メニューイベント

メニュー項目のイベント処理

前回まで、メニューの表示に関する技術を紹介してきましたが
もちろん、メニューは表示するだけでは何の役にも立ちません
メニューをユーザーが選択した時のイベントを処理して、初めて機能が発揮されます

MenuItemクラスのイベント設計も、当然 Controlクラスと同じ形です
On***()というプロテクトメソッドが呼び出され、その後対応したイベントハンドラが実行されます
通常、MenuItemを継承してイベントを実装するという方法は考えられないので
この場では、イベントメンバによるイベント実装方法のみを解説します

ユーザーがメニューを選択すれば **MenuItem.Click** イベントが
ポップアップが表示される時は **MenuItem.Popup** イベントが
カーソルがメニュー項目にセットされた時は **MenuItem.Select** が呼び出されます

public event EventHandler Click;
public event EventHandler Popup;
public event EventHandler Select;

これらのイベントは、メニュー項目ごとに設定することができるため
それぞれに専用のイベントハンドラを割り当ててもかまいませんし
あるいは、イベント発生元のオブジェクトを割り出して、処理を分岐させるのもよいでしょう

```
using System.Windows.Forms;

class WinMain : Form {
 public static void Main(string[] args) {
  Application.Run(new WinMain());
 }

 public void MenuClick(object obj , System.EventArgs e) {
  MenuItem mi = (MenuItem)obj;
  mi.Checked = !mi.Checked;
 }

 public WinMain() {
  MainMenu mm = new MainMenu();
  MenuItem[] mi = {
   new MenuItem("Rena") , new MenuItem("Yuki") ,
   new MenuItem("Mimi")
  };

  for (int i = 0 ; i < mi.Length ; i++)
   mi[i].Click += new System.EventHandler(MenuClick);

  mm.MenuItems.Add("Kitty on your lap" , mi);

  Menu = mm;
 }
}
```

このプログラムは、メニュー項目を選択するとチェックを付けたり、外したりします
この動作は、MenuClick イベントハンドラの mi.Checked = !mi.Checked という文で実現しています

前のページへ　戻る　次のページへ