**WORK SAMPLES**

function loadBillingMonthlyData( month, year, monthSlot ) {

document.getElementById("detail\_" + monthSlot).style.display = '';

for( i=0; i<arBillingMonths.length; i++ )

{

if( arBillingMonths[i][1] == parseInt(monthSlot) )

{

//alert("setting bMonthsLoaded["+i+"] = false");

bMonthsLoaded[i] = false;

break;

}

}

var url = "<cfoutput>#variables.ajaxBaseComURL#</cfoutput>/BillingDetail.cfc"; // ajaxBaseComURL is defined in loadappsettings.cfm

var ajaxCall = new mxAjax.Data({

executeOnLoad : true, // URL URL URL URL URL URL URL

paramArgs : new mxAjax.Param(url,{cffunction:"loadBillingMonthly", param:"month=" + month + ",year=" + year + ",numberOfArrows=" + monthSlot}),

postFunction : handleData

});

var ajaxCall2 = new mxAjax.Data({

executeOnLoad : true,

paramArgs : new mxAjax.Param(url,{cffunction:"inquireBillImage", param:"month=" + month + ",year=" + year + ",numberOfArrows=" + monthSlot}),

postFunction : handleDataForBillImage

});

}

// this is for going to a specific month

function goToMonth()

{

var month = $("selectMonth").value;

var year = $("selectYear").value

var date = new Date( month + "/1/" + year );

var newDate = new Array();

var loadSlot = false;

if( areMonthsLoaded() )

{

for( i=0; i < arBillingMonths.length; i++ )

{

newDate[i] = new Date();

if( i == 0)

{

dateCopy( date, newDate[i]);

loadSlot = true;

}

else

{

date.setMonth( date.getMonth() + 1 );

dateCopy( date, newDate[i]);

loadSlot = false;

}

arBillingMonths[i][0] = newDate[i];

writeSlot( arBillingMonths[i][1], loadSlot );

}

}

}

<script type="text/javascript">

//var startDate = "2009-01-01";

var loaded = false;

window.MonYr = false;

var $j = jQuery.noConflict();

$j(document).ready(function(){

$j("#GoMonthAndYear").click(

function() {

window.MonYr = true;

//$j( ".snake").show();

pleaseWaitModal(true,200,400, '', "Loading...<br/>",false);

// pleaseWaitModal(true,200,400, '', "Loading...<br/>foo",false);

<cfoutput>getEventHistory(#jsonAcctData#);</cfoutput>

}

);

$j("#GoDateRange").click(

function() {

window.MonYr = false;

//$j( ".snake").show();

pleaseWaitModal(true,200,400, '', "Loading...<br/>",false);

<cfoutput>getEventHistory(#jsonAcctData#);</cfoutput>

}

);

$j("BucketTransactionsGoButton").click(

function() {

//$j( ".snake").show();

pleaseWaitModal(true,200,400, '', "Loading...<br/>",false);

<cfoutput>getBucketTransactions(#jsonBucketTransactionRequestData#);</cfoutput>

}

);

$j("#BucketBalancesGoButton").click(

function() {

//$j( ".snake").show();

pleaseWaitModal(true,200,400, '', "Loading...<br/>",false);

<cfoutput>getBucketBalances(#jsonBucketTransactionRequestData#);</cfoutput>

}

);

});

var getEventHistory = function(data){

//var dateRange = searchByMonthAndyear();

//dump(searchByMonthAndyear());

//dump(lastDay(searchByMonthAndyear()));

pleaseWaitModal(true,200,400, '', "Loading...<br/>",false);

if(window.MonYr == true){

var fromDate = searchByMonthAndyear();

//alert(fromDate);

var toDate = lastDay1(searchByMonthAndyear());

} else {

var fromDate = $("fromEffectiveDate").value;

var toDate = $("toEffectiveDate").value;

}

//dump(fromDate+"\n");

//dump(toDate+"\n");

if(toDate < fromDate){

alert("The from date " +fromDate+" cannot be after the to date "+ toDate);

$j(".snake").hide();

return false;

}

data["START"] = fromDate;

data["END"] = toDate;

//dump(data["END"]);

// create an instance of the proxy.

var oBO = new EventHistory();

// Setting a callback handler for the proxy automatically makes

// the proxy's calls asynchronous.

oBO.setCallbackHandler(handleEventHistory);

oBO.setErrorHandler(myErrorHandler);

// The proxy getEventHistory function represents the CFC

// AjaxInquireBillingOrders function.

oBO.AjaxInquireBillingOrders(data);

}</script>

function addressPop()

{

var myUrl;

myURL = "/js/exampleAddresses.cfm"

TheNewWin = window.open(myURL,'addressesPop','height=300,width=466,toolbar=0,location=0,directories=0,status=0,menubar=0,scrollbars=1,resizable=0');

}

function dspToggle(objId)

{

var el = document.getElementById(objId);

// if it's not currently being obscured, obscure it. otherwise, show it.

if (el.style.display != 'none')

{

dspToggleHide(objId);

}

else

{

dspToggleShow(objId);

}

}

function dspToggleShow(objId)

{

var el = document.getElementById(objId);

el.style.display = '';

}

function dspToggleHide(objId)

{

var el = document.getElementById(objId);

el.style.display = 'none';

}

function diagShowSwitches()

{

var losNo = $('losToValidate').value;

var dspText = "esnValidated: " + eval("document.pInfo.esnValidated"+losNo).value + "\n"

+ "esnOnActiveAccount: " + eval("document.pInfo.esnOnActiveAccount"+losNo).value + "\n"

+ "esnHasHistory: " + eval("document.pInfo.esnHasHistory"+losNo).value + "\n"

+ "isCricketPhone: " + eval("document.pInfo.isCricketPhone"+losNo).value + "\n"

+ "decisionOptionsLoaded: " + document.pInfo.decisionOptionsLoaded.value + "\n"

+ "bypassIdValidation: " + document.pInfo.bypassIdValidation.value + "\n"

+ "bypassEsnValidation: " + document.pInfo.bypassEsnValidation.value;

alert(dspText);

}

function Left(str, n)

{

if (n <= 0)

return "";

else if (n > String(str).length)

return str;

else

return String(str).substring(0,n);

}

function getBillingCityState()

{

var status;

var validZip = true;

if(isNaN(document.pInfo.zip5.value)) {

validZip = false;

} else if (document.pInfo.orgZip.value != document.pInfo.zip5.value) {

validZip = false;

status = getData("#MYSELF#m\_account.getCityState&zip=" + document.pInfo.zip5.value);

if (status == 1 && postresult.length > 2)

{

validZip = true;

document.pInfo.orgZip.value = document.pInfo.zip5.value;

var arrList = postresult.split(",");

document.pInfo.city.value = trim(arrList[0]);

for (var i=0; i< document.pInfo.state.options.length; i++)

{

if (document.pInfo.state.options[i].value == Left(arrList[1],2))

document.pInfo.state[i].selected = true;

}

}

}

if(!validZip) {

alert("Please enter a valid zip code.");

document.pInfo.orgZip.value = "";

document.pInfo.city.value = "";

if (document.pInfo.state.options.length > 0) document.pInfo.state[0].selected = true;

document.pInfo.zip5.focus();

document.pInfo.zip5.select();

}

return;

}

function showEMail()

{

var losNo = $('losToValidate').value;

if (document.getElementById("phoneModel" + losNo)) {

if (document.getElementById('phoneModel'+losNo).value.length > 0) {

var JSDataCardList = '<cfoutput>#application.EVERGREEN\_DATACARDS#</cfoutput>';

var useDataCard = false;

var arrDataCards = JSDataCardList.split(",");

//check to see if user has a datacard

for(var i=0;i<arrDataCards.length;i++){

if (document.getElementById('phoneModel'+losNo).value == trim(arrDataCards[i])){

var useDataCard = true;

break;

}

}

if( document.getElementById("emailspan") && document.getElementById("optinspan") ) {

if (useDataCard == true) {

document.getElementById("emailspan").style.display="";

document.getElementById("optinspan").style.display="";

}

else {

document.getElementById("emailspan").style.display="none";

document.getElementById("optinspan").style.display="none";

}

}

}

}

}