

# Windows Workflow Foundation Using Visual C# 2005

page 1

**Meet the expert:** Ken Getz is a featured instructor for several of our Visual Studio courses. He is a Visual Basic and Visual C# expert and has been recognized multiple times as a Microsoft MVP. Ken is a seasoned instructor, successful consultant, and the author or co-author of several best-selling books. He is a frequent speaker at technical conferences like Tech-Ed, VSLive, and DevConnections and he has written for several of the industry's most-respected publications including Visual Studio Magazine, CoDe Magazine, and MSDN Magazine.

Robert Green is a Visual Studio expert and a featured instructor for several of our Visual Basic and Visual C# courses. He is currently a Technical Evangelist in the Developer Platform and Evangelism (DPE) group at Microsoft. He has also worked for Microsoft on the Developer Tools marketing team and as Community Lead on the Visual Basic team. Robert has several years of consulting experience focused on developer training and is a frequent speaker at technology conferences including TechEd, VSLive, VSConnections, and Advisor Live.

**Runtime:** 11:59:13

**Course description:** Microsoft has provided the Windows Workflow Foundation, a set of assemblies based on the .NET Framework 2.0 (as part of the .NET Framework 3.0), along with a full-featured workflow designer built on top of Visual Studio 2005, and it's free. Windows Workflow Foundation makes it easy to create applications that follow a flow, or react to events and move from state to state. By providing a visual interface for designing the applications, and by binding code activities directly to the design of the application, Windows Workflow Foundation makes it easier than ever to create and maintain complex business applications. In addition, Windows Workflow Foundation provides a runtime engine that manages your application's behavior (that is, the "plumbing") and enforces the workflow. The runtime makes sure the workflow behaves in the manner in which you designed and coded it.

## Course outline:

### Workflow Foundation

- Introduction
- Overview: WF
- What App Types Benefit
- What is Needed for WF
- Verify Pieces are Installed
- Two Types of Workflows
- Base Classes
- Activity Controls
- Create First Workflow
- Workflow Activities
- Add Code Activities
- Test the Workflow
- Debug Your Workflow
- Workflow Architecture
- Types of Host Apps
- Workflow Startup Code
- Two Ways of Create Workflows
- Create Workflow w/XOML
- Summary

### Windows Form Host

- Introduction
- Hosting Workflow
- Passing Parameters
- Create the Host App

- Create the Workflow
- Add Amount/ItemName
- Set Delay Activity Behavior
- Add IfElse Activity Condition
- Add Code Activities
- Create the Form
- Create Parameters Coll.
- WorkFlowCompleted
- Test the Workflow
- Summary

### If / Else / While Conditions

- Introduction
- Overview: Conditions
- IfElse Activity
- Add/Remove IfElse Branches
- Use Conditional Expression
- Use Methods as a Condition
- Demo: IfElse Activity
- Demo: Conditional Expression
- Demo: Add/Remove Branches
- Demo: Use Code Conditions
- While Activity
- Demo: Use While Activity
- Summary

### CAG / Replicator

- Introduction

- Overview: CAG
- Demo: CAG Activity
- Replicator Activity
- Demo: Replicator Activity
- Using a Custom Activity
- Demo: Custom Activity
- Summary

### Exception Handling

- Introduction
- Overview
- Use FaultHandler Activity
- Use Throw Activity
- Demo: Exceptions
- OnWorkflowTerminated Event
- Demo: FaultHandler Activity
- Demo: Throw Activity
- Summary

### Delay Activity

- Introduction
- Overview: Delay Activity
- TimeoutDuration
- Why Bind?
- Work with Delay Activity
- Work w/Enabled Property
- Set the Description Prop.
- Set the TimeoutDuration

- Set InitializeTimeoutDuration
- Bind the TimeoutDuration
- Bind to a Field
- Bind to a Property
- Bind to Dependency Prop.
- Multiple Instances
- Summary

### Listen / Suspend / Terminate

- Introduction
- Listen Activity
- Demo: Listen Activity
- Review: Listen Activity
- Suspend and Terminate
- Suspend Activity
- Demo: Suspend Activity
- Terminate Activity
- Demo: Terminate Activity
- Summary

### Parallel

- Introduction
- Parallel Activity
- Demo: Parallel Activity
- Delay Activity
- Demo: Delay Activity
- Synchronize Access
- SynchronizationHandles

(Continued on page 2)

# Windows Workflow Foundation Using Visual C# 2005

page 2

- Demo: SynchronizeScope
- Handling Cancellation
- Use Multiple Handlers
- Demo: CancellationHandler
- Demo: Multiple Handler
- Summary

## Rule Sets

- Introduction
- Overview: Workflow Rules
- Using a RuleSet
- Using a Policy Activity
- Demo: Policy Activity
- Associate RuleSet
- Demo: Define Rules
- Call Methods in Rules
- Demo: Call Methods
- View RuleSet Editor
- Setting Priorities
- Demo: Set Priorities
- Add New Rules
- Evaluate Rule Order
- Summary

## Chaining

- Introduction
- Forward Chaining
- Demo: Forward Chaining
- Attribute Chaining
- Explicit Chaining
- Demo: Attribute Chaining
- Demo: Explicit Chaining
- Control Chaining
- Demo: Halt Chaining
- Summary

## Host Methods

- Introduction
- Overview: Host Methods
- Cross-Threading Issues
- Important Specifics
- Create a Console Host
- Create Shared Interface
- Create the Workflow
- Interface Implementation
- ExternalDataExchangeService
- Add Windows App Host
- Review: Host Methods
- Summary

## Host Events

- Introduction
- Overview: Host Events
- Event Arguments
- Create a Solution
- Add EventArgs Class
- Create the Interface

- Create the Host App
- Interface Implementation
- Hook up the Events
- Create the Workflow
- Raise the Event
- Bi-Directional Comm.
- Modify the Interface
- Modify the Implementation
- Modify the Workflow
- Summary

## State Machines

- Introduction
- Overview: State Machines
- State Activity
- SetState Activity
- StateInitialization Activity
- StateFinalization Activity
- EventDriven Activity
- Create a State Machine WF
- Create Workflow States
- Define State Transitions
- Create the EventArgs Class
- Create an Interface
- Create Host Application
- Implement Host Interface
- Examine the Form Code
- Complete the Workflow
- Run the Application
- Summary

## Work with State Machines

- Introduction
- Circular Navigation
- Demo: Circular Navigation
- Interact w/State Machine
- StateMachineWorkflowInst...
- Summary

## Intro to Persistence

- Introduction
- Overview
- Save / Restore State
- Examine Sample Workflow
- View Workflow Project
- View Host Application
- WorkflowRuntime Events
- Start Workflow Runtime
- Create / Unload Workflows
- Release Workflow
- View app.config
- Life without Persistence
- Unloading Workflows
- What About Timers
- Unload vs TryUnload

- SQL Persistence Service
- Setup SQL Persistence
- SqlWorkflowPersistence
- Automatic Unload on Idle
- Modify Load Interval
- Ownership Duration
- Hooking up Persistence
- Persist from Config File
- Run 2 Persistence Services
- Persistence / Serialization
- Summary

## Custom Persistence

- Introduction
- Overview: Persistence
- Demo: Persistence Service
- Create Persistence Service
- MyTimerList Class
- MyFilePersistenceService Cls.
- Handle Timer Expiration
- Create the Timer List
- Reload Persistence Service
- Summary

## Tracking Services

- Introduction
- Overview: Tracking Services
- SqlTrackingService
- Tracking Events
- Using SqlTracking Service
- Set up SQL Tracking
- Add Tracking to Workflow
- Query Data by Hand
- Query Data w/Code
- SqlTrackingWorkflowInstance
- Tracking Records
- Create Tracking Points
- Track Evaluation of Rules
- User Config. File
- Setup Config. File
- Summary

## Tracking Profiles

- Introduction
- Overview
- Default Tracking Profile
- Custom Tracking Profiles
- Tracking Profile in XML
- View Default Profile
- Create Profile in XML
- Tracking Profile in Code
- Define Workflow Events
- Define Activity Events
- Extract Workflow Fields
- Define User Events

- Save Profile to Database
- Display Tracking Data
- Track Profile Version
- Modify a Profile
- Delete a Profile
- Summary

## Custom Tracking Service

- Introduction
- Overview
- TrackingChannel Class
- TrackingService Class
- Create Custom Service
- Build Custom Service
- Test Custom Service
- Summary

## Intro to Custom Activities

- Introduction
- Custom Activities
- Why Create Activities?
- Simple vs. Composite
- Create Simple Activity
- Add Dependency Props
- wdp shortcut
- Add Functionality
- Test the Activity
- Dependency Properties
- Set Default Values
- Set the Display Name
- Summary

## More Complex Activity

- Introduction
- LoadTextFile Activity
- Test LoadTextFile Activity
- Add Toolbox Bitmap
- Demo: Toolbox Bitmap
- Add a Validator
- Demo: Add a Validator
- Add a Designer
- Demo: Add a Designer
- Modify Activity Theme
- Demo: Investigate Themes
- Demo: Create New Theme
- Summary

## Composite Activities

- Introduction
- Overview
- Create Composite Activity
- Add DisplayFileToConsole
- Use Property Promotion
- Demo: Property Promotion
- Summary

## Transactions

- Introduction
- Overview: Transactions

(Continued on page 3)

# Windows Workflow Foundation Using Visual C# 2005

page 3

- Transactions in Workflows
- TransactionScope Activity
- Transactions in Workflows
- Transaction Batches
- Demo: Using Batches
- Summary

## Compensation

- Introduction
- Overview
- CompensatableTransaction...
- Demo: CompensatableTran...
- Compensate Activity
- Demo: Compensate
- CompensatableSequence
- Demo: CompensatableSeq...
- Summary

## Dynamic Updates

- Introduction
- Overview
- Perform Dynamic Updates
- Update from Workflow
- Update from Host App
- Steps to Update a Workflow
- Restrict Updates
- Modify from Host App
- Investigate Workflow
- Investigate Host Application
- Add an Activity
- Delete Activity
- Restrict Workflow Changes
- Modify from Workflow
- Re-Investigate Workflow
- Re-Investigate Host App
- Summary

## Dynamic Rule Conditioning

- Introduction
- Overview: DRC
- Examine .rules File
- Modify Rule Condition
- Demo: Modify Rule
- Replace All Rule Conditions
- Demo: Replace All Rules
- Summary

## Workflows as Web Services

- Introduction
- Overview
- WebServiceInput Activity
- WebServiceOutput Activity
- Build a Web Service
- Create the Interface
- Create the Workflow
- Publish Web Service
- Test the Web Service

- WebServiceFault Activity
- Add Validation
- Add a Win App Host
- Test Win App Host
- Summary

## Maintaining State

- Introduction
- Overview
- Add WS Input/Output
- Test Web Service
- Modify Client Form
- Run the Web Service
- Maintain Web Service State
- Store State On Client
- Summary

## Invoking Web Services

- Introduction
- Invoke a Web Service
- Demo: Invoke a WS
- Demo: Add Fault Handling
- Summary