

## **ColdFusion Yahoo Package**

Welcome to the ColdFusion Yahoo Package. This is a set of CFCs built to integrate with Yahoo's various web-based services. As new services are added, the package will be updated. Be sure to read the readme.txt file included in the zip for information about recent changes.

The ColdFusion Yahoo Package was created by Raymond Camden. You may email me at [ray@camdenfamily.com](mailto:ray@camdenfamily.com) or visit my blog at <http://ray.camdenfamily.com>.

If you find this package useful, please consider purchasing something from my Amazon wishlist: <http://www.amazon.com/o/registry/2TCL1D08EZEYE>. I also ask that you consider linking to the project URL: <http://cfyahoo.riaforge.org>. This is also the best place to visit for updates.

### ***License***

Copyright 2006 Raymond Camden

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## ***Installation and Setup***

Installation is rather simple. Just take the org file from the zip and copy it to your web server. Where you copy the files depends on how you will use them. If you are only using the files for one application, you may simply put them in a subfolder under where you place to call them. If you want multiple applications to use the package, then consider putting the folder in your web root and creating a ColdFusion mapping to point to the folder.

Secondly - all users of Yahoo's services need an application ID. The code from my package will not work without it. You can get an application ID from free from Yahoo here:

[http://api.search.yahoo.com/webservices/register\\_application](http://api.search.yahoo.com/webservices/register_application)

Once you have your application ID, you can use it in one of two ways.

- 1) Edit org.camden.yahoo.base.cfc. Find this line of code: `<cfset variables.appid = "">`. Set your application ID here and all the code on your server will use that ID.
- 2) All service CFCs extend Base.cfc. Base.cfc has a setAppID function. When creating an instance of the service CFC, simply call the setAppID() function and pass in your application ID. This would be useful in a situation where more than one application uses the Yahoo Services and you do not want to run into the limitations of having just one application ID.

## Answers API

The answers API may be used with answers.cfc. The test script is answer\_test.cfm. The Answers API lets you search Yahoo's Answers service for questions. The Answer CFC has one method, **questionSearch**, with the following arguments:

<i>Name</i>	<i>Purpose</i>
query	The search query.
search_in	Tells Yahoo where to search for the query. Values may be: all (default), question, best_answer, nickname
category_id	Either one, or a list of, category ID values to restrict your results.
category_name	Like category_id, this lets you restrict your results.
region	A geographical region to restrict your results by. Supported regions are: us (default),uk,ca,au,in,es,br,ar,mx,e1,it,de,fr,sg.
date_range	Filter based on date submitted. Values may be: all (default),7,7-30,30-60,60-90,more90. Values are in days.
sort	Sorts results. Supported values are: relevance (default), date_desc, and date-asc.
type	Filters based on question status. Valid values are: all (default), resolved, open, undecided
start	The starting position of the results. This allows you to page through sets of results. This defaults to 0.
results	The number of results to return. This defaults to 10 and the max is 50.

The method returns a query of results. The query contains the following columns.

<i>Name</i>	<i>Purpose</i>
category	The category of the question.
categoryid	The ID of the category.
chosenanswer	If the question has a chosen answer (picked by the author of the question), then this column will contain the value.
chosenanswerawarddate	If the question has a chosen answer, this is the date value of when the answer was selected.
chosenanswerdate	If the question has a chosen answer, this is the date when the answer was submitted.
chosenanswerid	The ID value of the chosen answer.
chosenanswernick	The nickname of the user who submitted the chosen answer.
date	The date when the question was created.
id	The ID of the question.

<i><b>Name</b></i>	<i><b>Purpose</b></i>
link	A link to the question.
numanswer	The number of answers to the question.
numcomments	The number of comments to the question.
question	The actual question.
subject	The subject of the question.
type	The status of the question.
userid	The user id of the question writer.
username	The nick name of the user who wrote the question.
userphotourl	An optional link to the question writer's online photo.