Tom VanHeuvelen
Autohotkey Basics
WFIU 2015
2015-05-14

Basics of Getting Autohotkey setup

The below instructions will help you install the macro program "Autohotkey", get an Autohotkey script set up, get it to automatically start when your computer starts, and will provide some useful codes.

1. **Install Autohotkey from their website:** Go to http://www.autohotkey.com/ and install the program from their homepage. Install it like you would any other program from the internet.
2. **Create an .ahk macro file:** After installing Autohotkey (AHK), creating a macro file is easy. Simply right click on your desktop. Move the cursor over "New." There will now be the option: "Autohotkey Script". Select this.
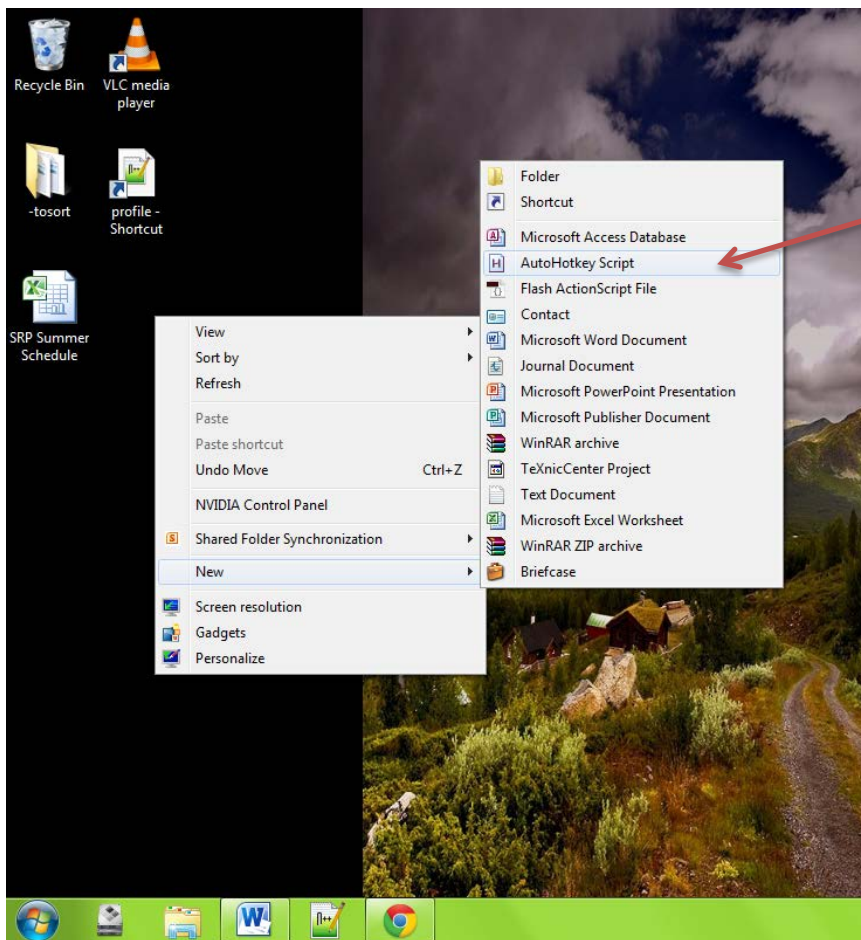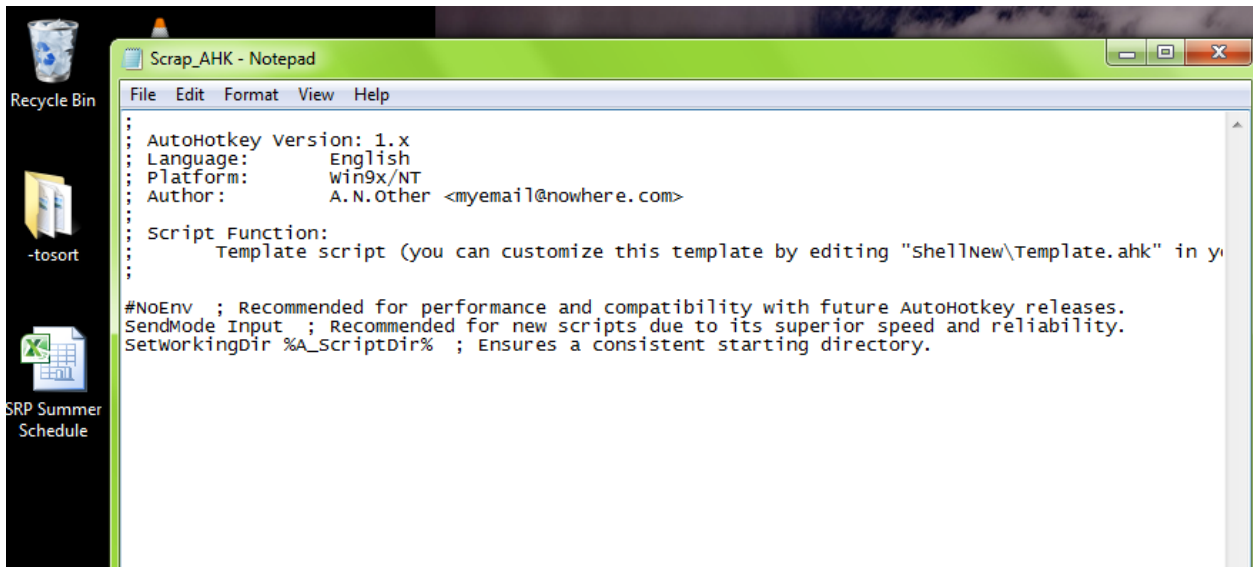
There should now be a new Autohotkey.ahk file on your desktop. Name it whatever you want.
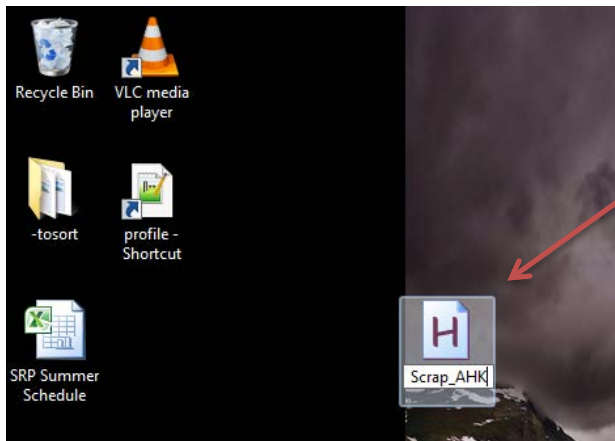
3. **Edit your Autohotkey Script:** To edit your AHK script, right click it and select "Edit Script." A new text file will open up. More information is in the next section regarding some useful macros. Make your changes by adding text below the already present text. Save. Exit.
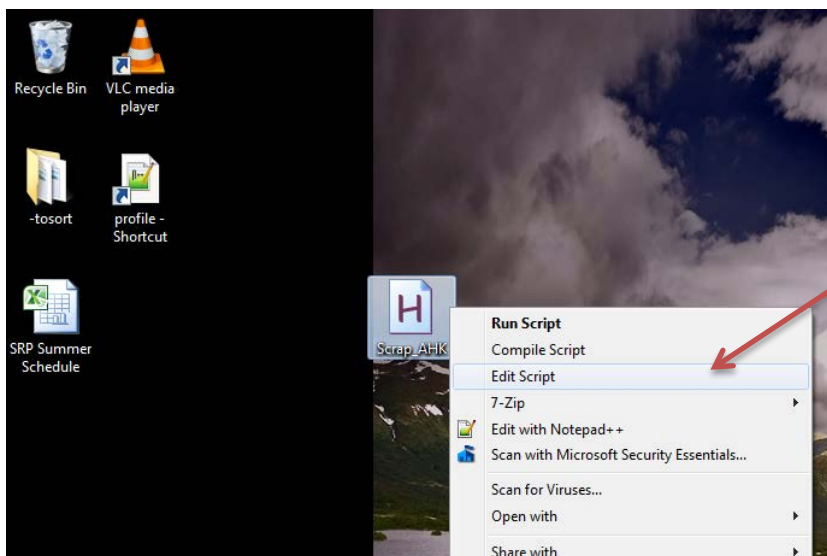
4. **Start your macro.** After you've added useful macro commands to your .ahk file, save and exit the text file. To get the file to start up, simply double click it. After this a green box with an "H" should show up in the Notification Area at the bottom right of Windows. In Windows 8, I've had to run Autohokey program as an administrator by right clicking the Autohotkey program from the Start menu and selecting "Run as administrator."
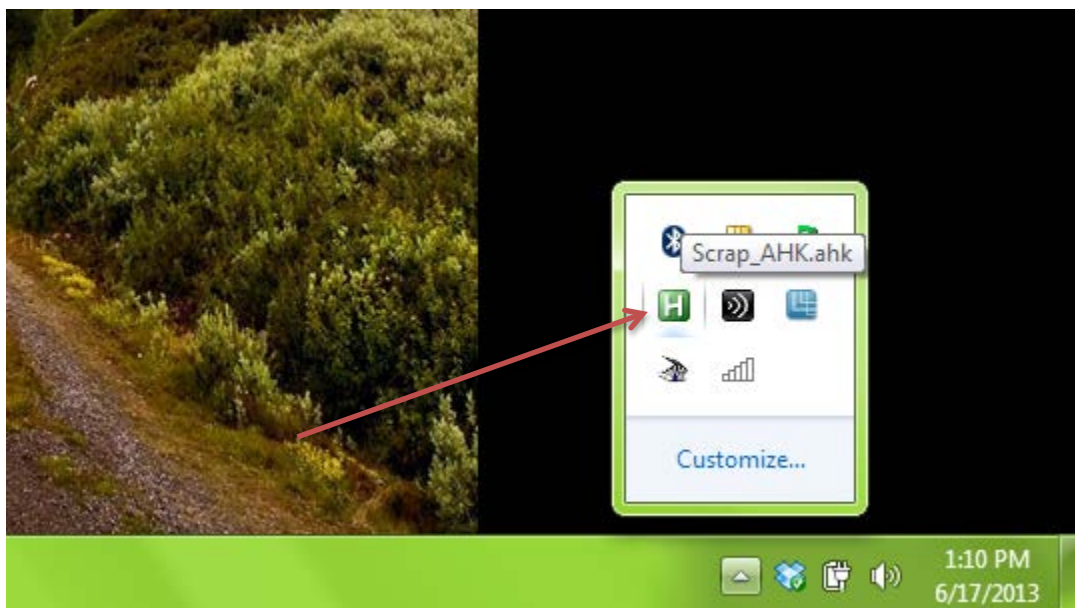
5. **Edit your .ahk script.** Sometimes you want to make changes to your macro script. To do this, right click on the AHK icon in the Notification area. Select "Edit Script." After you've made your changes, save and exit the text document. Right click on the AHK icon again and select "Reload this script." Voila!
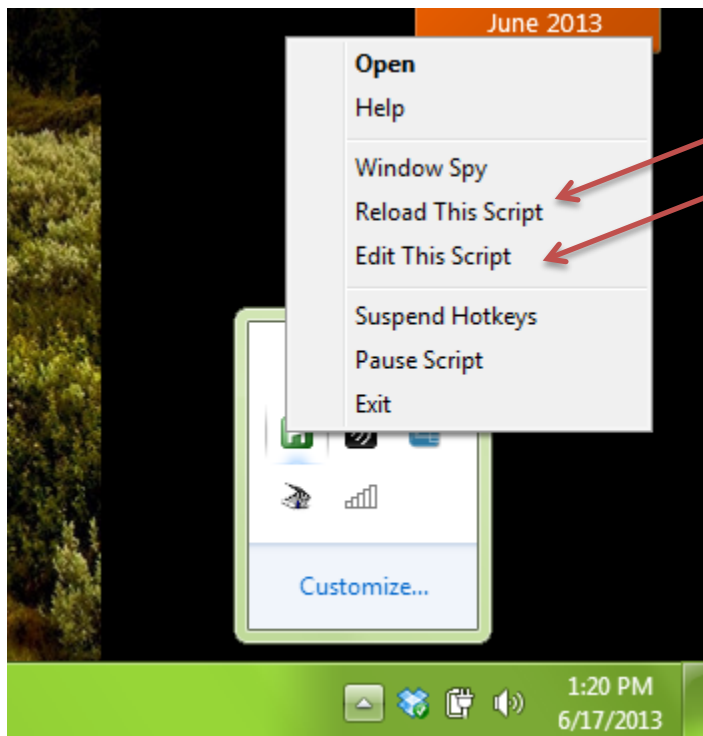
Some useful AHK languages, macros, and tips

AHK Language

Below are some useful commands to know. For simplicity, anything that can be directly copied into an .ahk file is `in this font`.

- A **semicolon** (;) creates a comment. Anything following a semi colon in an AHK script is considered a comment and will NOT be used as part of your macro. So don't use semicolons ANYWHERE except to create comments.
- The commands I use take the following basic shape:

  <keyboard or mouse shortcut> ::
  <List of commands to perform>
  <Exit Macro>

  There are all kinds of complexities you can incorporate, if you're so inclined: logical operators, mouse clicks, GUI boxes, etc. Refer to the help documents and the online forums provided by AHK if you want to get into these. I'm currently sticking with the basics.

- Keyboard Shortcuts. You may want to use certain keys on your keyboard for your macros: control, alt, shift, the space bar, etc. These buttons correspond to certain symbols in AHK:

Keyboard Key	AHK symbol
Windows Button	#
Control Button	^
Alt Button	!
Shift Button	+
Space Bar	Space
Tab Button	Tab

- Be patient! Sometimes your macros will blow up. Sometimes this happens to me and I need to start a new AHK macro from scratch. I build it one step at a time (adding a new macro, saving, testing. Then adding another macro, saving, testing, etc).

Time Stamp

The following code makes a timestamp that looks like this: 2013-06-17

```
; - Control & 1: Insert Date Time stamp
^1::
   FormatTime, xx,, yyyy-MM-dd
   SendInput, %xx%
return
```

Note that pressing the "Control" and the "1" buttons create the time stamp. Also note that I used a comment that describes the macro.

Opening a program

The following code opens my Chrome internet browser.

```
; - Windows Button & Shift & c: Run Chrome
#+C::
    run
"C:\Users\Tom\AppData\Local\Google\Chrome\Application\chrome.exe"
return
```

If you want to do this, you will need to find the file location of any of your .exe files and change the file directory. For example, if your chrome.exe file is in the location C:\Fakefile\chrome.exe, you'd need to change
"C:\Users\Tom\AppData\Local\Google\Chrome\Application\chrome.exe"
above to "C:\Fakefile\chrome.exe". You can open any program this way simply making a new macro as above with a different program.exe and corresponding file path.

Insert Text
To put an email signature, I simply type = and then "tom" to get the following:

Tom VanHeuvelen
Ph.D. Candidate
Indiana University

```
; - equal sign & tom: email signature

::=tom::
(
Tom VanHeuvelen
Ph.D. Candidate
Indiana University
)
return
```

Another example of sending text is:

```
; send k drive path
^!K::
Send \\bl-soc-theseus\groups$
return
```

Copy contents of a file, paste into new file

This is my preferred method of opening my default do-file, although the next section shows another methods. The below command reads and copies the contents of my default do-file, located in C:\Sociology and titled default-do-jsl. It then pastes this information into the text document that is currently open.

```
; copy default JSL do-file
#!N::
FileRead, Clipboard, C:\Sociology\default-do-jsl.do
Send, ^v
return
```

Advanced: Open and automatically save a do-file in Notepad++.

To run this code, you need the "default-do-jsl.do" file and have Notepad++ installed. The following code checks to see if Notepad++ is open. If it isn't, it then opens Notepad++. Then, it opens "default-do-jsl.do" and automatically selects "Save as". I do this because I'm lazy and don't want to remember the template for do-files. And if I don't have a program that automatically "Saves as" a new document, I continually write over my default do-file and get frustrated.

```
; Windows Key & Shift & s : default do-file
#+S::
Process, Exist, notepad++.exe
If Not ErrorLevel {
    Run "C:\Program Files\Notepad++\notepad++.exe"
    RunWait "C:\Stata\Tom\default-do-jsl.do", max
    send {Control Down}{Alt Down}s{Control Up}{Alt Up}
}
If (ErrorLevel != 0){
    RunWait "C:\Stata\Tom\default-do-jsl.do", max
    send {Control Down}{Alt Down}s{Control Up}{Alt Up}
}
Return
```

A few things to note:

1. The shortcut for this macro is: Windows Button & Shift Button & s button.
2. You need to change the location in the code above from "C:\Program Files\Notepad++\notepad++.exe" to wherever notepad++.exe is on your computer.
3. You need to change the location of "C:\Stata\Tom\default-do-jsl.do" in BOTH parts of the above code to wherever you saved default-do-jsl.do
4. I find it helpful to restrict my ability to write on the "default-do-jsl.do" file. I right click on it, select "Properties" and select the option "Read Only". This way it's more difficult to accidentally write over my default do-file.
5. default-do-jsl.do can be replaced by any .do file you'd like, though, as long as you change the name of the do-file in the code above.