

そこで、ASP.NETでは専用の「検証（バリデーション）コントロール」が用意されました。検証コントロールを利用すれば、妥当性検証のためにいちいちプログラムを記述する必要はありません。テキストボックスの横幅を揃えるのとまったく同じ要領で、バリデーションコントロールのプロパティを設定しさえすればよいのです。

また、検証コントロールは、クライアントサイドとサーバーサイドとの双方で妥当性検証を行ないます。そのため、ブラウザのスクリプト機能が有効になっている場合にはクライアント側でチェックが行なわれ、余計なトラフィックがネットワークに流れることはありませんし、スクリプト機能が無効になっている場合にもサーバー側でちゃんと検証が行なわれます。つまり、検証コントロールを使用することで、

開発生産性が向上するだけでなく、高いパフォーマンスと堅牢性が同時に得られるのです。

### ※ 検証コントロール

それでは、実際にinsert.aspxに検証コントロールを組み込んでみることにしましょう。今回、組み込む検証コントロールとその設定値を表1に示しておきます（図1）。各コントロールの意味と利用可能なプロパティについては、表2と表3にまとめておきます。

以上のコントロールを配置することにより、[予定の保存] ボタンをクリックすると、項目の必須チェックや日付の範囲チェックなどが行なわれます。以降のアクション（データベースへの登録処理）は、これらの検証チェックがすべてクリアだった場合にのみ行な

### O N E P O I N T

RegularExpressionValidatorコントロールのValidationExpressionプロパティでは、よく使用する主な正規表現パターンを一覧表から選択することができます（図3）。これによって、URLやメールアドレス、電話番号、郵便番号など比較的よく使う正規表現については、自分でイチから書かなくても済みます。

われます（図2）。

表2のように、ASP.NETには多くの検証コントロールが存在しますが、その中でもValidationSummaryコントロールだけはいささか特殊である点に注目してみてください。ValidationSummaryコントロールは、それ自体、検証チェックを行なうものではありません。その名の通り、検証結果をサマリするためのコントロールです。もちろん、

表1：insert.aspx上の検証コントロール

コントロール	名前 (ID)	プロパティ	設定値
RequiredFieldValidator	reqTitle	ControlToValidate	txtTitle
		Display	Static
		ErrorMessage	タイトルは必須入力です
	reqDat	ControlToValidate	txtDat
		Display	Static
		ErrorMessage	日時は必須入力です
RangeValidator	rngDat	ControlToValidate	txtDat
		Display	Static
		ErrorMessage	日時は1975/1/1～2010/12/31の範囲で入力してください
		MaximumValue	2010/12/31
		MinimumValue	1975/1/1
		Text	*
		Type	Date
ValidationSummary	valSum	DisplayMode	BulletList
		HeaderText	以下のエラーが発生しました
		ShowMessageBox	True
		ShowSummary	False