HCL Debugging

Monday, 17 February 2025 10:00 pm

Ahhh, **HCL Notes 9**—a bit old, but **still workable!** 💪🔥 Since HCL took down some of the older documentation, I'll break down **what actually works for debugging in Notes 9** for you! 🎀

🛠 1️⃣ Debugging LotusScript in Notes 9

Notes 9 **DOES have a debugger**, but it's old-school and can be unreliable. Here's how to activate it:

✅ **Method 1: Use the Built-in LotusScript Debugger**

1️⃣ Open **Domino Designer (DDE)**
2️⃣ Go to **Tools → Debug LotusScript**
3️⃣ **Press F5 while running your script** to step through it
🚨 **If the debugger does NOT launch:**

- Make sure **you have script debugging enabled** (Preferences > LotusScript Debugging).
- **Check if the script is signed properly**—unsigned agents might not execute.

✅ **Method 2: Use Print and MsgBox for Manual Debugging**

If the built-in debugger **is buggy or disabled**, **use these instead**:

```
lotusscript
CopyEdit
MsgBox "Checkpoint 1: Script started!"
MsgBox "Value of X: " & X
```

or print output to the Notes log:

```
Lotusscript
==========
CopyEdit
Print "Debug Info - Value of X: " & X
```

This **forces Notes to reveal what's happening**, even if the debugger fails.

📜 2️⃣ Checking Logs for Errors in Notes 9

Notes **doesn't always tell you what went wrong**—you have to check the logs.

✅ **Where to Find Errors in Notes 9**

- ◇ Open log.nsf (**File → Database → Open** → log.nsf)
- ◇ Look for **Agent Errors, Execution Logs, or Script Failures**
- ◇ If you don't see anything useful, **enable debug logging** (next step 👇)

✅ **Method 2: Enable notes.ini Debugging**

If logs **aren't detailed enough**, modify **notes.ini** to get better debugging output.

Steps: 1️⃣ Open your notes.ini file (usually in C:\Program Files\IBM\Lotus\Notes).
2️⃣ Add these lines:

```
Ini
==========
CopyEdit
Debug_Outfile=c:\notes_debug.txt
Debug_LS_Dump=1
Debug_Show_Console=1
```

3️⃣ Restart Notes.
4️⃣ Check C:\notes_debug.txt for logs.

🐞 3️⃣ Debugging Formula Language in Notes 9

If you're using **Formula Language**, debugging is even **more limited** than LotusScript.

✅ **How to Debug Formula Language**

- ◇ Use @Prompt to manually display values:

```
Formula
==========
CopyEdit
@Prompt([Ok]; "Debug Info"; "Value of X is: " + @Text(X))
```

- ◇ Use @SetProfileField to store debugging values inside a profile document for later review.

💻 4️⃣ Debugging JavaScript in Notes 9 Web Apps

If your script runs in Notes via a web browser, debugging works like any normal web app.

✅ **Steps to Debug JavaScript in Notes 9:**

1️⃣ **Open the Web Application in Chrome/Firefox**
2️⃣ Press **F12** (Open Developer Tools)
3️⃣ Go to the **Console** tab and type:

```
javascript
==========
CopyEdit
console.log("Checkpoint 1: Script loaded!");
```

4️⃣ Use console.log(variable); to track JavaScript execution.

🚨 5️⃣ Common Errors & Fixes in Notes 9

✅ **Issue: "Agent did not run" (LotusScript agent not executing)**

- **Solution:** Make sure the agent is **signed** by an authorized user.

✅ **Issue: "Field Not Found"**

- **Solution:** Check for missing fields:

```
lotusscript
==========
CopyEdit
If document.HasItem("FieldName") Then
    MsgBox "Field Exists!"
End If
```

✅ **Issue: Notes crashes when script runs**

- **Solution:** Use On Error Resume Next to catch where it fails:

```
lotusscript
==========
CopyEdit
On Error Resume Next
```

✅ **Issue: JavaScript in Notes Web Forms Doesn't Work**

- **Solution:** Open **DevTools (F12)** and check for errors.

🛠 6️⃣ Best Practices for Debugging Notes 9

✓ **Use MsgBox & Print for instant feedback** (since the debugger is unreliable)
✓ **Check log.nsf frequently**—it records agent crashes and script failures
✓ **Enable notes.ini debugging flags** for extra error logs
✓ **Run scripts on a test database first** before deploying
✓ **Make backups of .nsf files** before modifying scripts

🔥 Final Thoughts: Debugging in Notes 9 is Hard, but Not Impossible!

💬 **You just need the right mix of debugging tools, logs, and patience.**
Even if **IBM/HCL took down the documentation**, these tricks **WILL still work**.