

Dynamics CRM 2013 JavaScript debugging in Internet Explorer 11

Microsoft rebuilt the F12 tools from the ground up in Internet Explorer11. They have a brand new UI and new functionality to make your developing and debugging faster and easier.

You can access them from within a browser window by hitting F12. If your keyboard doesn't have function keys, you can select '*F12 developer tools*' from the '*Tools*' menu.

You can use the **Debugger tool** to examine what your code is doing, when it's doing it, and how it's doing it. Pause code in mid-execution, step through it line-by-line, and watch the state of variables and objects at each step. New features in the Debugger tool include:

- No-refresh debugging. Set your breakpoints and go without reloading and losing state.
- Tabbed document interface for easier management of multiple scripts.
- Scrollbar that highlights breakpoints and search matches.

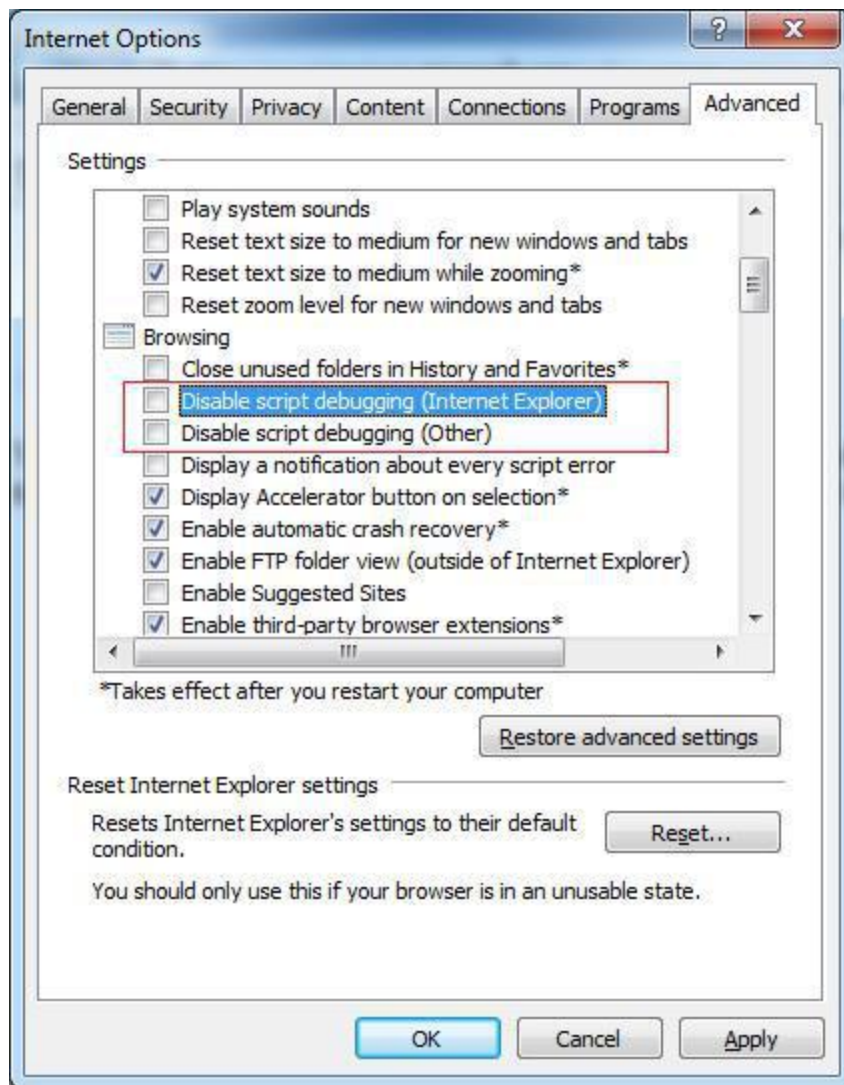
Development and debugging tasks it makes easier:

- Seeing what led to a function call using the Callstack.
- Making compressed or minified code more readable.
- Monitoring web worker creation and execution.

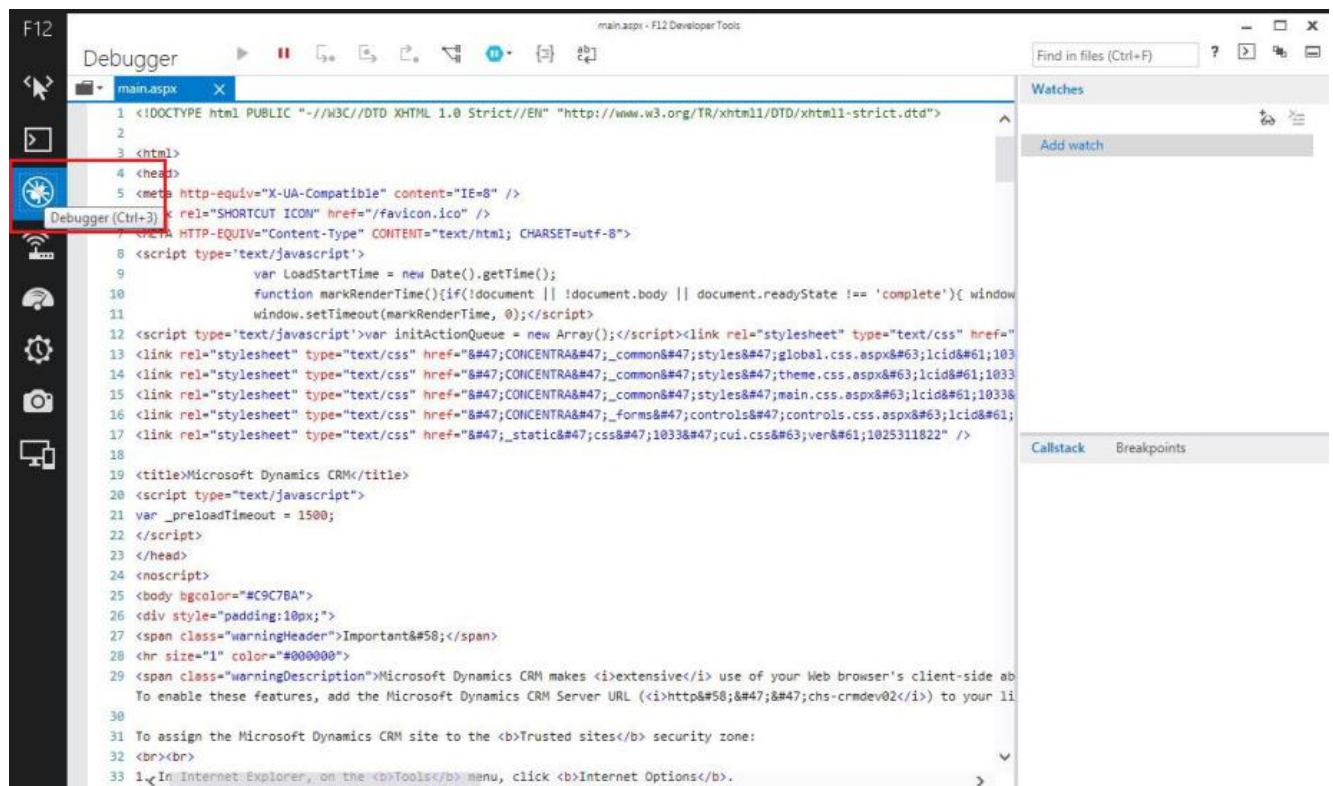
There are eight distinct tools, each with their own tab in the F12 tools interface. Here you'll find an image of each tool, a quick summary of what it does and what's new, and a couple of typical development or debugging tasks the tool makes easier. Check the link 'Using the F12 developer tools in Internet Explorer 11' [http://msdn.microsoft.com/en-us/library/ie/bg182326\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ie/bg182326(v=vs.85).aspx)

Here are the steps how to debug Dynamics CRM JavaScript code in Internet Explorer11.

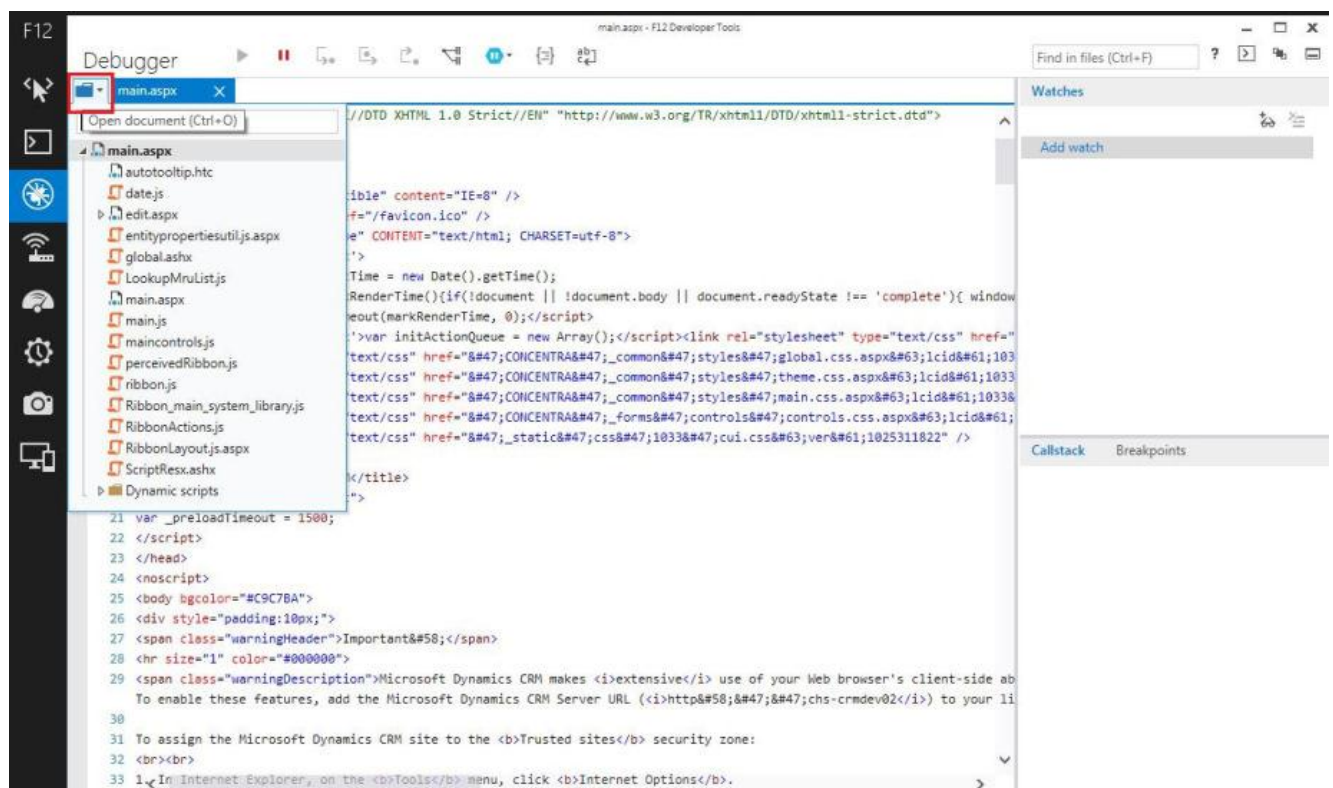
Step1: Before you try to debug JavaScript code make sure 'Disable script debugging (Internet Explorer)' and 'Disable script debugging (Other)' options are un-checked in 'Advanced' tab of Internet Explorer options, these options are checked by default.



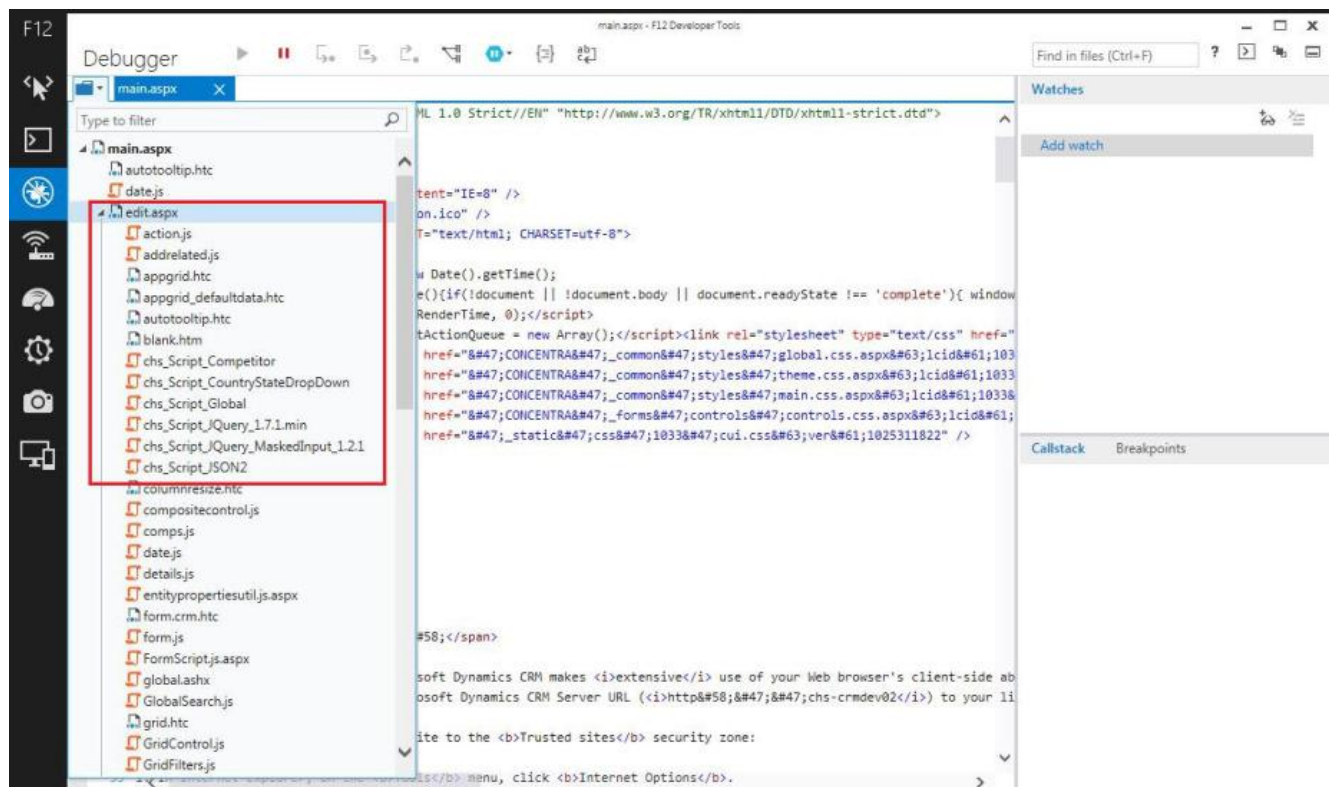
Step2: Press the **F12** key on your keyboard to open the tools or select '*F12 developer tools*' from the '*Tools*' menu. You will see '*DOM Explorer tool (CTRL + 1)*' left side, click on '*Debugger (CTRL+3)*'



Step3: Click on 'Open Document (CTRL+O)' and check CRM page.



Step4: Expand CRM page in 'Open Document (CTRL+O)' and check your JavaScript file is loaded and click on your JavaScript file and set the debugger wherever you want



F12

Debugger

main.aspx | chs_Script...mpetitor | chs_Script_JSON2 | chs_Script...7.1.min

```
1 // Populate Territory Field Based on Change to Zip Code Field
2 function ZipCode_OnChange() {
3     var PostalCode = Xrm.Page.getAttribute("address1_postalcode").getValue();
4
5     if (PostalCode != null && PostalCode != "") {
6         var zipCodeTerritory = getZipCodeTerritory(PostalCode);
7         //populate the
8         var territory = zipCodeTerritory;
9         var territoryValues = [];
10        if (zipCodeTerritory) {
11            territoryValues.push(territory);
12            territoryValues.push(territory);
13            territoryValues.push(territory);
14            territoryValues.push(territory);
15            if (territoryValues.length > 0) {
16                Xrm.Page.getAttribute("chs_territory").setValue(territoryValues);
17            }
18        }
19        else {
20            alert("The zip code, " + PostalCode + ", cannot be found in C4.\n\nPlease review the data and");
21        }
22    }
23
24    //function to get ZipCode Territory based on ZipCode
25    function getZipCodeTerritory(zipCode) {
26        // Set REST URL parameter for later use
27        oDataPath = location.protocol + "://" + location.host + "/" + Xrm.Page.context.getOrgUniqueName();
28        oDataPath = oDataPath + "/xrm/services/2011/organizationdata.svc";
29        // Retrieve Zip Code Data
30        var oColumns = "chs_TerritoryZipCodeId";
31        var oFilters = "chs_ZipCode eq '" + zipCode + "'";
32        var oDataParams = "?chs_zipcodeSet?$select=" + oColumns + "&$filter=" + oFilters;
33        var oDataURL = oDataPath + oDataParams;
34        var zipCodeTerritory;
```

Debugger window showing a function call and a variable watch.

Variable: zipCodeTerritory (object)

- prototype: [object]
- __metadata: [object]
- Id: "2b43b986-59b3-e211-93c9-0050569102..."
- LogicalName: "territory"
- Name: "CNE510"

Watches

[Locals]

- this: [object DisHTMLWindow2]
- arguments: [object (Arguments)]
- PostalCode: "01001"
- zipCodeTerritory: [object]
- territory: undefined
- territoryValues: undefined
- [Globals]: Add watch

Callstack

Breakpoints

Main frame

- ZipCode_OnChange: chs_Script_Com...
- address1_postalcode_onchange_han...: FormScript.js.as...
- Anonymous function: globalashx; 5
- fireOnChange: form.js; 1
- \$1c: form.js; 1
- \$2V_4: form.js; 1
- \$4I_4: form.js; 1
- Anonymous function: globalashx; 5
- b: globalashx; 5