

# PV178: Programming for .NET Framework

## Globalization, Windows Forms

Vojtěch Forejt, [forejt@fi.muni.cz](mailto:forejt@fi.muni.cz)  
Martin Osovský, [osovsky@ics.muni.cz](mailto:osovsky@ics.muni.cz)

**Faculty of Informatics and Institute of Computer Science**  
Masaryk University

March 5, 2009

# Lecture Overview

- Globalization
- Windows Forms

# Example

## ■ ComparingExample.cs

# Key Terms

- Internationalization – the way application handles international data
  - Character encoding, date and time, numeric and currency format.
- Localization – the proces of adapting the application for specific local market

# System.Globalization Namespace

Includes classes for

- comparing strings with culture awareness  
ABAA < ABBA (ordinal)  
coté < côte (culture dependent)
- Date Time formats  
yy/mm/dd vs. dd/mm/yy
- numeric format  
12,000.00 vs. 12.000,00
- calendars  
Gregorian and non-Gregorian

# CultureInfo Class

- basic provider of cultural preferences
- access to culture-specific instances of `DateTimeFormatInfo`, `NumberFormatInfo`, `CompareInfo`, and `TextInfo`
- each `Thread` object has two `CultureInfo` properties
  - `Thread.CurrentCulture` (Date and number formatting  
String comparison and casing  
Replaces and extends the locale ID in Win32)
  - `Thread.CurrentUICulture` (resource selection for UI)

# Culture Names

- Culture name structure:

`<language>-<country/region>-[Cyril/Latn]`

- language code: 2 or 3 lowercase letters (en, cs, div)
- country/region code: 2 uppercase letters (US, CZ)
- Cyril/Latn denotes the alphabet used
  
- cs-CZ, en-US, az-AZ-Cyril...

# Cultures

- Invariant culture (identified by empty string (`String.Empty`))
- Neutral culture (identified by language name)
  - fr
  - Influences only resources selection
  - No formatting information
  - `CurrentUICulture`
- Specific culture
  - fr-CA, fr-FR
  - Both resources and formatting
  - Both `CurrentCulture` and `CurrentUICulture`



# Classes and Namespaces That Use Culture

- `System.String`
- `System.Globalization.StringInfo`
- `System.Globalization.Calendar`
- `System.Resources`
- `System.DateTime`
- ...

# Example

- `CalendarExample.cs`

# Resources

- Allow to store other kinds of data besides executable code
- Strings, multimedia required by application (icons, sounds)
- Use when you need to distribute data with application
- Are **localised**, you may use different versions for different cultures.
- Can be strongly typed

# Example

- ResourceExample

# Delegates

- Class encapsulating (static or instance) method
- Type-safe
- Contains
  - pointer to method
  - pointer to instance (or `null` if method is static)
  - pointer to the next delegate (if any)
- Linked list of pointers to methods of the same type.

# Example

- DelegatesExample

# Events

- Allow objects to notify other object when something occurs (mouse click, synchronization)
- Defined using `event` keyword
- E.g. mouse click, thread synchronization
- Encapsulate delegates of type

```
delegate void Name(object sender , EventArgs e)
```

- Unlike delegate...
  - event can be raised only from class that declared it.
  - event can be included in interface declaration.

# Example

- EventsExample



# Windows Forms – Overview

- Most widely used .NET API for graphical UI
- Wrapper around Windows API
- Features:
  - Component model, versioning, licensing
  - Globalization
  - Rich design time support in Visual Studio

# Windows Forms – Classes

- Controls (widgets) known from Windows  
  TextBox, ComboBox...
- Menus, toolbars
- Data binding (ADO.NET) DataGridView...
- Layout control classes TableLayoutPanel,  
  FlowLayoutPanel...
- Predefined dialog boxes PrintDialog, OpenFileDialog...
- ActiveX

# Basic Notions

- Component
- Control
- Form
- Property
- Event

# Component

- Atomic element of graphical interface
- Configurable objects with certain properties
- Support for configuration in Visual Studio
- Visual vs. non-visual

# Control

- Component which
  - has visual representation
  - allows user interaction
- `System.Windows.Forms.Control` class

# Form

- Window within an application.
- Control that serves as a container for other components.
- Can be displayed using the following methods
  - `Show()` – non modal, no relation with the currently active form
  - `ShowDialog()` – modal, the form is owned by calling form.

# Events

- Run by components...
  - as a response to user interaction  
KeyUp,MouseMove, DragDrop...
  - in certain states of the component life-cycle  
Load, Closing...
  - as a response to changed states or performed tasks  
Invalidated, Paint...

# Form – Life Cycle

- Constructor (it calls `InitializeComponets`)
- Before showing the form, the `Load` event is raised
- `Activated` (also raised when user switches to the form)
- `Showed`
- ...
- `FormClosing`
- `FormClosed`
- `Deactivated`



# Example

- FormEventsExample

# Positioning of Controls

- “Fixed”: Location and Size properties
- “Dynamic”: Anchor and Dock
- Controls used for positioning:
  - Panel, SplitContainer, GroupBox, TabControl
  - Properties similar to forms (anchoring, docking...).

# Positioning of Controls cont.

- `FlowLayoutPanel`
  - Child controls positioned in specified order  
`LeftToRight`, `TopDown`, ...
  - When panel is resized, the components are moved while preserving the order.
- `TableLayoutPanel`
  - Child controls ordered in rows and columns
  - Every row and column has either absolute size, or size in percents
  - Every cell contains at most 1 control
  - Controls may span more cells (`RowSpan`, `ColumnSpan`)

# Example

- FormEventsExample

# Common Controls

- Button
- Label
- TextBox
- CheckBox
- RadioButton
- ListBox
- ComboBox
- ProgressBar
- DateTimePicker

# Common Controls

- Button
- Label
- TextBox
- CheckBox
- RadioButton
- ListBox
- ComboBox
- ProgressBar
- DateTimePicker
- MaskedTextBox – Input of text data in certain format
- ListView – Showing collection items in four different ways
- TreeView – Showing tree-like data
- RichTextBox – Showing and entering formatted data
- NumericUpDown – Number input
- CheckedListBox – ListBox with checkboxes before items
- NotifyIcon – Shows icon in the taskbar notify area
- WebBrowser – Allows to show web pages in an application
- DataGridView – Display table with data

# Localization

- Using resources
- Rich support in Visual Studio designer

# Example



# WinForms Topics *Not* Covered Here

- Data binding
- Visual inheritance
- Drag and drop
- Validation