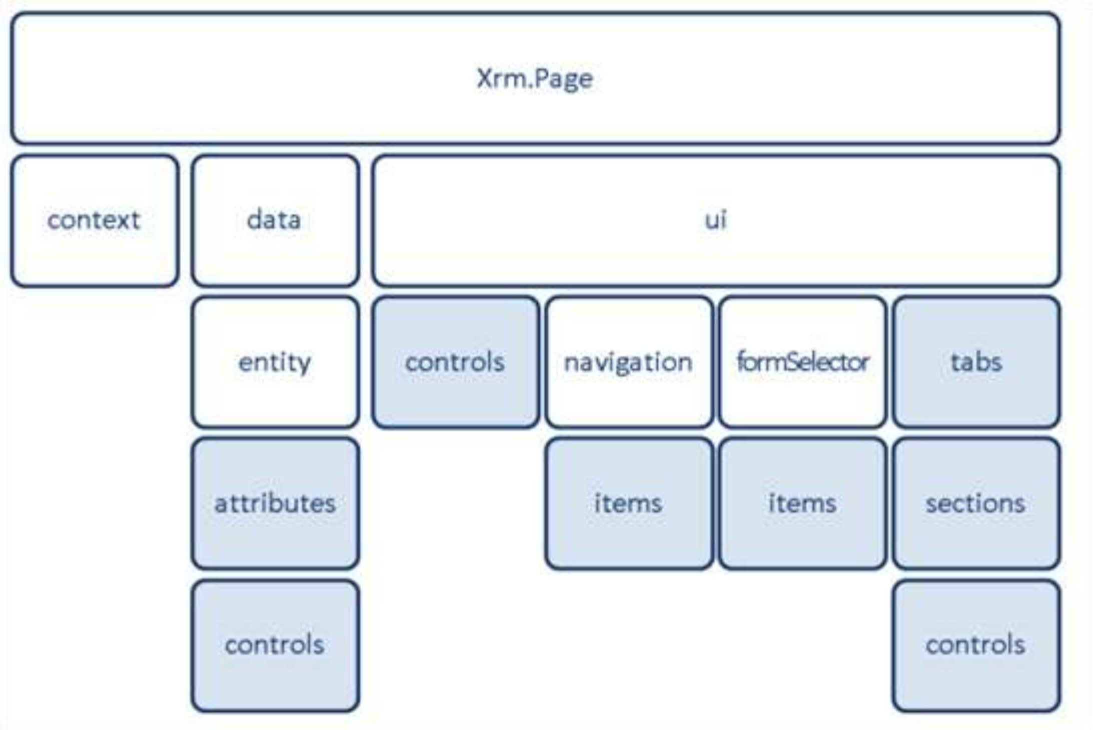
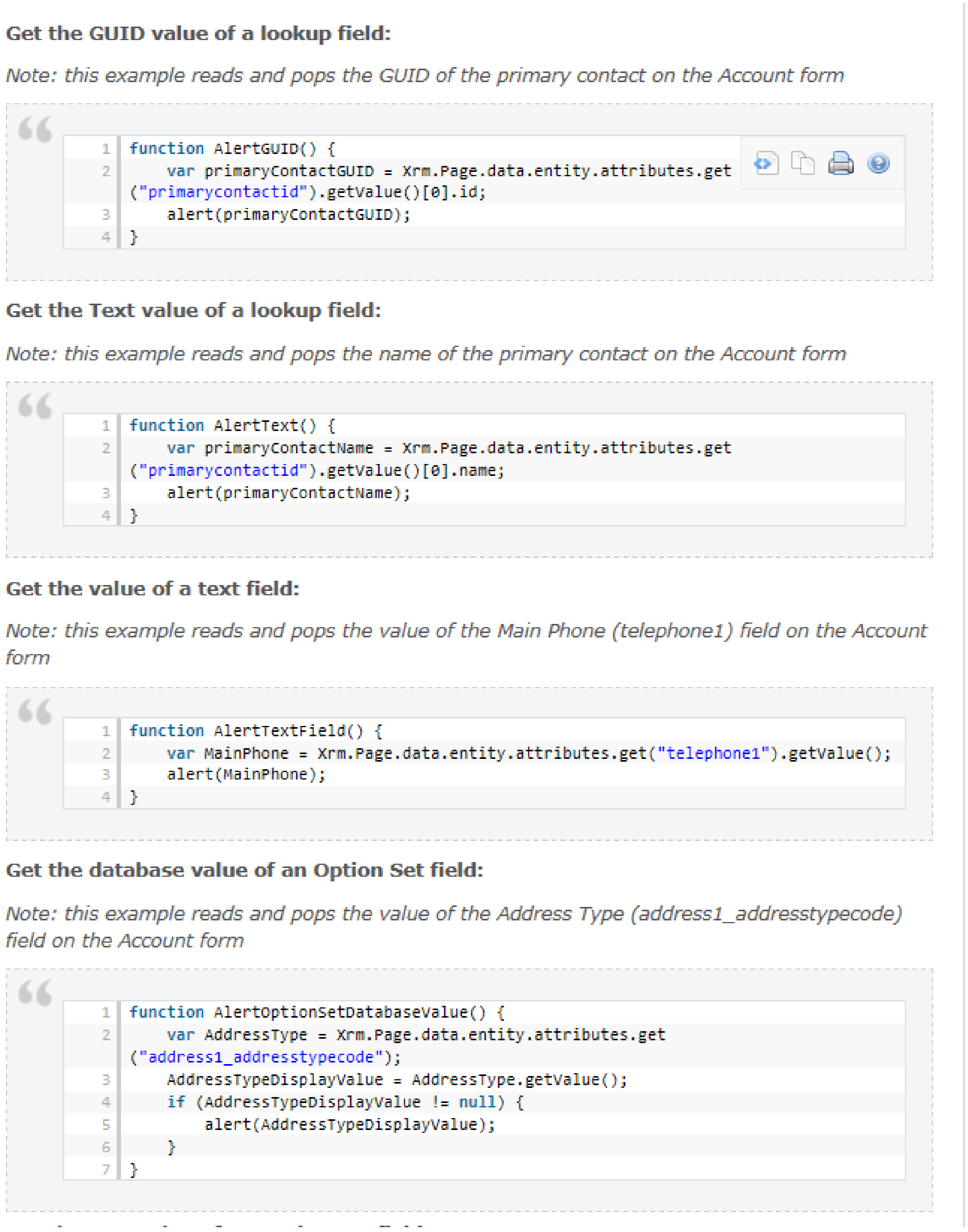

Xrm.Page
context
data
ui
entity
controls
navigation
formSelector
tabs
attributes
items
items
sections
controls
controls



Get the GUID value of a lookup field:

Note: this example reads and pops the GUID of the primary contact on the Account form

```
function AlertGUID() {
    var primaryContactGUID = Xrm.Page.data.entity.attributes.get("primarycontactid").getValue()[0].id;
    alert(primaryContactGUID);
}
```

Get the Text value of a lookup field:

Note: this example reads and pops the name of the primary contact on the Account form

```
function AlertText() {
    var primaryContactName = Xrm.Page.data.entity.attributes.get("primarycontactid").getValue()[0].name;
    alert(primaryContactName);
}
```

Get the value of a text field:

Note: this example reads and pops the value of the Main Phone (telephone1) field on the Account form

```
function AlertTextField() {
    var MainPhone = Xrm.Page.data.entity.attributes.get("telephone1").getValue();
    alert(MainPhone);
}
```

Get the database value of an Option Set field:

Note: this example reads and pops the value of the Address Type (address1_addresstypecode) field on the Account form

```
function AlertOptionSetDatabaseValue() {
    var AddressType = Xrm.Page.data.entity.attributes.get("address1_addresstypecode");
    AddressTypeDisplayValue = AddressType.getValue();
    if (AddressTypeDisplayValue != null) {
        alert(AddressTypeDisplayValue);
    }
}
```

Get the database value of a Bit field:

```
// example GetBitValue("telephone1");
function GetBitValue(fieldname) {
    return Xrm.Page.data.entity.attributes.get(fieldname).getValue();
}
```

Get the value of a Date field:

returns a value like: Wed Nov 30 17:04:06 UTC+0800 2011

and reflects the users time zone set under personal options

```
// example GetDate("createdon");
function GetDate(fieldname) {
    return Xrm.Page.data.entity.attributes.get(fieldname).getValue();
}
```

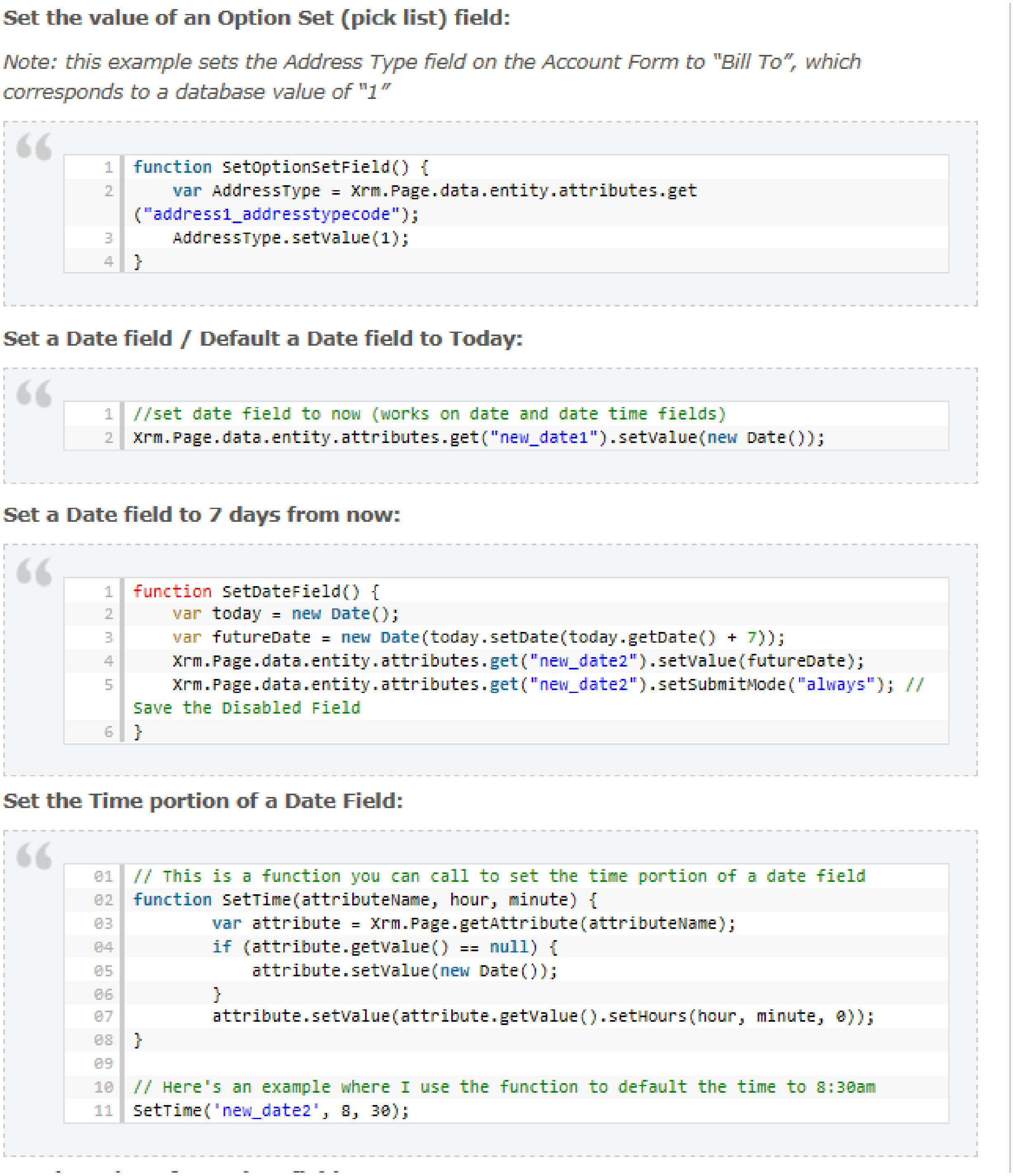
Get the day, month and year parts from a Date field:

```
// This function takes the fieldname of a date field as input and returns a DD-MM-YYYY value
// Note: the day, month and year variables are numbers
function FormatDate(fieldname) {
    var d = Xrm.Page.data.entity.attributes.get(fieldname).getValue();
    if (d != null) {
        var curr_date = d.getDate();
        var curr_month = d.getMonth();
        curr_month++;  // getMonth() considers Jan month 0, need to add 1
        var curr_year = d.getFullYear();
        return curr_date + "-" + curr_month + "-" + curr_year;
    }
    else return null;
}

// An example where the above function is called
alert(FormatDate("new_date2"));
```

Set the value of a string field:

Note: this example sets the Account Name field on the Account Form to "ABC"

```
function SetStringField() {
    var Name = Xrm.Page.data.entity.attributes.get("name");
    Name.setValue("ABC");
}
```



Set the value of an Option Set (pick list) field:

Note: this example sets the Address Type field on the Account Form to "Bill To", which corresponds to a database value of "1"

```
function SetOptionSetField() {
    var AddressType = Xrm.Page.data.entity.attributes.get("address1_addresstypecode");
    AddressType.setValue(1);
}
```

Set a Date field / Default a Date field to Today:

```
//set date field to now (works on date and date time fields)
Xrm.Page.data.entity.attributes.get("new_date1").setValue(new Date());
```

Set a Date field to 7 days from now:

```
function SetDateField() {
    var today = new Date();
    var futureDate = new Date(today.setDate(today.getDate() + 7));
    Xrm.Page.data.entity.attributes.get("new_date2").setValue(futureDate);
    Xrm.Page.data.entity.attributes.get("new_date2").setSubmitMode("always"); // Save the Disabled Field
}
```

Set the Time portion of a Date Field:

```
// This is a function you can call to set the time portion of a date field
function SetTime(attributeName, hour, minute) {
        var attribute = Xrm.Page.getAttribute(attributeName);
        if (attribute.getValue() == null) {
            attribute.setValue(new Date());
        }
        attribute.setValue(attribute.getValue().setHours(hour, minute, 0));
}

// Here's an example where I use the function to default the time to 8:30am
SetTime('new_date2', 8, 30);
```

Set the value of a Lookup field:

Note: here I am providing a reusable function…

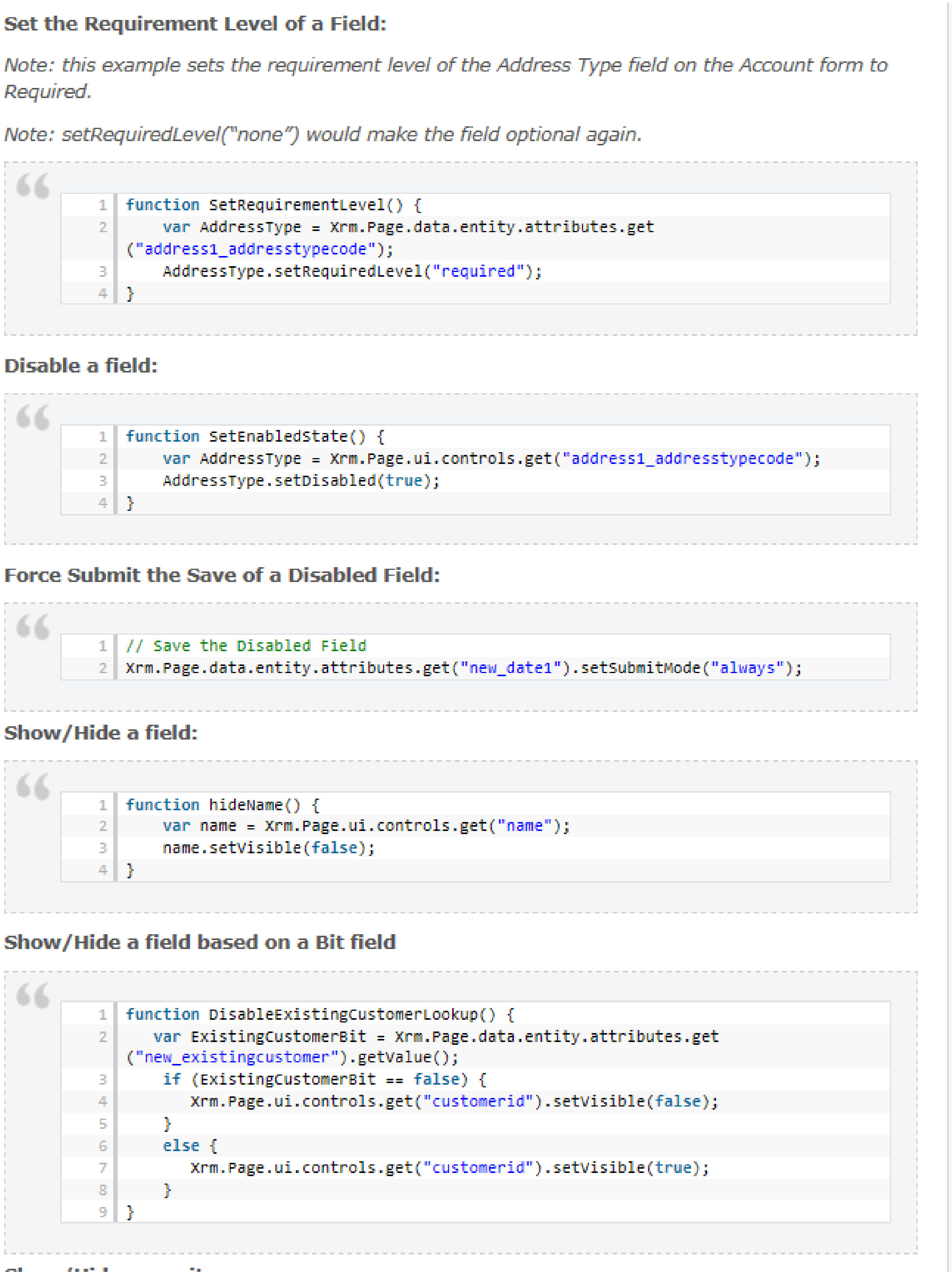
```
// Set the value of a lookup field
function SetLookupValue(fieldName, id, name, entityType) {
    if (fieldName != null) {
        var lookupValue = new Array();
        lookupValue[0] = new Object();
        lookupValue[0].id = id;
        lookupValue[0].name = name;
        lookupValue[0].entityType = entityType;
        Xrm.Page.getAttribute(fieldName).setValue(lookupValue);
    }
}
```

Here's an example of how to call the function (I retrieve the details of one lookup field and then call the above function to populate another lookup field):

```
var ExistingCase = Xrm.Page.data.entity.attributes.get("new_existingcase");
if (ExistingCase.getValue() != null) {
    var ExistingCaseGUID = ExistingCase.getValue()[0].id;
    var ExistingCaseName = ExistingCase.getValue()[0].name;
    SetLookupValue("regardingobjectid", ExistingCaseGUID, ExistingCaseName, "incident");
}
```

Split a Full Name into First Name and Last Name fields:

```
function PopulateNameFields() {
    var ContactName = Xrm.Page.data.entity.attributes.get("customerid").getValue()[0].name;
    var mySplitResult = ContactName.split(" ");
    var fName = mySplitResult[0];
    var lName = mySplitResult[1];
    Xrm.Page.data.entity.attributes.get("firstname").setValue(fName);
    Xrm.Page.data.entity.attributes.get("lastname").setValue(lName);
}
```



Set the Requirement Level of a Field:

Note: this example sets the requirement level of the Address Type field on the Account form to Required.

Note: setRequiredLevel("none") would make the field optional again.

```
function SetRequirementLevel() {
    var AddressType = Xrm.Page.data.entity.attributes.get("address1_addresstypecode");
    AddressType.setRequiredLevel("required");
}
```

Disable a field:

```
function SetEnabledState() {
    var AddressType = Xrm.Page.ui.controls.get("address1_addresstypecode");
    AddressType.setDisabled(true);
}
```

Force Submit the Save of a Disabled Field:

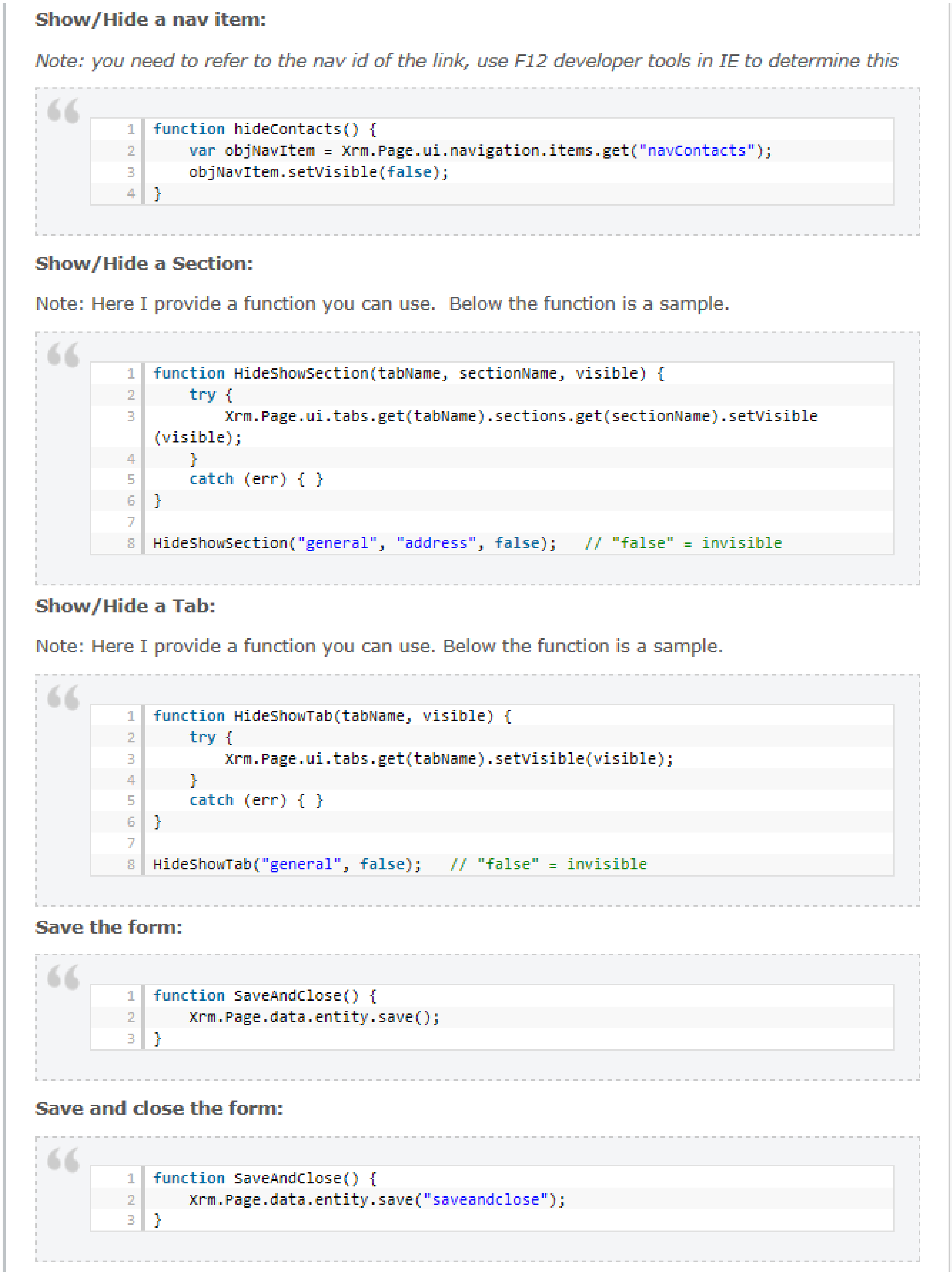
```
// Save the Disabled Field
Xrm.Page.data.entity.attributes.get("new_date1").setSubmitMode("always");
```

Show/Hide a field:

```
function hideName() {
    var name = Xrm.Page.ui.controls.get("name");
    name.setVisible(false);
}
```

Show/Hide a field based on a Bit field

```
function DisableExistingCustomerLookup() {
   var ExistingCustomerBit = Xrm.Page.data.entity.attributes.get("new_existingcustomer").getValue();
    if (ExistingCustomerBit == false) {
       Xrm.Page.ui.controls.get("customerid").setVisible(false);
    }
    else {
       Xrm.Page.ui.controls.get("customerid").setVisible(true);
    }
}
```



Show/Hide a nav item:

Note: you need to refer to the nav id of the link, use F12 developer tools in IE to determine this

```
function hideContacts() {
    var objNavItem = Xrm.Page.ui.navigation.items.get("navContacts");
    objNavItem.setVisible(false);
}
```

Show/Hide a Section:

Note: Here I provide a function you can use. Below the function is a sample.

```
function HideShowSection(tabName, sectionName, visible) {
    try {
        Xrm.Page.ui.tabs.get(tabName).sections.get(sectionName).setVisible(visible);
    }
    catch (err) { }
}

HideShowSection("general", "address", false);   // "false" = invisible
```

Show/Hide a Tab:

Note: Here I provide a function you can use. Below the function is a sample.

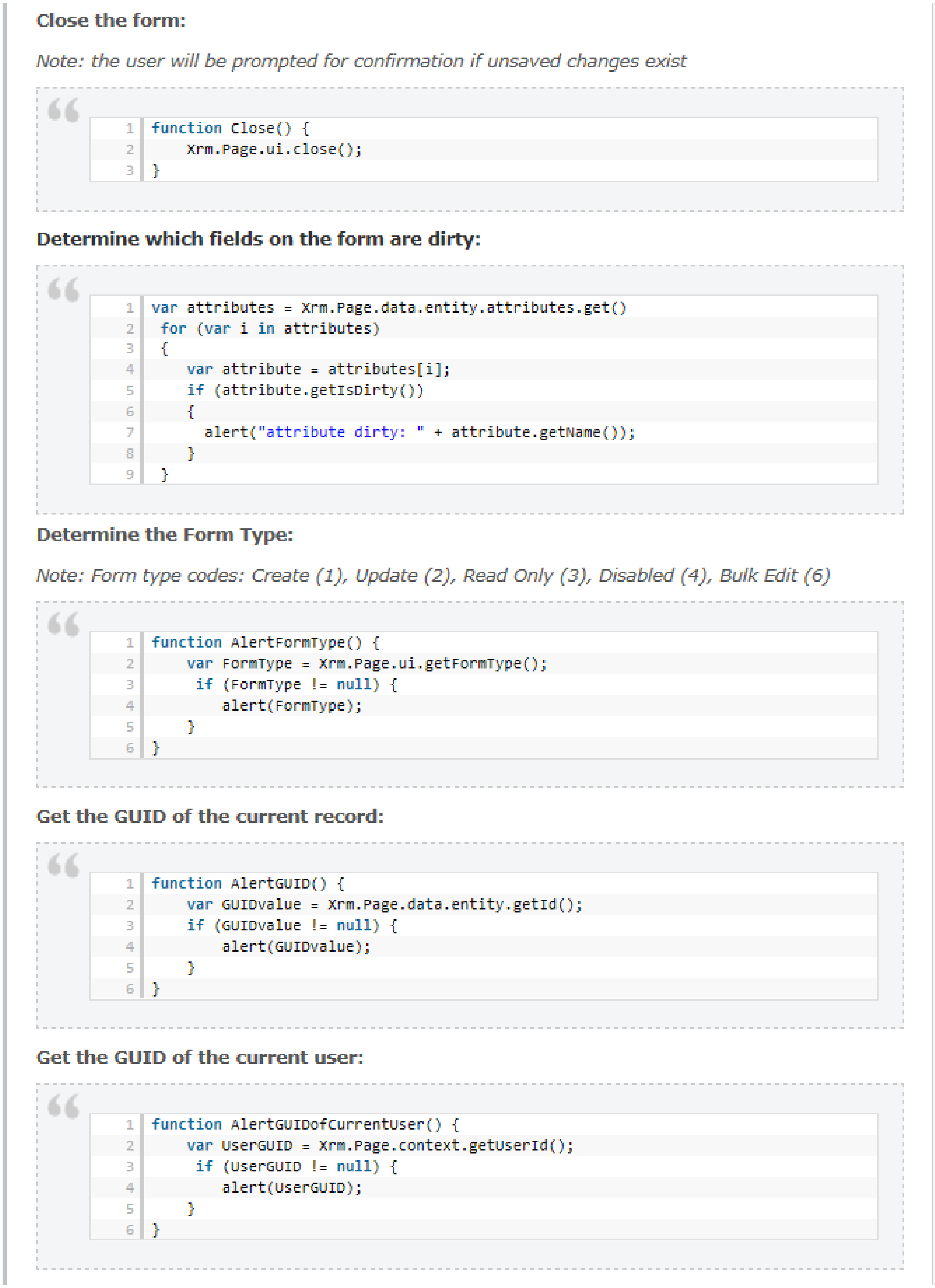
```
function HideShowTab(tabName, visible) {
    try {
        Xrm.Page.ui.tabs.get(tabName).setVisible(visible);
    }
    catch (err) { }
}

HideShowTab("general", false);   // "false" = invisible
```

Save the form:

```
function SaveAndClose() {
    Xrm.Page.data.entity.save();
}
```

Save and close the form:

```
function SaveAndClose() {
    Xrm.Page.data.entity.save("saveandclose");
}
```



Close the form:

Note: the user will be prompted for confirmation if unsaved changes exist

```
function Close() {
    Xrm.Page.ui.close();
}
```

Determine which fields on the form are dirty:

```
var attributes = Xrm.Page.data.entity.attributes.get()
 for (var i in attributes)
 {
    var attribute = attributes[i];
    if (attribute.getIsDirty())
    {
      alert("attribute dirty: " + attribute.getName());
    }
 }
```

Determine the Form Type:

Note: Form type codes: Create (1), Update (2), Read Only (3), Disabled (4), Bulk Edit (6)

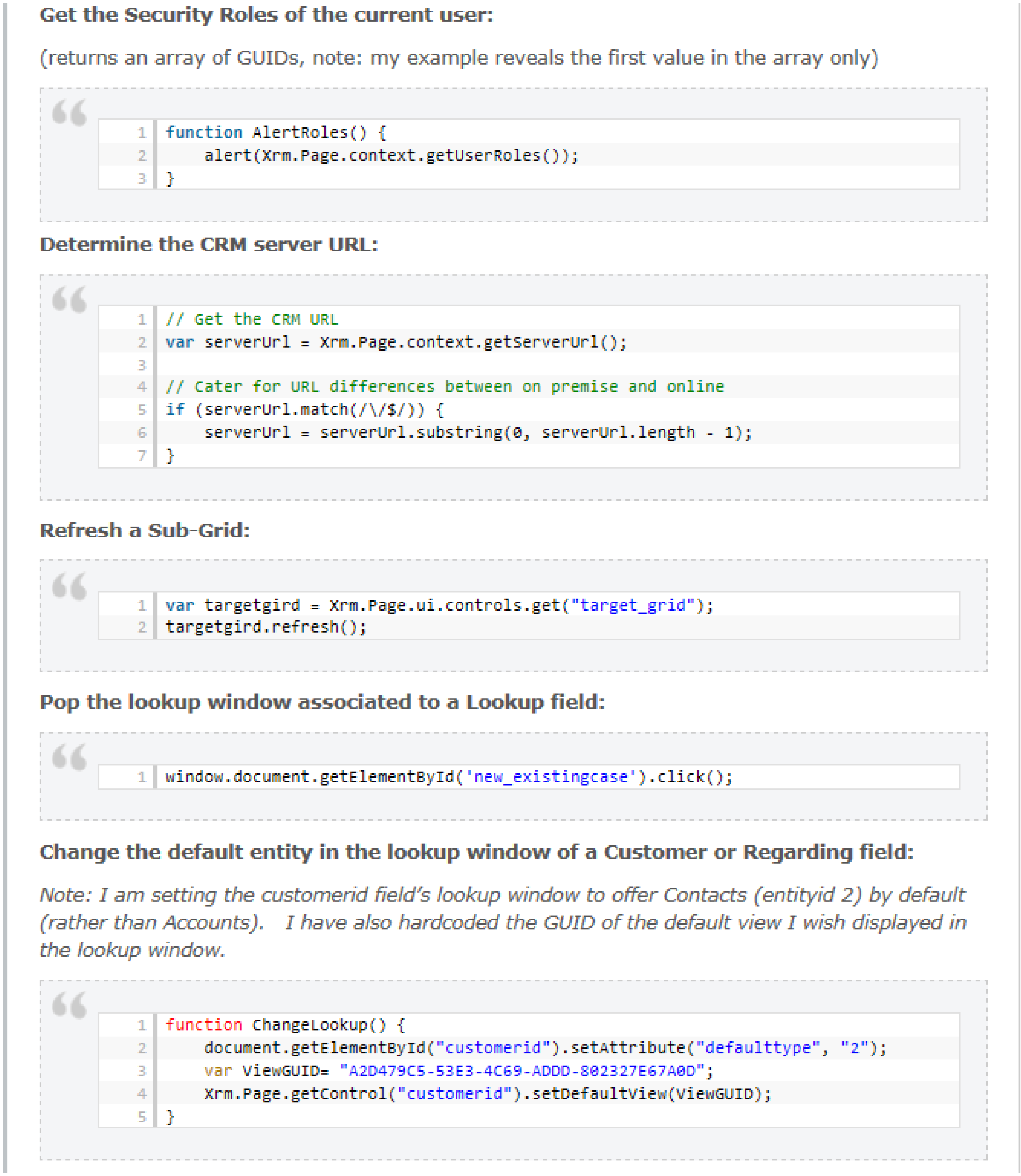
```
function AlertFormType() {
    var FormType = Xrm.Page.ui.getFormType();
     if (FormType != null) {
        alert(FormType);
    }
}
```

Get the GUID of the current record:

```
function AlertGUID() {
    var GUIDvalue = Xrm.Page.data.entity.getId();
    if (GUIDvalue != null) {
        alert(GUIDvalue);
    }
}
```

Get the GUID of the current user:

```
function AlertGUIDofCurrentUser() {
    var UserGUID = Xrm.Page.context.getUserId();
     if (UserGUID != null) {
        alert(UserGUID);
    }
}
```



Get the Security Roles of the current user:

(returns an array of GUIDs, note: my example reveals the first value in the array only)

```
function AlertRoles() {
    alert(Xrm.Page.context.getUserRoles());
}
```

Determine the CRM server URL:

```
// Get the CRM URL
var serverUrl = Xrm.Page.context.getServerUrl();

// Cater for URL differences between on premise and online
if (serverUrl.match(/\/$/)) {
    serverUrl = serverUrl.substring(0, serverUrl.length - 1);
}
```

Refresh a Sub-Grid:

```
var targetgird = Xrm.Page.ui.controls.get("target_grid");
targetgird.refresh();
```

Pop the lookup window associated to a Lookup field:

```
window.document.getElementById('new_existingcase').click();
```

Change the default entity in the lookup window of a Customer or Regarding field:

Note: I am setting the customerid field's lookup window to offer Contacts (entityid 2) by default (rather than Accounts). I have also hardcoded the GUID of the default view I wish displayed in the lookup window.

```
function ChangeLookup() {
    document.getElementById("customerid").setAttribute("defaulttype", "2");
    var ViewGUID= "A2D479C5-53E3-4C69-ADDD-802327E67A0D";
    Xrm.Page.getControl("customerid").setDefaultView(ViewGUID);
}
```

Pop an existing CRM record:

Note: this example pops an existing Case record. The GUID of the record has already been established and is stored in the variable IncidentId.

```
//Set features for how the window will appear
var features = "location=no,menubar=no,status=no,toolbar=no";

// Get the CRM URL
var serverUrl = Xrm.Page.context.getServerUrl();

// Cater for URL differences between on premise and online
if (serverUrl.match(/\/$/)) {
    serverUrl = serverUrl.substring(0, serverUrl.length - 1);
}

window.open(serverUrl + "/main.aspx?etn=incident&pagetype=entityrecord&id=" + encodeURIComponent(IncidentId), "_blank", features, false);
```

Pop the Create form of a CRM record type:

Note: this example pops the Case form from the Phone Call form, defaulting the Case's CustomerID based on the Phone Call's SenderID and defaulting the Case Title to "New Case"

```
//Collect values from the existing CRM form that you want to default onto the new record
var CallerGUID = Xrm.Page.data.entity.attributes.get("from").getValue()[0].id;
var CallerName = Xrm.Page.data.entity.attributes.get("from").getValue()[0].name;

//Set the parameter values
var extraqs = "&title=New Case";
extraqs += "&customerid=" + CallerGUID;
extraqs += "&customeridname=" + CallerName;
extraqs += "&customeridtype=contact";

//Set features for how the window will appear
var features = "location=no,menubar=no,status=no,toolbar=no";

// Get the CRM URL
var serverUrl = Xrm.Page.context.getServerUrl();

// Cater for URL differences between on premise and online
if (serverUrl.match(/\/$/)) {
    serverUrl = serverUrl.substring(0, serverUrl.length - 1);
}

//Pop the window
window.open(serverUrl + "/main.aspx?etn=incident&pagetype=entityrecord&extraqs=" + encodeURIComponent(extraqs), "_blank", features, false);
```

Pop a Dialog from a ribbon button

Note: this example has the Dialog GUID and CRM Server URL hardcoded, which you should avoid. A simple function is included which centres the Dialog when launched.

```
function LaunchDialog(sLeadID) {
    var DialogGUID = "128CEEDC-2763-4FA9-AB89-35BBB7D5517D";
    var serverUrl = "https://avanademarchdemo.crm5.dynamics.com/";
    serverUrl = serverUrl + "cs/dialog/rundialog.aspx?DialogId=" + "{" + DialogGUID + "}" + "&EntityName=lead&ObjectId=" + sLeadID;
    PopupCenter(serverUrl, "mywindow", 400, 400);
    window.location.reload(true);
}

function PopupCenter(pageURL, title, w, h) {
    var left = (screen.width / 2) - (w / 2);
    var top = (screen.height / 2) - (h / 2);
    var targetWin = window.showModalDialog(pageURL, title, 'toolbar=no, location=no, directories=no, status=no, menubar=no, scrollbars=no, resizable=no, copyhistory=no, width=' + w + ', height=' + h + ', top=' + top + ', left=' + left);
}
```

Pop a URL from a ribbon button

Great info on the window parameters you can set here: *http://javascript-array.com/scripts/window_open/*

```
function LaunchSite() {
    // read URL from CRM field
    var SiteURL = Xrm.Page.data.entity.attributes.get("new_sharepointurl").getValue();
    // execute function to launch the URL
    LaunchFullScreen(SiteURL);
}

function LaunchFullScreen(url) {
 // set the window parameters
 params  = 'width='+screen.width;
 params += ', height='+screen.height;
 params += ', top=0, left=0';
 params += ', fullscreen=yes';
 params += ', resizable=yes';
 params += ', scrollbars=yes';
 params += ', location=yes';

 newwin=window.open(url,'windowname4', params);
 if (window.focus) {
     newwin.focus()
 }
 return false;
}
```

Using a SWITCH statement

```
function GetFormType() {
    var FormType = Xrm.Page.ui.getFormType();
    if (FormType != null) {
        switch (FormType) {
            case 1:
                return "create";
                break;
            case 2:
                return "update";
                break;
            case 3:
                return "readonly";
                break;
            case 4:
                return "disabled";
                break;
            case 6:
                return "bulkedit";
                break;
            default:
                return null;
        }
    }
}
```

Pop an Ok/Cancel Dialog

```
function SetApproval() {
    if (confirm("Are you sure?")) {
        // Actions to perform when 'Ok' is selected:
        var Approval = Xrm.Page.data.entity.attributes.get("new_phaseapproval");
        Approval.setValue(1);
        alert("Approval has been granted - click Ok to update CRM");
        Xrm.Page.data.entity.save();
    }
    else {
        // Actions to perform when 'Cancel' is selected:
        alert("Action cancelled");
    }
}
```

Here is a little more info that will help you get your head around the general design of all this…

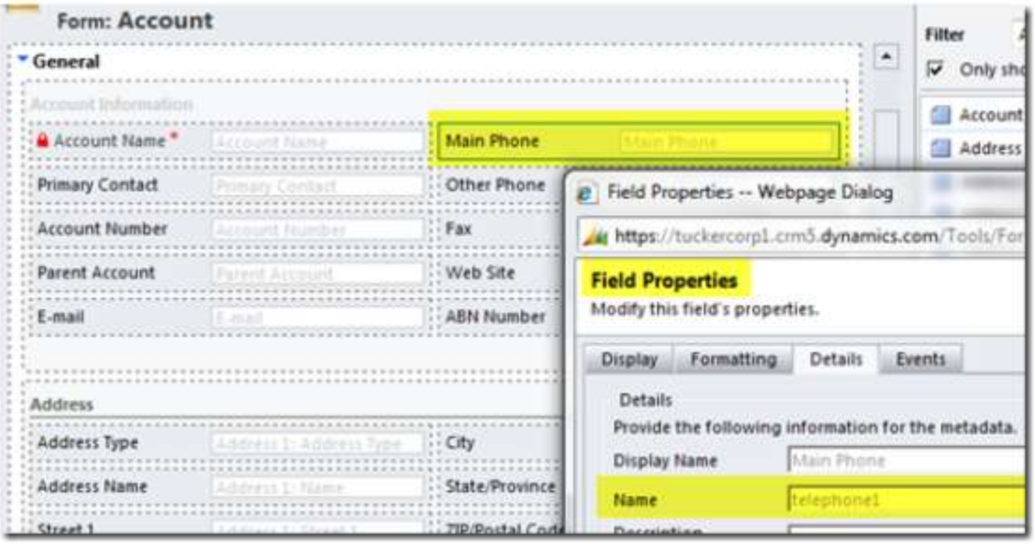
Depending upon what you want to do you will interact with one of the following:

Xrm.Page.data.entity.attributes – The data fields represented by fields on the form

Xrm.Page.ui.controls – The user interface controls on the form

Xrm.Page.ui.navigation.items – The navigation items on the form

When referring to fields or controls you must specify the name of the field and surround with quotes (and make sure you get the case right):



Display Name: Main Phone

Name: telephone1

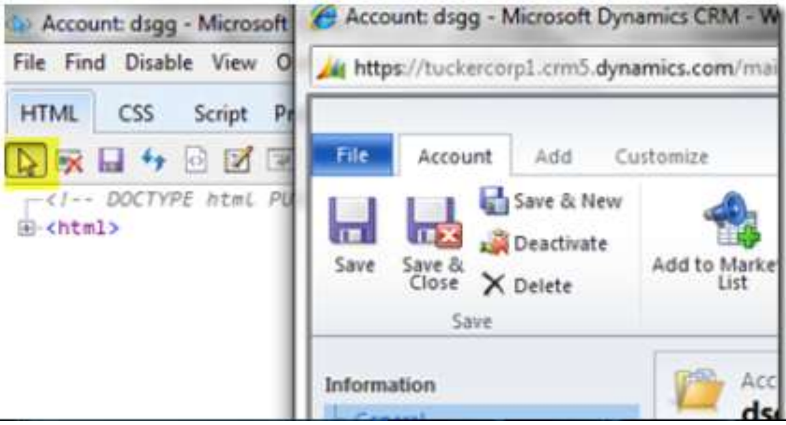
When referring to nav items you must specify the nav ID and surround it with quotes. To determine the nav ID:

- open the CRM form in IE

- hit F12 to activate IE's developer tools (if it doesn't appear, check for it under Task Manager and maximise it from there)

- in the developer tools window click the arrow to activate the "Select element by click" mode

- on the CRM form click the nav item (the dev tools window will take you to HTML definition of that element)


Account: dsgg - Microsoft
File
Find
Disable
View
HTML
CSS
Script
<!-- DOCTYPE html PU
<html>
Account: dsgg - Microsoft Dynamics CRM - W
https://tuckercorp1.crm5.dynamics.com/mai
File
Account
Add
Customize
Save
Save & Close
Save & New
Deactivate
Delete
Add to Marke List
Save
Information
Acc
ds