ServiceNow Application Developer

Client-side Scripting > Exercise: Debugging Client-side Scripts

In this exercise you will practice debugging client-side scripts:

- JavaScript Log and *jslog()*
- Field Watcher
- *try/catch*

Creating an Error Filled Client Script

1. If the *NeedIt* application is not open in Studio from the last exercise, open it now.

   a. In the main ServiceNow browser window use the Application Navigator to open **System Applications > Studio**.

   b. In the *Select Application* dialog, click the **NeedIt** application.

2. Create a Client Script.

   a. In Studio, click the **Create Application File** link.

   b. In the *Filter...* field enter the text **Client** OR select **Client Development** from the categories in the left hand pane.

   c. Select **Client Script** in the middle pane as the file type then click the **Create** button.

3. Configure the Client Script:

    Name: **NeedIt Practice Debugging**

    Table: **NeedIt [x_<your_company_code>_needit_needit]**

    UI Type: **All**

    Type: **onSubmit**

    Description: **Practice debugging techniques in ServiceNow.**

4. Replace the contents of the *Script* field with this script:

```
function onSubmit() {
    try{
        var x === 7;
        jslogger("This message is from jslog().");
        jslog("The value of x = " + x)
        jslog("The NeedIt State value = " + g_form.getValue('state'));
        jslog("The currently logged in user is " + g_user.getFullName() +
".");



g_form.setValue('short_description',g_form.getValue('short_description') + "
-  Updated field value");
    }
    catch(err){
        jslog('A JavaScript runtime error occurred: ' + err.message);
   }
```

Practice Debugging

Although you may quickly see errors in the script without debugging, it is important to practice debugging techniques. Please follow these steps without giving in to the temptation to fix the script without practicing the different strategies.

1. Fix the errors and warnings from the syntax checker.

   a. The syntax editor shows errors and warnings next to the script line numbers. Hover the mouse over the error icon on line 1. Use the error message to determine the cause of the problem and fix it.

   b. Hover the mouse over the error icon on line 3. You will see both errors and warnings in the messages. Use the messages to determine the cause of the problem and fix it.

   c. Hover over the warning icon on line 5. Use the message to determine the cause of the problem and fix it.

   d. Click the **Submit** button to save the changes.

   ▶ **CHALLENGE**: Fix the syntax checker errors and warnings.

2. Switch to the main ServiceNow browser window. Use the browser's reload button to load the new client-side logic.

3. In the Application Navigator, open **NeedIt > All**.

4. Open the *NeedIt* record of your choice for editing.

5. Open the JavaScript Log and practice debugging the *jslog()* messages.

   a. In the main ServiceNow browser window, click the **Settings** button (⚙).

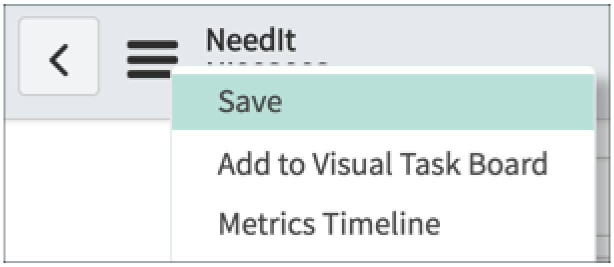
   b. Open the **Developer** pane.

   c. Enable the **JavaScript Log and Field Watcher**.

   d. Close the *System Settings* dialog.

   e. Make a change to the record.

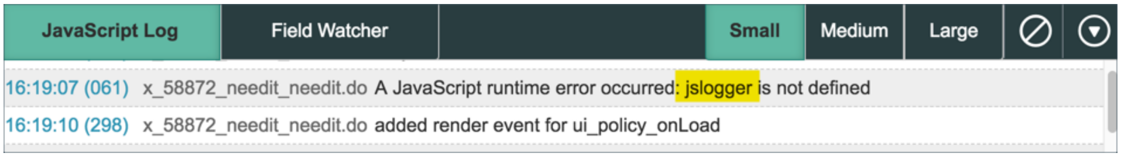
   f. To save the record and remain on the form, click the **Additional actions** menu (☰) in the form header and select the **Save** menu item. DO NOT click the *Submit* button as this will save the record but close the form.



6. Use your browser's find capability to look for the string **error**. You should see a runtime error. Use the error message to determine the cause of the error and make a correction to the script.

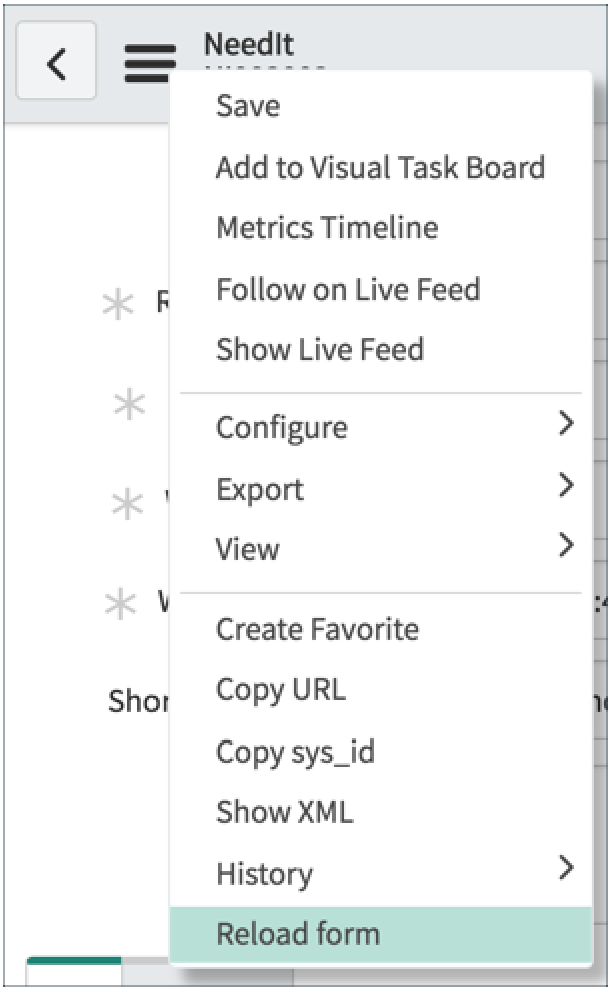
▼ **QUESTION**: How do you fix the runtime error?

SOLUTION: The *try/catch* statement trapped a call to a method that does not exist. Edit the script and change *jslogger* to **jslog**.



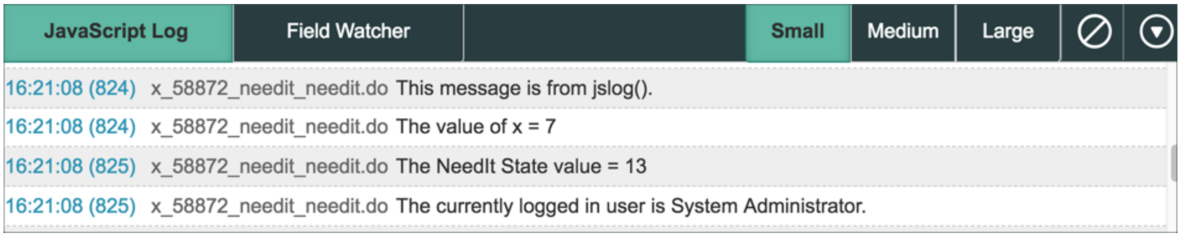
```
16:19:07 (061)  x_58872_needit_needit.do A JavaScript runtime error occurred: jslogger is not defined
16:19:10 (298)  x_58872_needit_needit.do added render event for ui_policy_onLoad
```

7. Return to the main ServiceNow browser window and reload the *NeedIt* form.



8. Make a change to the record and **Save** (not Update) the record.

9. Use the browser's find capability to search for **jslog**. You should see the log messages. Your *NeedIt State* value may be different than the value shown.



```
16:21:08 (824)  x_58872_needit_needit.do This message is from jslog().
16:21:08 (824)  x_58872_needit_needit.do The value of x = 7
16:21:08 (825)  x_58872_needit_needit.do The NeedIt State value = 13
16:21:08 (825)  x_58872_needit_needit.do The currently logged in user is System Administrator.
```

10. Return to Studio and make the *NeedIt Practice Debugging* Client Script **inactive**.