

## 42.3 .NET: PrintDialog

### 42.3.1 説明

PrintDialog は、ユーザーがプリンタを選択してその属性を設定できるダイアログ ボックスを表示する。  
以下、.NET Framework 開発者ガイドより抜粋。

Windows フォームの PrintDialog コンポーネントは、Windows アプリケーションで出力先プリンタや印刷範囲などの印刷設定に使用する、定義済みダイアログ ボックスです。このダイアログ ボックスは、独自のダイアログ ボックスを使用せずにプリンタなどの印刷設定を行うための簡易ソリューションとして使用します。ドキュメント全体、選択したページ範囲、選択した部分など、さまざまな印刷範囲を指定できます。Windows の標準のダイアログ ボックスを使用して、一般的な基本機能を持つアプリケーションを作成できます。PrintDialog コンポーネントは、CommonDialog クラスを継承しています。実行時にダイアログ ボックスを表示するには、ShowDialog メソッドを使用します。このコンポーネントには、印刷ジョブ単位で適用されるプロパティ (PrintDocument クラス) とプリンタ単位で適用されるプロパティ (PrinterSettings クラス) があります。このいずれかを複数のプリンタで共有できます。

## 42.4 Qt: QPrintDialog

### 42.4.1 説明

QPrintDialog は、フォントを選択するダイアログである。ただし、Windows と Mac OS X ではプラットフォーム固有のダイアログを表示する。

42.4.2 スクリーンショット

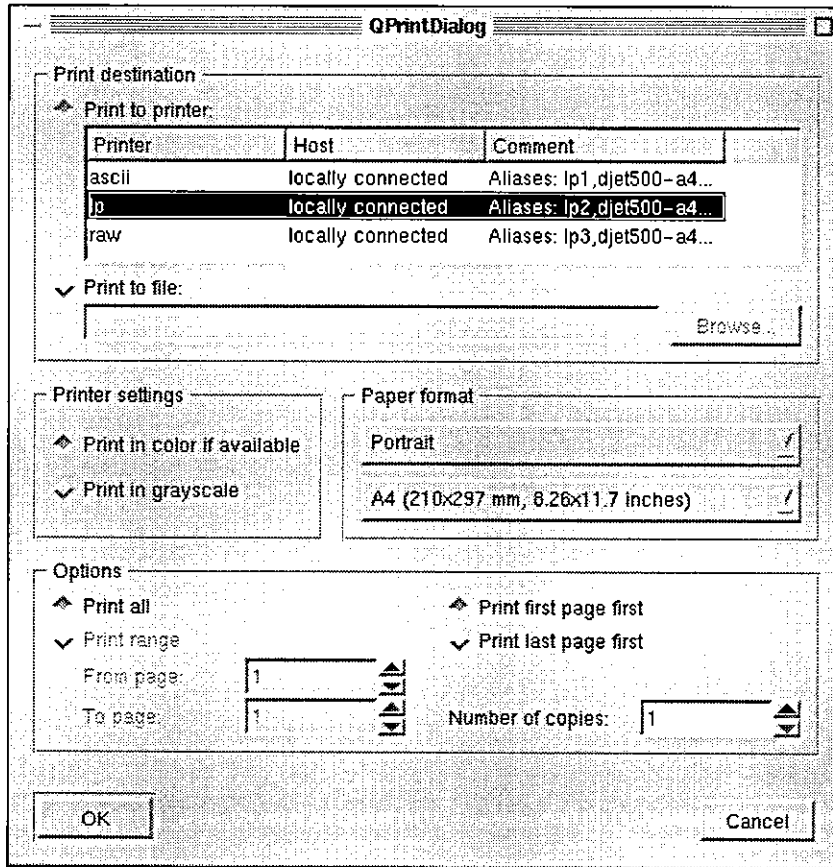


図 42.1 QPrintDialog の Linux 上での実行例 (Qt のリファレンスより抜粋)

## 第 43 章

# 印刷プレビューダイアログ

### 43.1 概要

印刷プレビューダイアログは、印刷時にアプリケーションの文書がどのように印刷されるかを示すダイアログボックスである。

プラットフォーム	ツールキット	コントロール
Windows	Windows Controls .Net Windows Forms Visual Basic 6	PrintPreviewDialog
Mac OS X	Carbon Cocoa	
X11R6	GTK+ GNOME Qt KDE	
Java	AWT Swing SWT	
Mozilla	Gecko	
Web	HTML XForms	

### 43.2 .NET: PrintPreviewDialog

#### 43.2.1 説明

PrintPreviewDialog は、PrintDocument オブジェクトが印刷時にどのように表示されるかを示すダイアログボックスを表示する。

以下、.NET Framework 開発者ガイドより抜粋。

Windows フォームの PrintPreviewDialog コントロールは、PrintDocument を印刷時の状態で表示する、定義済みダイアログボックスです。このコントロールは、独自のダイアログボックスを使用しない簡易ソリューションとして、Windows アプリケーションで使用します。このコントロールには、印刷を開始するボタン、

ズームイン用のボタン、1 ページまたは複数ページを表示するボタン、およびダイアログ ボックスを閉じるためのボタンがあります。

このコントロールの主要なプロパティは、プレビューするドキュメントを設定する Document プロパティです。ドキュメントは、PrintDocument オブジェクトである必要があります。ダイアログ ボックスを表示するには、コントロールの ShowDialog メソッドを呼び出す必要があります。アンチエイリアシングを使用すると、文字の外観は滑らかになりますが、表示は遅くなります。使用する場合は、UseAntiAlias プロパティを true に設定します。

いくつかのプロパティは、PrintPreviewDialog に含まれる PrintPreviewControl を通して使用できます。PrintPreviewControl をフォームに追加する必要はありません。コントロールは、ダイアログをフォームに追加したときに自動的に PrintPreviewDialog に含められます。PrintPreviewControl を通して使用できるプロパティの例として、Columns プロパティおよび Rows プロパティがあります。これらのプロパティは、コントロール上で横方向および縦方向に表示されるページの数を決めます。Columns プロパティには、Visual Basic では PrintPreviewDialog1.PrintPreviewControl.Columns で、Visual C# では printPreviewDialog1.PrintPreviewControl.Columns で、Visual C++ では printPreviewDialog1->PrintPreviewControl->Columns でアクセスできます。

## 第 44 章

# ファイル保存ダイアログ

### 44.1 概要

ファイル保存ダイアログは、ユーザーがドライブやディレクトリ、ファイルやファイル群の名前を指定して、ファイルを保存することができるようにするダイアログである。

プラットフォーム	ツールキット	コントロール
Windows	Windows Controls .Net Windows Forms Visual Basic 6	Open and Save Dialog Boxes SaveFileDialog CommomDialog
Mac OS X	Carbon Cocoa	NSSavePanel
X11R6	GTK+ GNOME Qt KDE	GtkFileSelection QFileDialog
Java	AWT Swing SWT	JFileChooser
Mozilla	Gecko	
Web	HTML XForms	

### 44.2 Windows Controls: Open and Save Dialog Boxes

#### 44.2.1 説明

Open and Save Dialog Boxes コントロールは、ユーザーがドライブやディレクトリ、ファイルやファイル群の名前を指定して、ファイルを開いたり、保存することができるようにする。

## 44.3 .NET: SaveFileDialog

### 44.3.1 説明

SaveFileDialog は、ユーザーがファイルを保存できるダイアログ ボックスを表示する。

以下、.NET Framework 開発者ガイドより抜粋。

Windows フォームの SaveFileDialog コンポーネントは、定義済みダイアログ ボックスです。このダイアログ ボックスは、Windows で使用される標準の [ファイルの保存] ダイアログ ボックスと同じダイアログ ボックスです。SaveFileDialog コントロールは、CommonDialog クラスを継承しています。

このコントロールは、独自のダイアログ ボックスを設定せずにファイルの保存を行うための簡易ソリューションとして使用します。Windows の標準のダイアログ ボックスを使用して、一般的な基本機能を持つアプリケーションを作成できます。ただし、SaveFileDialog コンポーネントを使用する場合は、独自のファイル保存ロジックを記述する必要があります。

実行時にダイアログ ボックスを表示するには、ShowDialog メソッドを使用します。読み取り/書き込みモードでファイルを開くには、OpenFile メソッドを使用します。

## 44.4 Visual Basic 6: CommomDialog

### 44.4.1 説明

CommomDialog は、ファイルを開く、ファイルを保存する、印刷オプションを設定する、色を選択する、フォントを指定するなどの標準的な操作のためのダイアログ ボックスである。

## 44.5 GTK+: GtkFileSelection

### 44.5.1 説明

GtkFileSelection は、ユーザによるファイル名やディレクトリ名の入力待ちを行うユーザーインタフェース要素である。

## 44.5.2 スクリーンショット

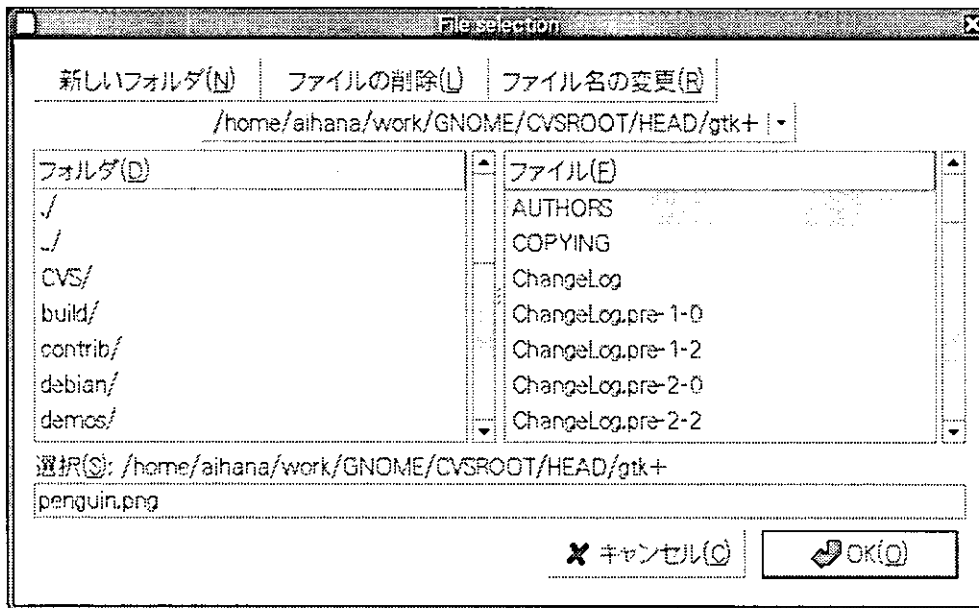


図 44.1 GtkFileSelectionDialog の実行例 (GTK+ 2.0 チュートリアルより)

## 44.6 Qt: QFileDialog

## 44.6.1 説明

QFileDialog は、ユーザーにファイルとディレクトリを選択させるダイアログである。Windows では、プラットフォーム固有のダイアログを表示することもできる。

## 44.6.2 スクリーンショット

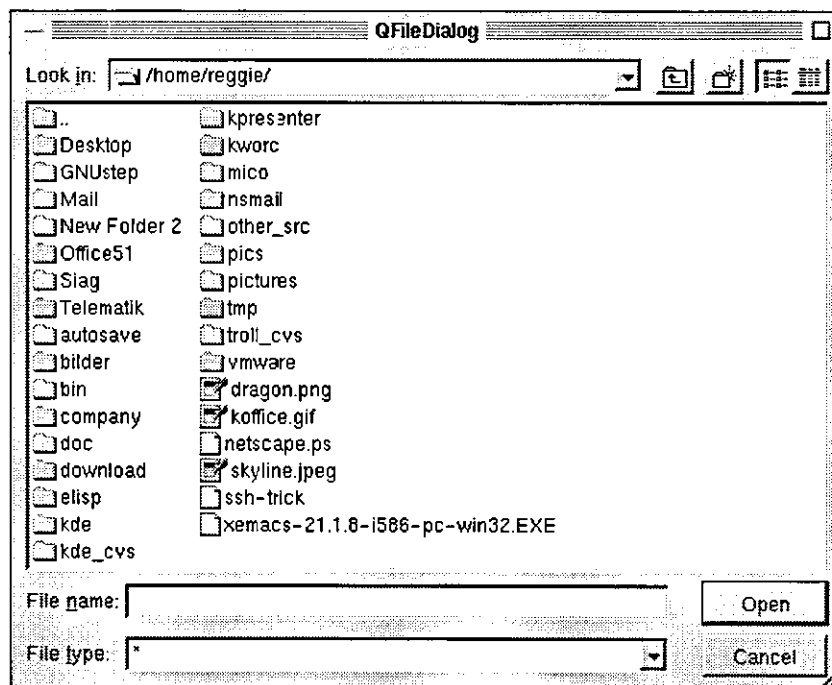


図 44.2 QFileDialog の Linux 上での実行例 (Qt のリファレンスより抜粋)

## 44.7 Swing: JFileChooser

### 44.7.1 説明

JFileChooser には、ユーザがファイルを選択するための簡単な機構が用意されている。

### 44.7.2 スクリーンショット

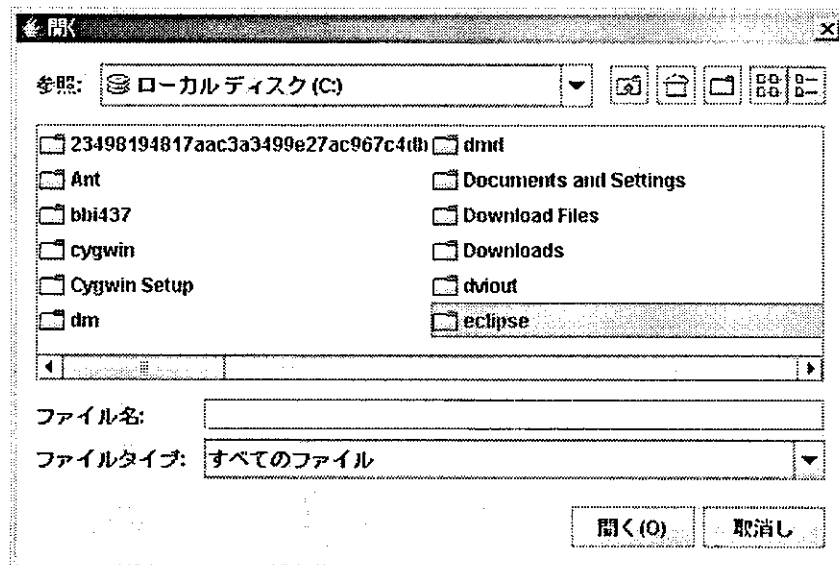


図 44.3 JFileChooser の実行例 (SwingSet2 の実行画面より)



## 第 45 章

# メッセージボックス

### 45.1 概要

プラットフォーム	ツールキット	コントロール
Windows Windows Controls	.Net Windows Forms Visual Basic 6	MessageBox MsgBox
Mac OS X	Carbon Cocoa	NSAlertPanel
X11R6	GTK+ GNOME Qt KDE	GtkMessageDialog GnomeMessaegBox QMessageBox
Java	AWT Swing SWT	JOptionPane
Mozilla	Gecko	
Web	HTML XForms	

### 45.2 .NET: MessageBox

#### 45.2.1 説明

メッセージボックスを表示する。メッセージボックスには、テキスト、ボタン、およびユーザーに情報や指示を伝えるための記号を格納できる。

以下、.NET Framework 開発者ガイドより抜粋。

メッセージボックスを表示します。メッセージボックスには、テキスト、ボタン、およびユーザーに情報や指示を伝えるための記号を格納できます。

MessageBox クラスの新しいインスタンスは作成できません。メッセージボックスを表示するには、静的 (Visual Basic では Shared) メソッドである MessageBox.Show を呼び出します。メッセージボックスに表示

されるタイトル、メッセージ、ボタン、およびアイコンは、このメソッドに渡すパラメータによって決定されます。

## 45.3 Visual Basic 6: MsgBox

### 45.3.1 説明

MsgBox は、ダイアログ ボックスにメッセージを表示し、ボタンがクリックされるのを待って、どのボタンがクリックされたのかを示す値を返す。

## 45.4 GTK+: GtkMessageDialog

### 45.4.1 説明

GtkMessageDialog は、エラーや質問などのメッセージをイメージとともに表示するダイアログである。

### 45.4.2 スクリーンショット

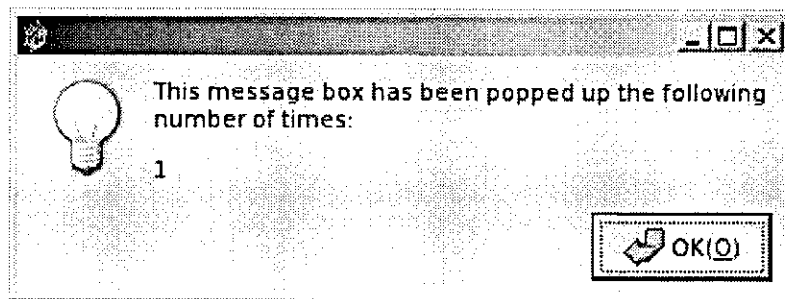


図 45.1 GtkMessageDialog の実行例 (gtk-demo の実行結果より)

## 45.5 GNOME: GnomeMessaegBox

### 45.5.1 説明

## 45.6 Qt: QMessageBox

### 45.6.1 説明

QMessageBox は、短いメッセージ、アイコン、いくつかのボタンを持つモーダルダイアログである。

## 45.6.2 スクリーンショット

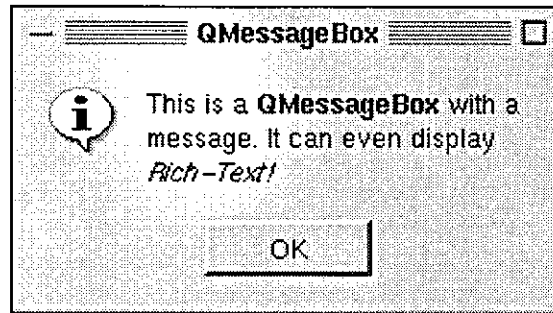


図 45.2 QMessageBox の実行例 (Qt のリファレンスより抜粋)

`JOptionPane.showMessageDialog`

## 45.7 Swing: JOptionPane

### 45.7.1 説明

`JOptionPane` は、ユーザに値の入力を求めたり、情報を提示したりするために、標準のダイアログボックスを表示する。

`showConfirmDialog` メソッドは `yes/no/cancel` 等の確認の要求に使う。

`showInputDialog` メソッドは入力プロンプトに使う。

`showMessageDialog` メソッドは起きた事情をユーザへメッセージとして伝えるために使う。

`showOptionDialog` メソッドはこれら 3 つを組合わせたものを表示するために使う。

### 45.7.2 スクリーンショット

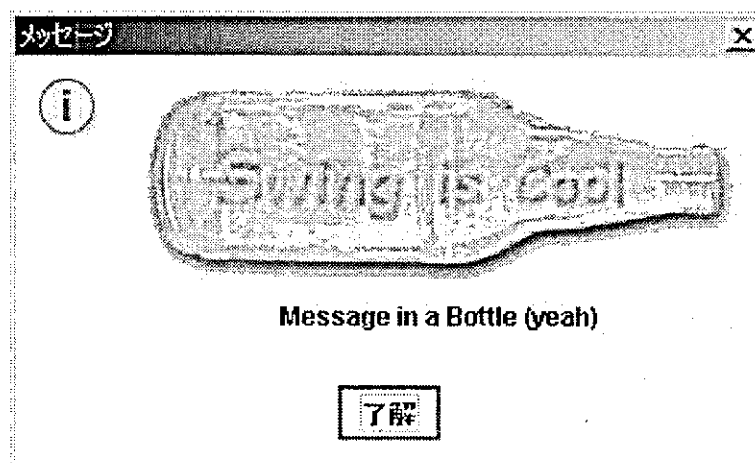


図 45.3 JOptionPane (`showMessageDialog` メソッド) の実行例 (SwingSet2 の実行画面より)

## 第 XV 部

# メニューコントロールの比較

## 第 46 章

### メニュー

#### 46.1 概要

プラットフォーム	ツールキット	コントロール
Windows	Windows Controls .Net Windows Forms Visual Basic 6	MainMenu Menu
Mac OS X	Carbon Cocoa	NSMenu
X11R6	GTK+ GNOME Qt KDE	GtkMenuBar QMenuBar
Java	AWT Swing SWT	JMenuBar
Mozilla	Gecko	
Web	HTML XForms	

#### 46.2 .NET: MainMenu

##### 46.2.1 説明

MainMenu は、メニューを作成するための設計時のインターフェイスを提供する。

以下、.NET Framework 開発者ガイドより抜粋。

Windows フォーム MainMenu コンポーネントを使用すると、実行時にメニューが表示されます。メインメニューに属するすべてのサブメニューおよび各項目は MenuItem オブジェクトです。

DefaultItem プロパティを真 (true) に設定することによって、メニュー項目を既定の項目として指定できます。既定の項目は、メニューをクリックしたときに太字のテキストで表示されます。メニュー項目の Checked プロパティは、真 (true) または偽 (false) のいずれかの値となり、メニュー項目が選択されているかどうかを示します。メニュー項目の RadioCheck プロパティでは、選択した項目の表示形式をカスタマイズしま

す。RadioCheck を真 (true) に設定している場合は、その項目の隣りにオプション ボタンが表示されます。RadioCheck を偽 (false) に指定している場合は、その項目の隣りにチェック マークが表示されます。

## 46.3 Visual Basic 6: Menu

### 46.3.1 説明

Menu を使用すれば、アプリケーションにそのアプリケーション固有のメニューを加えることができる。

## 46.4 GTK+: GtkMenuBar

### 46.4.1 説明

GtkMenuBar は、たくさんのメニューアイテムを保持する標準的なメニューバーである。また、外観の美しさを考慮するために影の種類を設定できる。

### 46.4.2 スクリーンショット

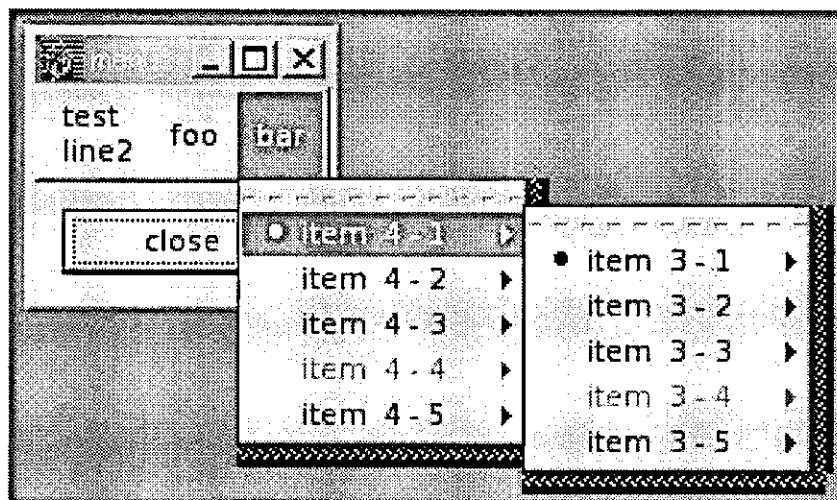


図 46.1 GtkMenuBar の実行例 (gtk-demo の実行結果より)

## 46.5 Qt: QMenuBar

### 46.5.1 説明

QMenuBar は、水平なメニューバーを提供するユーザーインターフェース要素である。メニューバーはサブメニューアイテムとしてプルダウンメニューを持つ。

## 46.5.2 スクリーンショット

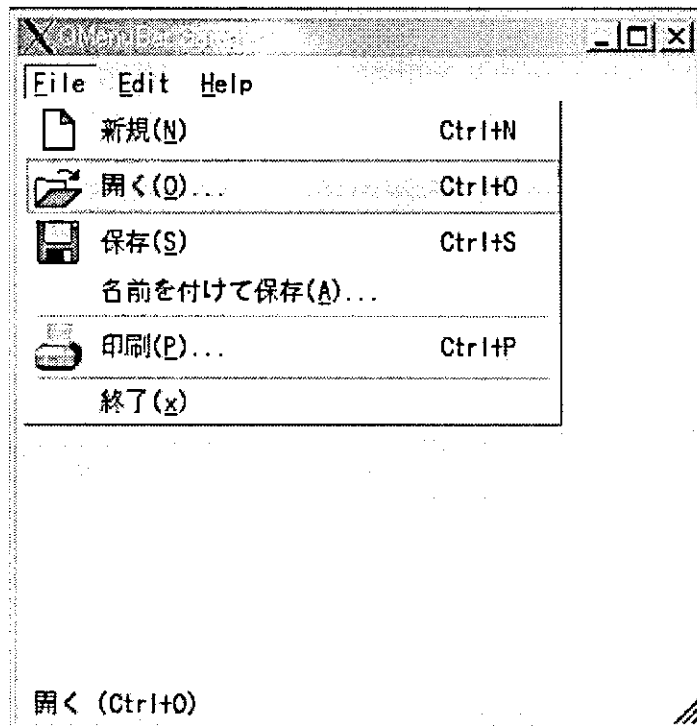


図 46.2 QMenuBar の実行例

## 46.6 Swing: JMenuBar

## 46.6.1 説明

JMenuBar はメニューバーの実装である。

## 第 47 章

# コンテキストメニュー

### 47.1 概要

プラットフォーム	ツールキット	コントロール
Windows	Windows Controls .Net Windows Forms Visual Basic 6	ContextMenu Menu
Mac OS X	Carbon Cocoa	NSMenu
X11R6	GTK+ GNOME Qt KDE	GtkOptionMenu QPopupMenu
Java	AWT Swing SWT	JPopupMenu
Mozilla	Gecko	
Web	HTML XForms	

### 47.2 .NET: ContextMenu

#### 47.2.1 説明

ContextMenu は、ユーザーがオブジェクトを右クリックしたときに表示されるメニューを実装する。

以下、.NET Framework 開発者ガイドより抜粋。

Windows フォームの ContextMenu コンポーネントは、選択したオブジェクトに関連付けられたコマンドのうち、頻繁に使用されるコマンドを簡単に実行するためのメニューを提供します。コンテキストメニューの項目は、多くの場合、アプリケーションのほかの場所に表示されるメインメニューの項目のサブセットです。通常、コンテキストメニューはマウスの右クリックで表示されます。Windows フォームでは、コンテキストメニューがコントロールに関連付けられています。



コンテキストメニューをコントロールに関連付けるには、コントロールの `ContextMenu` プロパティに `ContextMenu` コンポーネントを設定します。1 つのコンテキストメニューを複数のコントロールに関連付けることができますが、各コントロールに設定できるコンテキストメニューは 1 つだけです。

`ContextMenu` コンポーネントの主要なプロパティは、`MenuItems` プロパティです。メニュー項目を登録するには、プログラムで `MenuItem` オブジェクトを作成してコンテキストメニューの `MenuItems` コレクションに追加します。コンテキストメニューの項目は、通常、ほかのメニューを基に作成するため、ほかのメニュー項目をコピーしてコンテキストメニューに追加するのが一般的な方法です。メニュー項目は、無効または非表示にしたり、削除したりできます。

## 47.3 Visual Basic 6:Menu

### 47.3.1 説明

`Menu` はアプリケーションに独自のメニューを加えることができる。

## 47.4 GTK+: GtkOptionMenu

### 47.4.1 説明

`GtkOptionMenu` は選択リストからアイテムを選ぶ際に使用するユーザーインターフェース要素である。

## 47.5 Qt: QPopupMenu

### 47.5.1 説明

`QPopupMenu` は選択リストからアイテムを選ぶ際に使用するポップアップメニューのユーザーインターフェース要素である。

## 47.6 Swing: JPopupMenu

### 47.6.1 説明

`JPopupMenu` はポップアップメニューの実装である。

## 第 XVI 部

# コマンドコントロールの比較

## 第 48 章

# ボタンコントロール

### 48.1 概要

プラットフォーム	ツールキット	コントロール
Windows	Windows Controls .Net Windows Forms Visual Basic 6	Button Button CommandButton
Mac OS X	Carbon Cocoa	NSButton
X11R6	GTK+ GNOME Qt KDE	GtkButton  QPushButton KPushButton KIconButton KKeyButton
Java	AWT Swing SWT	JButton
Mozilla	Gecko	
Web	HTML XForms	<INPUT TYPE="button"> <xforms:submit> <xforms:trigger>

### 48.2 Windows Controls: Button

#### 48.2.1 説明

Button controls はユーザーがコントロールを選択した時に親ウィンドウに通知する時に使用するコントロールである。

## 48.3 .NET: Button

### 48.3.1 説明

Button クラスは Windows のプッシュボタンを操作するプロパティ、メソッド、イベントを持つ。以下、.NET Framework 開発者ガイドより抜粋。

Windows フォームの Button コントロールを使用すると、ユーザーはボタンをクリックしてアクションを実行できます。このボタンをクリックすると、ボタンを実際に押したときと同じように表示されます。ユーザーがボタンをクリックするたびに、Click イベントハンドラが呼び出されます。選択したアクションを実行するように、Click イベントハンドラにコードを割り当てます。

ボタン上に表示されるテキストは、Text プロパティに含まれています。テキストの長さがボタンの幅を超える場合は、次の行に折り返されて表示されます。ただし、コントロール内に全体の高さが収まらない場合は、テキストがクリッピングされます。Text プロパティには、アクセス キーを含めることができます。ユーザーは、Alt キーを押しながらアクセス キーを押して、コントロールを "クリック" するのと同じ操作ができます。Button コントロールでは、Image プロパティと ImageList プロパティを使用してイメージを表示することもできます。

## 48.4 Visual Basic 6: CommandButton

### 48.4.1 説明

CommandButton は Windows のプッシュボタンを操作するプロパティ、メソッド、イベントを持ち、処理の開始、中断、終了などを指示するのに使用する。プッシュ ボタンとも呼ぶ。

## 48.5 GTK+: GtkButton

### 48.5.1 説明

GtkButton はクリックするとシグナルを生成するユーザーインターフェース要素である。

### 48.5.2 スクリーンショット

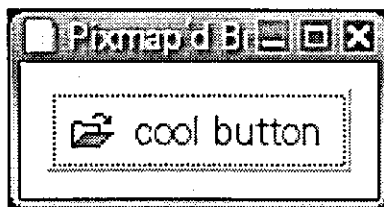


図 48.1 GtkButton の実行例 (GTK+ 2.0 チュートリアルより)