

## InstallShield

**Length:** 5 Days

**Summary:** This InstallShield course provides students with a solid understanding of installation terminology, the tasks an installation software program needs to perform, and the InstallShield development environment.

This course teaches students how to create Basic MSI projects, which use the Windows Installer service (MSI) for the installation's behavior and appearance. Students will learn how to customize installation with custom actions in order to make changes to their end users' systems. Students will also see how to correctly identify, install, and configure special types of files (e.g., COM servers, Windows services, and .NET assemblies). Finally, students will understand how to perform simple to sophisticated changes to your installation's user interface.

**Prerequisites:** Previous InstallShield experience is not required; however, attendees should be comfortable with:

- Using Windows Explorer and the command prompt to create and manage files, and run executables
- Basic Windows terminology (e.g. DLL, Control Panel, etc.)

**Objectives:** Attendees will learn how to:

- Create and organize an installation project
- Install files and create shortcuts
- Install and configure special types of files and data, such as registry and INI data, COM servers, Windows services, and .NET assemblies
- Work with MSI sequences and custom actions
- Customize your installer's user interface
- Package your Windows installer for Windows Vista, Windows 7, and later
- Use the Automation interface to modify your project outside the graphical environment

---

## COURSE CONTENT

### 1: INTRODUCTION

*You will begin by learning about the background and motivation for using the Windows Installer service and how InstallShield fits into the installation development process. You will also learn about the resources available as you work with InstallShield, including:*

- About the Windows Installer service
- About Windows Installer installation packages
- About the Windows Installer installation mechanism
- About the InstallShield environment
- Getting help

### 2: PROJECTS

*The foundation of your installation program is an InstallShield project. In this section, you will learn about general installation terminology and design issues and how to create a project that reflects the design of your product. In this section, you will learn about:*

- Installation terminology
- Using the Project Assistant
- Using the Installation Designer
- Working with features and components
- Global product and project data

### 3: INSTALLING FILES

*The primary job of your installation program is to transfer files to the user's system. In this section, you will learn about the different methods for adding files and folders to your installation project, including:*

- Static and dynamic file linking
- Working with destination folders
- Best Practice fundamentals
- "Key" resource usage
- GUID usage
- Restarts and the Windows Restart Manager
- Creating shortcuts
- Using merge modules and prerequisites

### 4: BUILDING RELEASES

*Before you can test your installation, you must build a set of disk images. In this section, you will learn how to use the InstallShield graphical tools and command-line tools for building release images for CD-ROM or DVD-ROM, a network location, or for the Web. In this section, you will learn about:*

- Using the Release Wizard
- Media types
- Web installations
- Command-line builds
- Using the InstallShield Automation interface
- Digitally signing the installation and files
- Using the Standalone build system

### 5: DEPLOYING AN INSTALLATION

*Windows Installer-based installations support several different installation modes. In this chapter, you will learn how to run an installation from Windows Explorer or from the command prompt, how to perform silent and limited-user interface installations, and how to pass information to the running installation from the command line. In this section, you will learn about:*

- Running an installation
- Understand how to interact with User Account Control (UAC)
- Maintenance and uninstallation
- Reducing installation time for large packages
- Using msixexec.exe
- Silent and reduced-UI installations
- Passing properties from the command line
- MSI Log Analyzer

### 6: NON-FILE SYSTEM CHANGES

*In addition to installing files, most installation programs need to make global changes to the user's operating system. In this section, you will learn how to install configuration data, such as:*

- Registry data
- Text files
- INI files, XML files, and arbitrary text files
- Environment variables
- File associations
- Setting file, folder, and registry permissions
- Scheduled tasks

### 7: INSTALLING SPECIAL FILES

*For most types of files, the installer's only responsibility is to transfer the files. However, some types of files require additional registration or configuration information to be written to the target system. In this section, you will learn how to install and configure the following types of files:*

- COM servers
- Windows services
- .NET Assemblies
- Running SQL scripts
- Forcing data removal

### 8: CREATING AND CONSUMING REUSABLE INSTALLATION ELEMENTS

*InstallShield provides methods for creating elements that can be used by multiple installation projects, and for combining multiple complete product installations. In this section, you will learn about:*

- Creating merge modules
- Creating InstallShield prerequisites
- Working with repositories
- Creating suite projects
- Chaining MSI installations

## 9: CONDITIONS AND ACTIONS

*Unlike traditional installations, Windows Installer installations do not use an explicit script, but instead perform actions arranged into your product's sequence tables. In this section, you will learn about:*

- MSI properties
- Condition syntax and examples
- Installation sequences
- Custom actions: action types and guidelines
- Getting and setting properties
- Launching executables
- Calling DLLs
- Managed code custom actions
- Editing MSI tables directly

## 10: USER INTERFACE

*Like standard Windows Installer actions, your installation's user interface is controlled by MSI tables and actions. In this section, you will learn about:*

- How MSI determines dialog order and conditions
- Working with control events
- Working with control conditions
- MSI Debugger
- Creating new dialog boxes
- Subscription to Actions
- Localization

## 11: UPDATES AND PATCHES

*Windows Installer supports different types of updates and different techniques for packaging them for distribution. In this section, you will learn about:*

- Types of updates
- Creating minor upgrades
- Creating major upgrades
- QuickPatch projects
- General patch projects
- Patches and User Account Control
- Uninstalling a patch

