

Homebrew Call Sign Lookup and Logger

Keeping a log of your amateur activity is complicated when you are away from your main machine. Sure, pen and paper will suffice, but that call sign lookup feature is missing. It is nice to know instantly the name of the person on the other end of the conversation even before he or she is done talking. Over 100 different logging software options exist, and the prices of these options vary extravagantly. When you choose your software, you are most likely going to be stuck using one machine.

You can use a free Google account in tandem with a free HamQTH account to build your own call sign lookup and logging software that runs on any Internet connected machine. This will allow you to keep a detailed record of your activity while you are away from your main machine. Imagine that your desktop is back at the shack and you just have your Linux laptop handy for the road. No problem. Open up this spreadsheet that you are about to make and log away. Later, you can update your main electronic logbook without having to do anything more than just open your custom Google logger in its own window and start transcribing.

First you will need to get a free HamQTH account from www.HamQTH.com. Record your user name and password for use in the Google spreadsheet that you will make according to the following instructions.

Next, from your Google home page, go to Google Drive and create a new spreadsheet by clicking the red “New” button and selecting “Google Sheets”, as shown in Figure 1. Name your new spreadsheet whatever you want. For reference purposes in the spreadsheet, as shown in Figure 2, edit the following four cells accordingly: A1 - My Call Sign, B1 - My HamQTH Password, A3 - SessionID, and A5 - Callsign. That is all you need to do to the spreadsheet.

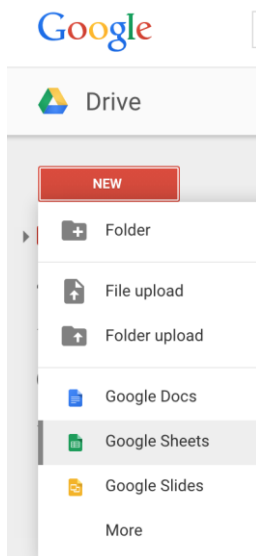


Figure 1 — You can make a new spreadsheet by clicking the red ‘New’ button and selecting ‘Google Sheets’ from the drop down menu.

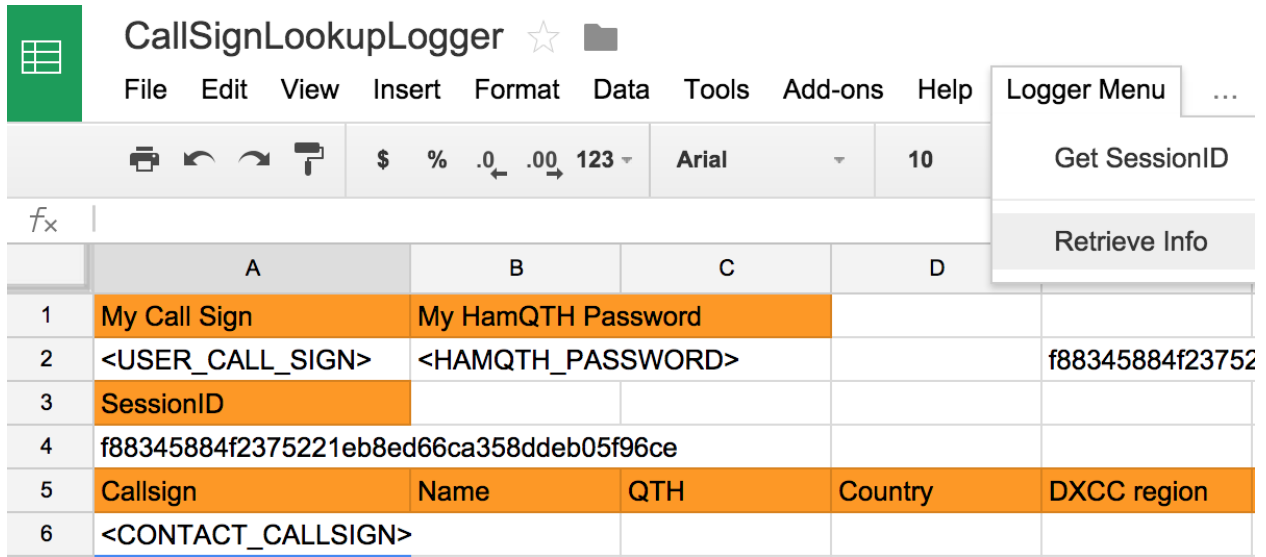


Figure 2 — Rename your spreadsheet anything you want by using the menu's File -> Rename option. The file 'CallSignLookupLogger' has highlighted values in A1, B1, A3, and A5:E5. You enter the important values in A2, B2, and A6. The string of letters and numbers in A4 is generated by the menu's Logger Menu -> Get SessionID function. Before you run this program, replace <USER_CALL_SIGN> with your call sign. Likewise, replace <HAMQTH_PASSWORD> with your HamQTH password and <CONTACT_CALLSIGN> with the call sign of the person whose information you wish to retrieve.

Now you need to get into the heart of what makes this logger work. From the menu, click on "Tools" and "Script Editor...", as shown in Figure 3. A new tab will open in the browser with a greyed out "Untitled Project" in the background behind a small window labeled "Google Apps Script". Click on the first option that you see, as shown in Figure 4. You are going to create a script for a "Blank Project".

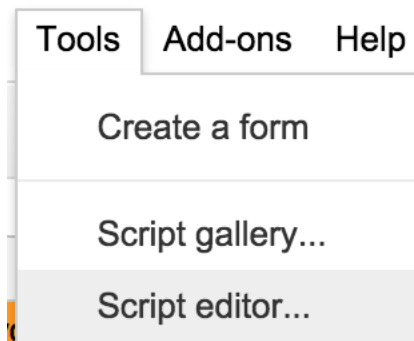


Figure 3 — From the several options available in the menu bar, select Tools -> Script editor...

Google Apps Script

Create script for







-  Blank Project
-  Google Sheets Add-on
-  Google Docs Add-on
-  Google Forms Add-on
-  Custom Functions in Sheets
-  Web App

Figure 4 — The 'Blank Project' actually has three lines of code in it. You will see the words "function myFunction", some parentheses, and some braces. Your next step will be to delete those three lines and copy the code included in this article.

Delete the three lines of code in the editor. Add the following lines of code:

```
// This code contains a series of functions that can be used
// to look up and log contacts using a HamQTH account.
// 1st - enter user call sign and HamQTH password in the spreadsheet to obtain a
// sessionID used to retrieve information from HamQTH
// 2nd - enter contact's call sign under 'Call Sign' and retrieve info; add notes as needed
// 3rd - insert four rows under 'Call Sign' making room for next contact call sign.

// This function, onOpen(), adds the "Logger Menu" to the standard
// Google Sheets menu. From the Logger Menu, you can select either
// "Get SessionID" or
// "Retrieve Info". You need to get the sessionID before you can begin to use the
// importxml function. According to HamQTH, the sessionID is good for one hour. If you
// want to save your callsign searches, you will need to copy them and paste them as
// 'values only' into another sheet or portion of the same sheet.

function onOpen() {
  var ui = SpreadsheetApp.getUi();
  ui.createMenu('Logger Menu')
  .addItem('Get SessionID', 'getSessionID')
  .addSeparator()
  .addItem('Retrieve Info', 'retrieveInfo')
  .addToUi();
}
```

```
// Below is the importxml function that works to retrieve sessionID from HamQTH
//
//
=importxml("http://www.hamqth.com/xml.php?u=<USER_CALL_SIGN>&p=<USER_PA
SSWORD>", "node()[1]")
//
// This function getSessionID reports the sessionID and places it in E2.
// It also copies it to A4 as value.
```

```
function getSessionID(){
  var ss = SpreadsheetApp.getActiveSpreadsheet();
  var sheet = ss.getActiveSheet();
  var urlStrPrefix = "http://www.hamqth.com/xml.php?u=";
  var urlStrAffixOne = "&p=";

  // Get USER_CALL_SIGN and USER_PASSWORD into variables
  var cellRefCallSign = sheet.getRange("A2");
  var cellRefUserPassword = sheet.getRange("B2");
  var userCallSign = cellRefCallSign.getValue();
  var userPassword = cellRefUserPassword.getValue();

  // Run importxml command with variables
  // Set sessionID into cell A4 as value copied from E2
  var command = "importxml(\"" + urlStrPrefix + userCallSign + urlStrAffixOne +
  userPassword + "\", \"node()[1]\")";
  var cellRefSessionID = sheet.getRange("E2");
  cellRefSessionID.setFormula(command);
  var globalSessionID = cellRefSessionID.getValue();
  sheet.getRange("A4").setValue(globalSessionID);
}
```

```
// Now copy the input contact's call sign from A6, copy sessionID from A4,
// perform importxml search, and insert four new rows.
// Need following importxml command:
//
////
=importxml("http://www.hamqth.com/xml.php?id=<SESSION_ID>&callsign=<CALLSIG
N>&prg=googleSheets", "//node()[2]")
```

```
function retrieveInfo() {
  var ss = SpreadsheetApp.getActiveSpreadsheet();
  var sheet = ss.getActiveSheet()
  var urlStringPrefix = "http://www.hamqth.com/xml.php?id=";
  var urlStringAffixOne = "&callsign=";
  var urlStringAffixTwo = "&prg=googleSheets";
```

```

// Get Call Sign and SessionID into variables
var cellReference = sheet.getRange("A6");
var sessionIDReference = sheet.getRange("A4");
var refCallSign = cellReference.getValue();
var refSessID = sessionIDReference.getValue();

// Run command
var command = "importxml(\"" + urlStringPrefix + refSessID + urlStringAffixOne +
refCallSign + urlStringAffixTwo + "\", \"//node()[2]\");";
var poplIntoThisCell = sheet.getRange("A7");
poplIntoThisCell.setFormula(command);

ss.insertRowsAfter(5, 4);
}

```

From the editor menu, go to “File” and choose “Save.” Name your project whatever you like. It is now ready to run. Close both the project editor and the spreadsheet. Open the spreadsheet again, and this time you will see the Logger Menu. You need to get a sessionID first, and the first time you try to run the “Get SessionID” function you will need to authorize the action. Go ahead and give the script permission to run. Enter a call sign in A6 and choose “Retrieve info” from the Logger Menu. Your fellow ham’s name will appear next to his or her call sign along with a whole slew of other pertinent information. Have fun!

You can have this project ready for action in less than 30 minutes. You do not need any special coding skills or technical knowledge beyond copy and paste. The copy and paste option becomes especially easy when you have access to a digital version of QST. Remember to copy and paste your contact information into another spreadsheet or portion of the same sheet before your sessionID expires. Do not worry if you forget. Your notes will not be erased. The call signs you entered will not be erased, but the retrieved data will disappear after one hour and be replaced with “Session does not exist or expired.” You can always resubmit the call signs later.

If you did want to learn more about the code involved, you could take the quick online course in Javascript offered by Codecademy. Google offers near endless pages of support if you should need it. HamQTH has a developer’s page with information on how to take advantage of their free service. In all, this project provides you with a call sign lookup and logging program that you can build to suit your needs. It is a convenient supplement to the logging software you already use, or you might just choose to use it as a clever homebrew solution.

Author information: David M. Schulz, WO6M, P.O. Box 536, Cedarville, CA 96104, email: WO6M@arrl.net