

OUTLINE

DELPHI 2010: OBJECT-ORIENTED DEVELOPMENT WITH DELPHI

This seminar is designed to provide developers with a foundation in the principles and techniques of object-oriented programming using Delphi and application development in Delphi. Techniques that are used in almost every Delphi application are discussed, including how to configure components, create event handlers, as well as efficient use of forms (windows). Next, focus turns to more advanced component design issues including design-time behavior, publishing subcomponents, message handling, and creating data-aware components. Finally, this seminar finishes with property editors and creating component editors. Delphi tips and tricks are included throughout the seminar.

CARY JENSEN, PH.D.

COPYRIGHT 2009-2010. CARY JENSEN. JENSEN DATA SYSTEMS, INC.
ALL RIGHTS RESERVED.

JENSEN DATA SYSTEMS, INC.

HTTP://WWW.JENSEN DATASYSTEMS.COM

PHONE: 281-359-3311

EMAIL: INFO@JENSEN DATASYSTEMS.COM

COURSE INCLUDES A COURSE BOOK PLUS ALL CODE EXAMPLES PRESENTED ALONG WITH ADDITIONAL CODE EXAMPLES.

DELPHI 2010: OBJECT-ORIENTED DEVELOPMENT WITH DELPHI

INTRODUCTION TO CLASS DESIGN

FROM RECORD TO CLASS

ENCAPSULATION AND MEMBER VISIBILITY

Private Members

Protected Members

Public Members

Published Members

DEFINING THE RUNTIME INTERFACE

INHERITANCE AND POLYMORPHISM

Reusing Code Through Inheritance

Generic Object Reference Through Polymorphism

CREATING AND TESTING A CLASS

GENERAL STEPS IN CLASS DESIGN

The Ancestor

Properties

Methods

Events

CREATING THE NEW CLASS

DEFINING PROPERTIES AND THEIR MEMBER FIELDS

INITIALIZING A CLASS

TESTING A NEW CLASS

OVERRIDING METHODS

Static Methods

Abstract Methods

Virtual and Dynamic Methods

Class Methods

CREATING EVENT PROPERTIES

FINISHING THE COMPONENT: ATTENTION TO DETAIL

Synchronizing Property Values

Writing Better Event Properties

INSTALLING THE COMPONENT

ADVANCED COMPONENT DESIGN ISSUES

DESIGN-TIME BEHAVIOR

Detecting Design Time

PUBLISHING SUBCOMPONENTS

COMPONENT MESSAGE HANDLING

Delphi's Message Handling

How Messages Are Associated with Methods

A Message Handling Example

CREATING DATA-AWARE COMPONENTS

Using Datalinks

Packages with Data-Aware Controls

PROPERTY EDITORS

DERIVING A PROPERTY EDITOR

OVERRIDING INHERITED METHODS

REGISTERING A PROPERTY EDITOR

ADDING A PROPERTY EDITOR TO THE COMPONENT LIBRARY

REPLACING A DEFAULT PROPERTY EDITOR

CREATING COMPONENT EDITORS

DECLARING THE NEW COMPONENT EDITOR

IMPLEMENTING OVERRIDDEN METHODS

REGISTERING A COMPONENT EDITOR