

## Advanced ColdFusion Training: Components, Error Handling and Complex Objects

### Course Summary

#### Description

In this advanced ColdFusion training class, you will learn the many methods of reusing code (including `<cfinclude>`, custom tags, and stored procedures). You will also learn how to predict and prevent errors in ColdFusion so your users will never see ugly error pages. In addition, you will learn to use Regular Expressions to perform advanced form validation.

Class is conducted in 2016 but students using versions 9, 10, 11, or 2016 may attend.

#### Objectives

After taking this course, students will be able to:

- Learn to create and use complex data objects.
- Learn to write `<cfscript>` code.
- Learn to build and use Custom Tags.
- Learn to build and use ColdFusion Components (CFCs).
- Learn to handle errors properly.
- Learn to page through recordsets.
- Learn advanced form validation with regular expressions.

#### Topics

- Charting
- Complex Objects
- Components
- Application Framework
- Error Handling
- Reusing Code
- Regular Expressions (optional)

#### Audience

This course is designed for those wanting to learn many methods of reusing code (including `<cfinclude>`, custom tags, and stored procedures).

#### Prerequisites

Before taking this course, students are required to have experience with basic ColdFusion.

#### Duration

Two days

## Advanced ColdFusion Training: Components, Error Handling and Complex Objects

### Course Outline

- I. Charting**
  - A. Course Introduction and <cfchart>
    - 1. Introduction
    - 2. Course Objectives
    - 3. Course Prerequisites
    - 4. Course Project
  - B. Charting and Graphing Options
    - 1. Using <cfchart> vs. <cfchartseries>
    - 2. Caching Charts
- II. Complex Objects**
  - A. Manipulating Complex Data Objects (Lists, Arrays, Structures and Queries)
    - 1. Scalar Variables
    - 2. Lists
    - 3. Arrays
    - 4. Structures
  - B. Looping with <cfloop>
    - 1. Syntax of <cfloop>
    - 2. Weeding Out the FIELDNAMES Variable
    - 3. Using <cfloop> for automatic query output
    - 4. Display Web-safe Colors using CFLOOP
  - C. Queries of Queries
- III. Components**
  - A. CFCs - Building ColdFusion Components
    - 1. Benefits to using CFCs
    - 2. Creating Components
    - 3. Calling or Invoking Components
    - 4. Passing Arguments to Components
    - 5. The main CFML page
    - 6. Three Ways to Pass Arguments
    - 7. Where to save CFCs
    - 8. Self-documenting - View a Component directly in the browser
  - B. Object-Oriented CFCs
    - 1. Object-Oriented Principles
    - 2. How does use of a CFC compare to other OO languages?
    - 3. Inheritance
    - 4. Access control
    - 5. Exposing CFCs as Web Services
- IV. Application Framework**
  - A. Application Framework
    - 1. Automatic Includes: Application.cfm and onRequestEnd.cfm
    - 2. New Application Framework: Application.cfc
  - B. Using Application Variables
- V. Error Handling**
  - A. Errors and Exceptions
    - 1. Standard Error Handling
    - 2. Application.cfm
  - B. Structured Error Handling
    - 1. <cftry> and <cfcatch>
    - 2. Transactions with <cftransaction> and <cftry>
    - 3. Classic <cftransaction> example
- VI. Reusing Code**
  - A. Reusing Code
    - 1. Limited Capabilities of <cfscript>
    - 2. Syntax Rules
    - 3. Reserved Words
    - 4. Looping with <cfscript>
    - 5. GetTickCount() - testing the speed of your code
  - B. Custom Tags
    - 1. Where to store Custom Tags
    - 2. CALLER and ATTRIBUTE scopes
    - 3. Using <cfmodule>
  - C. Stored Procedures
    - 1. Benefits of Stored Procedures
  - D. Showing portions of a Recordset
    - 1. Paging through records
- VII. Regular Expressions (optional)**
  - A. Advanced Validation with Regular Expressions (optional)
    - 1. Special Symbols for Matching Specific Characters
    - 2. Specify the number of occurrences
    - 3. Placement of the pattern
    - 4. POSIX Syntax
    - 5. Perl Syntax