

What's new and changed in ColdFusion (2018 release) Public Beta Update 1

Named parameters

1. In this update, we have renamed the parameters for the following functions. For a list of all functions with change in named parameters, see [Named parameters](#).

- `ArrayFind(array,value)`
- `ArrayFind(array,callback)`
- `ArrayFindAll(array,value)`
- `ArrayFindAll(array,callback)`
- `ArrayFindNoCase(array,value)`
- `CreateObject(type,className)`
- `CreateObject(type,className,context,server,action,domain,username,password)`
- `CreateObject(type,className,context,server)`
- `CreateObject(type,className, PortName,server)`
- `CreateObject(type,className, PortName, Locale)`
- `CreateObject(type,className,context, Locale)`
- `CreateObject(type,className, Assembly,server)`
- `CreateObject(type,className, Assembly, Locale)`
- `CreateObject(type,className,context,server,DotnetPort)`
- `CreateObject(type,className,context,server,action)`
- `CreateObject(type,className,context,locale,action)`
- `CreateObject(type,className,context,locale,DotnetPort)`
- `CreateObject(type,className,PortName,server,DotnetPort)`
- `CreateObject(type,className,PortName,server,action)`
- `CreateObject(type,className,PortName,locale,action)`
- `CreateObject(type,className,PortName,locale,DotnetPort)`
- `CreateObject(type,className,Assembly,server,DotnetPort)`
- `CreateObject(type,className,Assembly,server,action)`
- `CreateObject(type,className,Assembly,locale,action)`
- `CreateObject(type,className,Assembly,locale,DotnetPort)`
- `CreateObject(type,className,context,server,DotnetPort)`
- `CreateObject(type,className,context,server,action)`
- `CreateObject(type,className,context,locale,action)`
- `CreateObject(type,className,context,locale,DotnetPort)`
- `CreateObject(type,className,PortName,server,DotnetPort,protocol,secure)`
- `CreateObject(type,className,PortName,server,action,protocol,secure)`
- `CreateObject(type,className,PortName,locale,action,protocol,secure)`
- `CreateObject(type,className,PortName,locale,DotnetPort,protocol,secure)`

- CreateObject(type,className,Assembly,server,DotnetPort,protocol,secure)
- CreateObject(type,className,Assembly,server,action,protocol,secure)
- CreateObject(type,className,Assembly,locale,action,protocol,secure)
- CreateObject(type,className,Assembly,locale,DotnetPort,protocol,secure)
- DatePart(datePart,date)
- DayOfWeekAsString(dayOfWeek,[locale=defaultLocale])
- Decrypt(string,key,[algorithm=CFMX_COMPAT,encoding=UU,IV="",iterations=0])
- Decrypt(string,key,[algorithm=CFMX_COMPAT,encoding=UU,Salt="",iterations=0])
- Decrypt(string,key,[algorithm=CFMX_COMPAT])
- EncryptBinary(binaryData,key,[algorithm=CFMX_COMPAT,IV="",iterations=0])
- EncryptBinary(binaryData,key,[algorithm=CFMX_COMPAT,Salt="",iterations=0])
- FileRead(filePath,[charset=""])
- FileRead(filePath,[bufferSize=""])
- FileSeek(file,position)
- GetFileFromPath(filePath)
- GetTempFile(path,prefix)
- GetVFSMetadata(fileSystemType)
- Hash(string,[algorithm=CFMX_COMPAT,encoding="",additionalIterations=1])
- invoke(object,methodName,arguments)
- IsNull(value)
- IsXml(value, ParserOptions)
- ListFilter(list,callback,[delimiter=„,includeEmptyFields=false])
- ListSort(list,callback,[Delimiter=""])
- ObjectLoad(path)
- QueryAddRow(query,data)
- QueryAddRow(query,number)
- QueryExecute(sql,[params="",queryOptions=""])
- QueryGetRow(query,rowNumber)
- Replace(string,substring,replacement,[scope=ONE])
- Replace(string,substring,callback,[scope=ONE])
- ReplaceList(string,list1,list2,[delimiterList=""])
- ReplaceNoCase(string,substring,callback,[scope=ONE])
- SpreadsheetAddImage(spreadsheet,imagePath,anchor)
- SpreadsheetCreateSheet(spreadsheet,[sheetName=""])
- SpreadsheetGetColumnCount(spreadsheet,[sheetName=""])
- SpreadsheetGetColumnCount(spreadsheet,[sheetNumber=""])
- SpreadsheetRead(filePath,[sheetName=all])
- SpreadsheetRead(filePath,[sheetNumber=all])
- StructFindValue(struct,key,[scope=ONE])
- StructToSorted(struct,[sortType=text,sortOrder=asc,path="",localeSensitive=false])
- ToScript(CFVariable,JSVariableName,[outputFormat=true,ASFormat=false])
- Week(date,[calendar=""])
- XMLFormat(string,[escape=false])

- XmlParse(xmlString,[caseSensitive=false,validator=""])
 - XMLValidate(xmlString)
2. In this update, we have removed parameter names that had "or". For example, we changed ArrayFind(array, valueOrCallback) to:
- a. ArrayFind(array,value)
 - b. ArrayFind(array,callback)

ApplicationGetMetadata

Bug fixes for the function ApplicationGetMetadata.

Evaluate function

The Evaluate function accepts an array of expressions. This is a fix for the bug [CF-4201978](#).

For example,

```
<cfscript>
    finalvar = evaluate(expression=
    [
        "setvariable( 'first',1 )",
        "setvariable( 'second', 2 )",
        "setvariable( 'third', 3 )",
        "setvariable( 'fourth', 4 )"
    ]
    )
    result= evaluate(expression='1 + 1') // evaluate with single expression
</cfscript>
```

Issue fixed

In the last Public Beta, there were issues when you used certain keywords as variable names without semi-colons. In this release, we have fixed that. You can use the following as variables without semi-colons:

- Abort
- Lock
- Include
- Thread
- Param
- Transaction
- Throw
- Flush
- Setting
- Logout
- Exit
- Rethrow

For example, the snippet below now runs as expected:

```
<cfscript>
    try {
        throw="sample throw message"
        throw(message=#throw#, detail="xyz");
    }
    catch (any e) {
        WriteOutput("Error: " & e.message)
        Rethrow
    }
</cfscript>
```