Utility Functions: ListIsValid() and ListRemoveInvalid()

Posted At : February 27, 2009 9:28 AM | Posted By : Jon Hartmann
Related Categories: ColdFusion, Utility Function

One technique I often use is to create gateway objects to my database that allow the user to pass in multiple IDs, rather then a single one. I've run into a problem though; what if the user passes through something that is a valid list, but is not a valid list of the type I need? Read more to see ListIsValid() abd ListRemoveInvalid().

OK, so the basic premises is that I have a list, but all the items in that list need to be of a specific type, say email addresses, or numeric values. First, I need a way to test a whole list to see if its composed of that type or not:

```
<cffunction name="ListIsValid" output="false" returntype="boolean">
    <cfargument name="list" type="string" required="true" />
    <cfargument name="type" type="string" required="true" />
    <cfargument name="delimiters" type="string" required="false" default="," />

    <cfset var local = StructNew() />

    <cfset local.returnValue = true />

    <cfif ListLen(arguments.list, arguments.delimiters) gt 0>
        <cfset local.array = ListToArray(arguments.list, arguments.delimiters) />
        <cfset local.length = ArrayLen(local.array) />

        <cfloop from="1" to="#local.length#" index="x">
            <cfset local.returnValue = (local.returnValue AND IsValid(arguments.type, local.array[x])) />
        </cfloop>
    </cfif>

    <cfreturn local.returnValue />
</cffunction>
```

That will let me use any of the valid types from IsValid() to test me list, except the regex and range ones.

Now what do you do if you need to clean your list of those unwanted values, say to get the count of valid values in the list:

```
<cffunction name="ListRemoveInvalid" output="false" returntype="string">
    <cfargument name="list" type="string" required="true" />
    <cfargument name="type" type="string" required="true" />
    <cfargument name="delimiters" type="string" required="false" default="," />

    <cfset var local = StructNew() />
    <cfset local.newArray = ArrayNew(1) />

    <cfif ListLen(arguments.list, arguments.delimiters) gt 0>
        <cfset local.array = ListToArray(arguments.list, arguments.delimiters) />
        <cfset local.length = ArrayLen(local.array) />

        <cfloop from="1" to="#local.length#" index="x">
            <cfif IsValid(arguments.type, local.array[x])>
                <cfset ArrayAppend(local.newArray, local.array[x]) />
            </cfif>
        </cfloop>
    </cfif>
```

```
    <cfreturn ArrayToList(local.newArray, arguments.delimiters) />
</cffunction>
```

Examples

Lists

List1: 1,2,3
List2: 1,2,test@jonhartmann.com
List3: jon@jonhartmann.com,me@jonhartmann.com,test@jonhartmann.com

Numeric

ListIsValid(List1, "numeric"): YES
ListIsValid(List2, "numeric"): NO
ListIsValid(List3, "numeric"): NO

Email

ListIsValid(List1, "email"): NO
ListIsValid(List2, "email"): NO
ListIsValid(List3, "email"): YES

SSN

ListIsValid(List1, "ssn"): NO
ListIsValid(List2, "ssn"): NO
ListIsValid(List3, "ssn"): NO

Remove Invalid Email

ListRemoveInvalid(List1, "email"):
ListRemoveInvalid(List2, "email"): test@jonhartmann.com
ListRemoveInvalid(List3, "email"): jon@jonhartmann.com,me@jonhartmann.com,test@jonhartmann.com