

Adobe® Certified Expert Program

Product Proficiency Exam Bulletin

Adobe® ColdFusion® 9

Exam # 9A0-127

ACE Certification Checklist

The checklist below will help guide you through the process of obtaining ACE certification.

- Review this bulletin to become familiar with the topic areas and objectives of the exam.
- Identify the topic areas and objectives you need to study.
- Determine which study materials you need to improve your skills.
- For a list of recommended study materials, visit:
<http://www.adobe.com/misc/training.html>
- Study for the exam.
- Contact Pearson VUE to register for an exam: *<http://www.pearsonvue.com/adobe>*
- When registering for the exam, refer to the exam number located at the top of this bulletin.
- Take the exam. Your score is electronically reported to Adobe and will also be displayed when you complete your exam.
- Upon passing the exam, your certificate will be sent to you via email. Please ensure that you have provided an accurate email address.
- You can update your certification profile by going to: <http://www.adobe.com/support/certification/community.html>. You will need the information provided on your score report to create a user account.

Exam Structure

The following lists the topic areas and percentage of questions delivered in each topic area:

Topic Area	% of Exam	# of Questions
Programming fundamentals	13%	8
Working with the Application framework	8%	5
Interacting with databases	13%	8
Working with components	14%	9
Interacting with persistent components	8%	5
Working with XML	7%	4
Reusing code	8%	5
Interacting with remote systems	7%	4
Managing files	5%	3
Tuning application performance	7%	4
Building advanced user interfaces	5%	3
Reporting	5%	3

Number of Questions and Passing Score

- 61 questions
- 64% minimum required to pass

Test Content: Topic Areas and Objectives

Following is a detailed outline of the information covered on the exam.

1. Programming fundamentals

- Given a type of variable, explain how to create and use that variable by using implicit and explicit syntax. (Variable types include: simple, list, array, structure; syntax includes using braces and brackets)
- Control the flow of a program by using conditionals and boolean expressions.
- Manage program iteration by using tags.
- Given a task, select and use the appropriate built-in function to perform that task.
- Understanding built-in ColdFusion Scopes. (Scopes include: application, session, client, URL, form, this, variables, local)
- Handle exceptions by using `<cftry />`, `<cfcatch />`, `<cfthrow />`, and `<cfrethrow />` `<cffinally />`.
- Handle exceptions by using `<cferror />`.
- Explain when and how to use regular expressions.

2. Working with the Application framework

- Handle exceptions by using the onError method of Application.cfc.
- Given an Application.cfc method identify the use of that method. (Methods include: onApplicationStart, onSessionStart, onRequestStart, onServerStart and onRequestEnd)
- List and describe the differences between application scope variables and Application.cfc variables.

3. Interacting with databases

- Create, read, update, and delete data by using <cfquery />.
- Create dynamic queries by using conditional clauses, variables, the like operator, and wildcards.
- Return data by performing a Query of Queries.
- List and describe the properties of the RESULT structure of a query object.
- List and describe the information returned by the <cfdbinfo> tag.
- Implement database transactions by using <cftransaction />.
- Use <cfqueryparam /> in SQL statements.
- Use <cfstoredproc />, <cfproccparam />, and <cfprocresult /> to interact with database stored procedures.

4. Working with components

- Reuse code by creating ColdFusion Components (CFCs).
- Control public or private access to CFC properties and methods by using scope prefixes for properties and the access attribute for methods.
- Understand the options available for instantiating CFCs and invoking CFC methods. (Options include: <cfobject />, <cfinvoke />, createObject(), new/import keyword).
- Explain how to expose a CFC method to a remote system.
- Understand the importance of var/local scoping.
- Understand inheritance.
- Understand the use of <cfproperty/>.
- Understand the use of dot notation and packaging.

5. Interacting with persistent components

- Create a persistent component.
- Create, read, update, and delete data by using persistent components.
- Build a persistent component based on one or more tables.
- Understand how relationships work between components.
- Manage transactions.
- List and describe the benefits of lazy loading.

6. Working with XML

- List and describe the differences between an XML document and an XML object.

- Parse XML data by using `<cfxml />` or `xmlParse()`.
- Extract an XML document from an XML object by using the `toString()` method.
- Extract data from an XML object.

7. Reusing code

- Include one page of code into another page of code by using the `<cfinclude />` tag.
- Given an approach, reuse code by using a custom tag. (Approaches include: using the `<cf_... />`, or `<cfmodule />` tags.
- Reuse code by creating user-defined functions. (User-defined functions written with either `<cffunction />` or within a `<cfscript />` block.

8. Interacting with remote systems

- Given a scenario, create a Web service proxy and interact with that Web service. (Scenarios include: Using the Server Administrator interface and/or `<cfinvoke />`).
- Create and read Atom and RSS feeds by using `<cffeed />`.
- Programmatically retrieve information from a remote server by using `<cfhttp />`.

9. 9.0 Managing files

- Upload, read, write, and delete files by using `<cffile />` and `file()`.
- Create directories on an application server by using `<cfdirectory />`.
- Work with files in RAM.

10. Tuning application performance

- List and describe the features, benefits, and restrictions when using strict components.
- Manage application performance by using query caching. (Query cache approaches include: `<cfquery cachedWithin = "" />`, `<cfquery cachedAfter = "" />`, assigning the query object to a persistent scope.)
- List and describe the features and benefits of using the `<cfcache />` tag.
- List and describe the features and benefits of using the object cache.
- List and describe the use of the `<cfthread />` tag and the associated actions: `join`, `run`, `sleep`, and `terminate`.

11. Building advanced user interfaces

- Use `<cfajaxproxy />` to call a remote service from JavaScript.
- Use built-in UI components. (Components include: `DataGrid`, `TreeControl`, `Layout`, `MessageBox`, `CFDiv`.)
- Use built-in JSON functions to serialize data.

12. Reporting

- Create and manipulate PDF documents by using `<cfdocument />` and `<cfpdf />`.
- Create and manipulate Office documents by using `<cfspreadsheet />`, `<cfdocument />`, and `<cfpresentation />`.
- Create a chart by using `<cfchart />`.

Practice Exam

Try out these practice questions to get a feel for the types of questions on the ACE exam. Please note that your performance here does not indicate how you will do on the actual exam. To fully prepare for the exam, closely review the topic areas and objectives in this Exam Bulletin.

3.4 List and describe the properties of the RESULT structure of a query object

Given the following <CFQUERY>:

```
<cfquery name="qdata" datasource="cf9exam" result="stData">
  insert into person (firstname,lastname) values ('John','Doe')
</cfquery>
```

Which variable contains the generated primary key?

- A. stdata.generatedkey
- B. stdata.ROW_ID
- C. stdata.identitycol
- D. it varies, depending on your database

Correct answer: A

5.1 Create a persistent component.

Which scope can be used to contain an instance of a component that persists between page requests?

- A. request
- B. client
- C. session
- D. variables

Correct answer: C

5.1 Create a persistent component.

You want to utilize the ORM features of ColdFusion 9. Which property must be set in the Application.cfc file?

- A. useORM = true
- B. ormEnabled = true

- C. ormDisabled = false
- D. persistent = true

Correct answer: B

6.2 Parse XML data by using <cfxml /> or xmlParse()

Which ColdFusion function XML document validates XML against a DTD?

- A. xmlParse()
- B. xmlTransform()
- C. xmlNew()
- D. isXMLDoc()

Correct answer: A

7.3 Reuse code by creating user-defined functions. (User-defined functions written with either <cffunction /> or within a <cfscript /> block.

Which is valid syntax for caching a user-defined function in ColdFusion's Application scope?

- A. <cffunction name="application.foo">...</cffunction>
- B. <cffunction name="foo">...</cffunction><cfset application.foo = variables.foo>
- C. <cffunction name="foo" scope="application">...</cffunction>
- D. <cfscript>function application.foo() { ... }</cfscript>

Correct answer: B

9.3 Work with files in RAM

You want to view the amount of memory your application is using to store files in RAM. Which command should you use to accomplish this?

- A. <cfdump var="#getInMemoryFileSize("ram")#">
- B. <cfdump var="#getVFSMetaData("ram")#">
- C. <cfdump var="#getInMemoryFileSize("all")#">
- D. <cfdump var="#getVFSMetaData("all")#">

Correct answer: B