

BASIC SAS Workshop

September 27, 2006

Mona Lee Ostrowski
Oswald #812

865-4278

monao@pop.psu.edu

POPNET User Account:

Information for requesting a PopNet user account can be found at:

http://www.pop.psu.edu/help/obtaining_account.htm

On the form under TYPE OF ACCOUNT check off both Windows and Unix accounts.

Should your research require access to restricted data, like Add Health or NICHD, you may have see the data archivist to get clearance for access and to fill out more forms.

UNIX

SSH Secure Shell:

The SSH Secure Shell is software that allows you to access servers remotely, much like modems.

To acquire a download of the software log into your Web Access
And go to **downloads.its.psu.edu** ; then select from **File Transfer** the
download for **SSH Secure Shell Windows** ;

The computers in the Oswald Computer Lab will have it on the Desk Top.

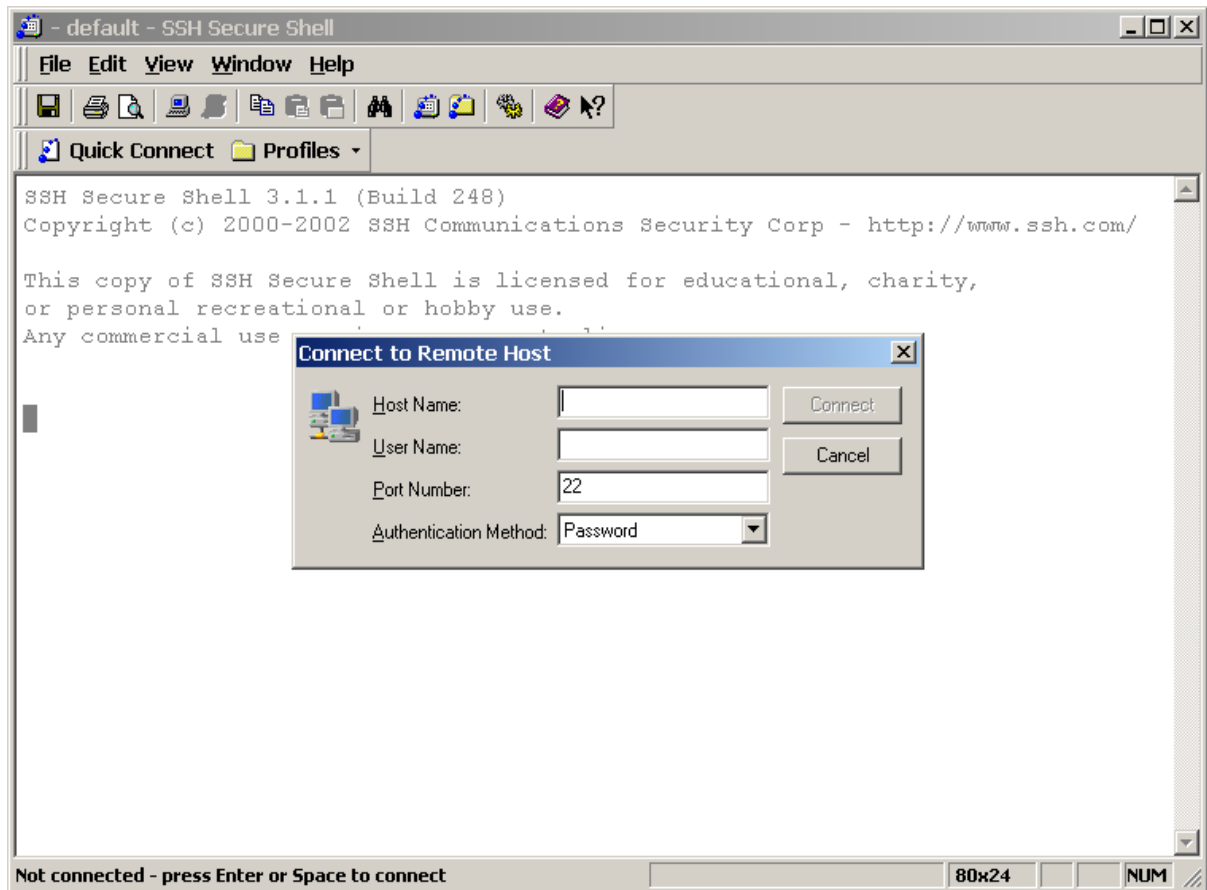
To start SSH:

Double click on the icon for **SSH Secure Shell Client** to start up the program;

*otherwise
in the Task bar*

START=>PROGRAMS=>SSH Secure Shell=>Secure Shell Client [double
click]

The resulting SSH window should look like this:



UNIX:

To Login:

In the *Host Name* box enter: (pick one)

UNIX servers:

Greenland.pop.psu.edu

Madagascar.pop.psu.edu

Tasmania.pop.psu.edu

LINUX servers:

Italy.pop.psu.edu

Israel.pop.psu.edu

Argentina.pop.psu.edu

In the *User Name* box enter

your pop user id => **CONNECT**

In the next window enter

your pop password

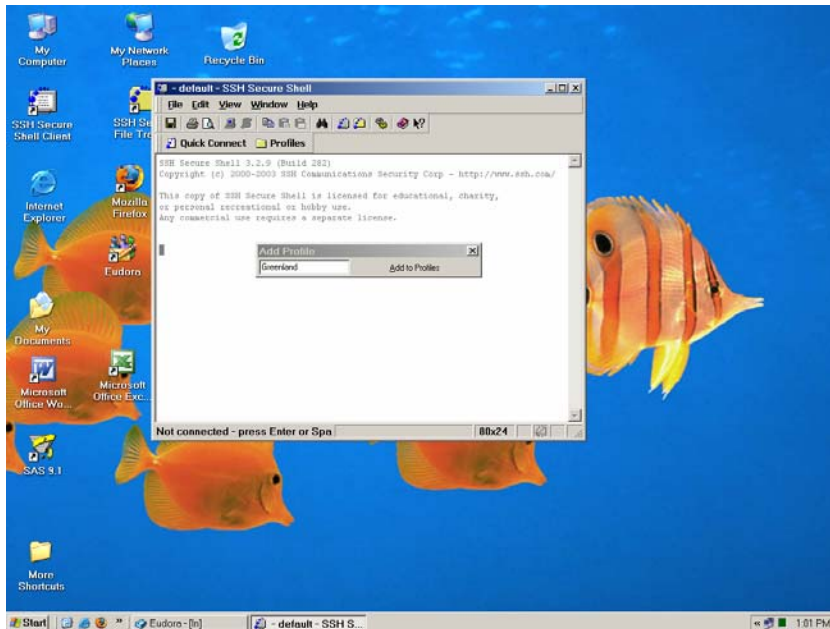
Profiles are information for servers to speed up the login process.
To set a profile:

Click on the Profiles button on the menu bar;

A drop down menu will appear => click **ADD Profile**;

Enter the name of the server (in the example: Greenland)

Click on **Add to Profile**;



Click on the Profiles button on the menu bar;

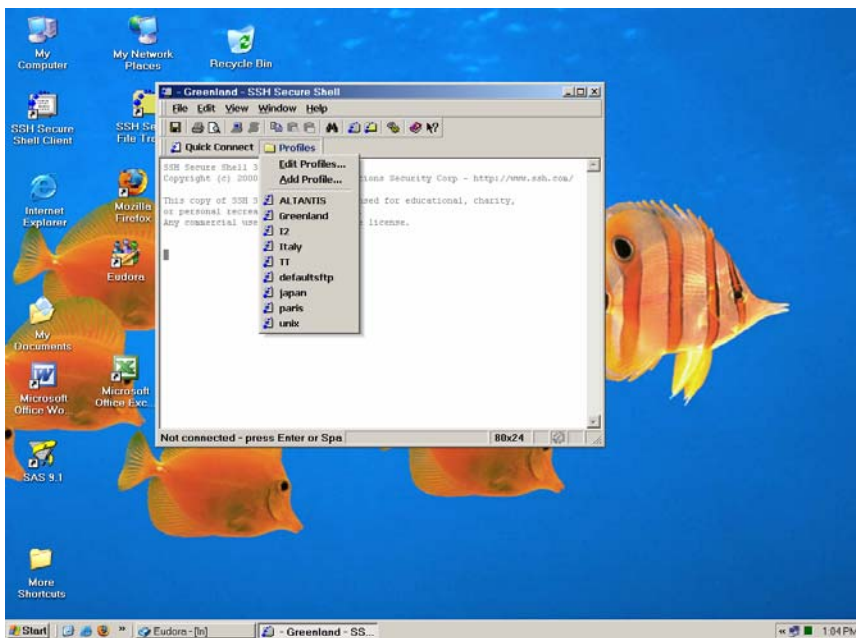
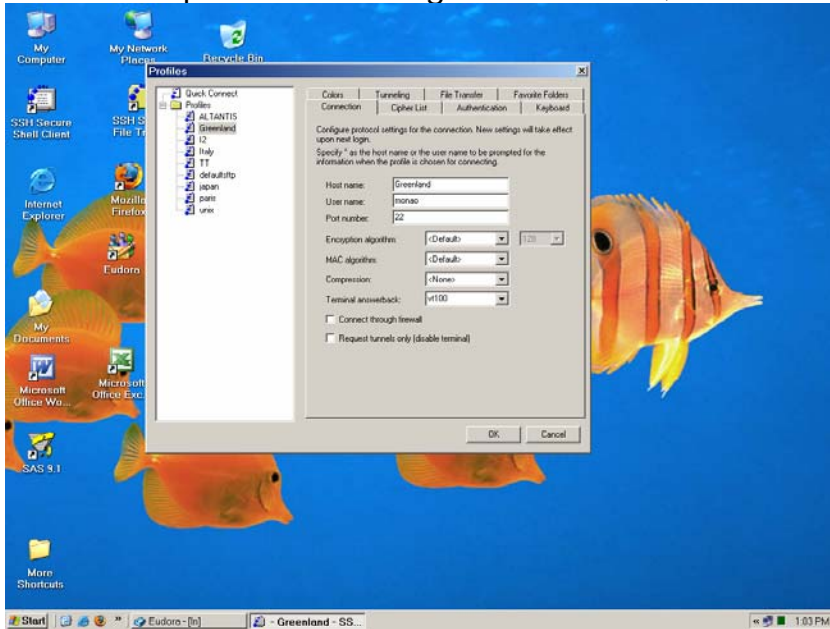
A drop down menu will appear => click **Edit Profile**;

The next window has the directory tree on the left of the Profiles folder;
Clicking on Greenland (the name you just created above) indicates you want the information on the right side of the window to be assigned to Greenland;
In this example Host server => Greenland

User name => monao (or your POP user id)

Port number => 22

Click OK to complete the editing of the Profile;



Some useful commands:

mkdir → creates a directory in your location;

ex: `>mkdir MySAS`

cd → change directory;

ex: `>cd MySAS` → locates you in the **MySAS** directory;

cat → show contents of file;

ex: `~/MySAS>cat sasp1.sas` → lists the file called **MySAS/sasp1.sas**

ls → list the contents of the directory you are located in;

ex: `~/MySAS>ls` → displays the directories/files in MySAS

sas program filename → submits for execution the SAS code in program filename;

sas program filename & → submits for execution the SAS code in program filename as a background job;

ex: `~/MySAS>sas sasp1.sas &` → submits the sas program in background
`>8907` → computer returns the job id number

ps -fu userid → shows the jobs you have running/submitted;

ex: `~/MySAS>ps -fu monao`

kill -kill pid → deletes or terminates the program with the job pid;

Useful if **run**; is forgotten in the program;

ex: `~/MySAS>kill 8907`

rm → remove a file or a directory and its entire contents;

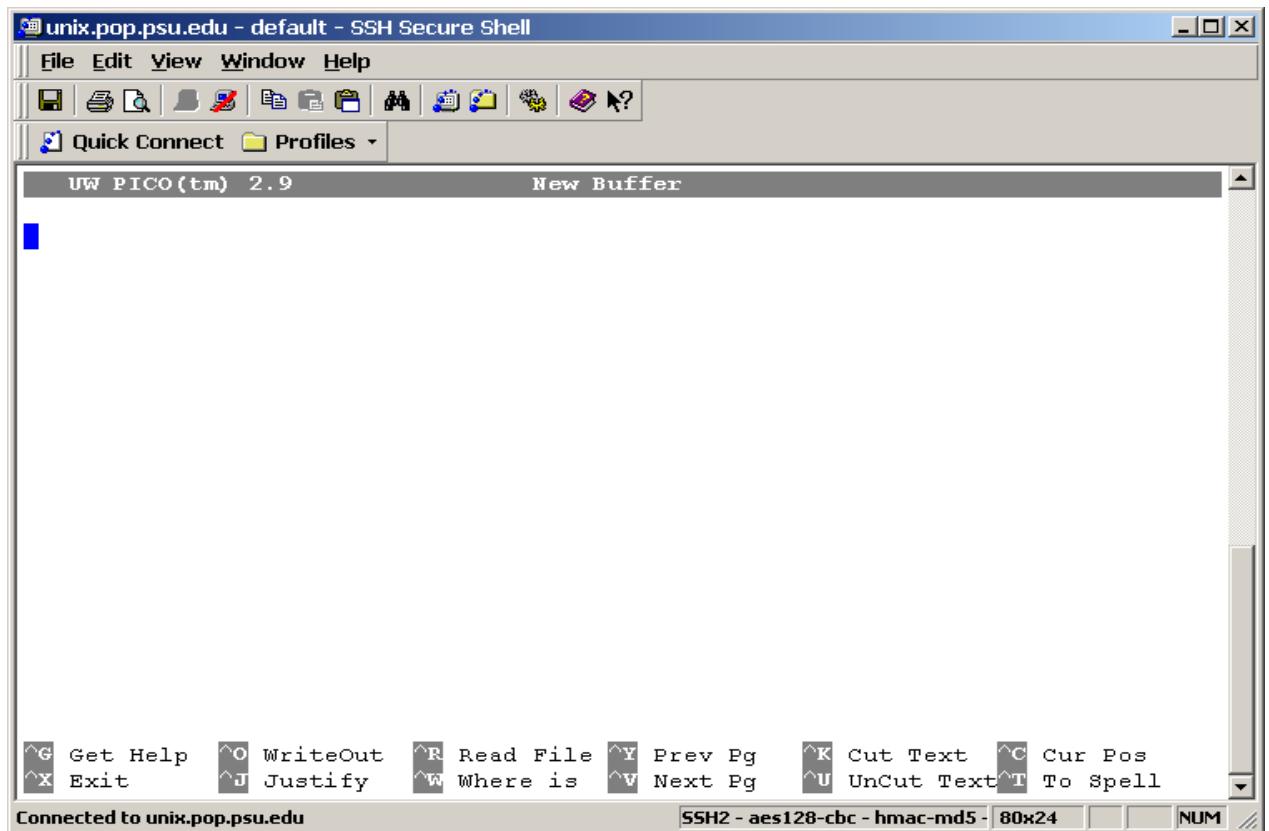
ex: `~/MySAS>rm sasp2` deletes the file called sasp2 in the MySAS directory

Editors:

So how do you write a SAS program? In UNIX you will need to use an editor. One editor is

pico:

Easy to use menu driven editor. Commands are visible at the bottom of the screen and adding text and modify text is fairly intuitive. A good editor for the beginning UNIX user.



When you are finished type the exit command (*control key & x key*). The **pico** editor will prompt you about saving the text entered and ask for the location/name to it save it to.

Another editor is
vi:

The editor **vi** is universally available, where **pico** is not always found on all systems. Although vi is more complicated and confusing, it is more powerful and flexible because of multiple file manipulation.

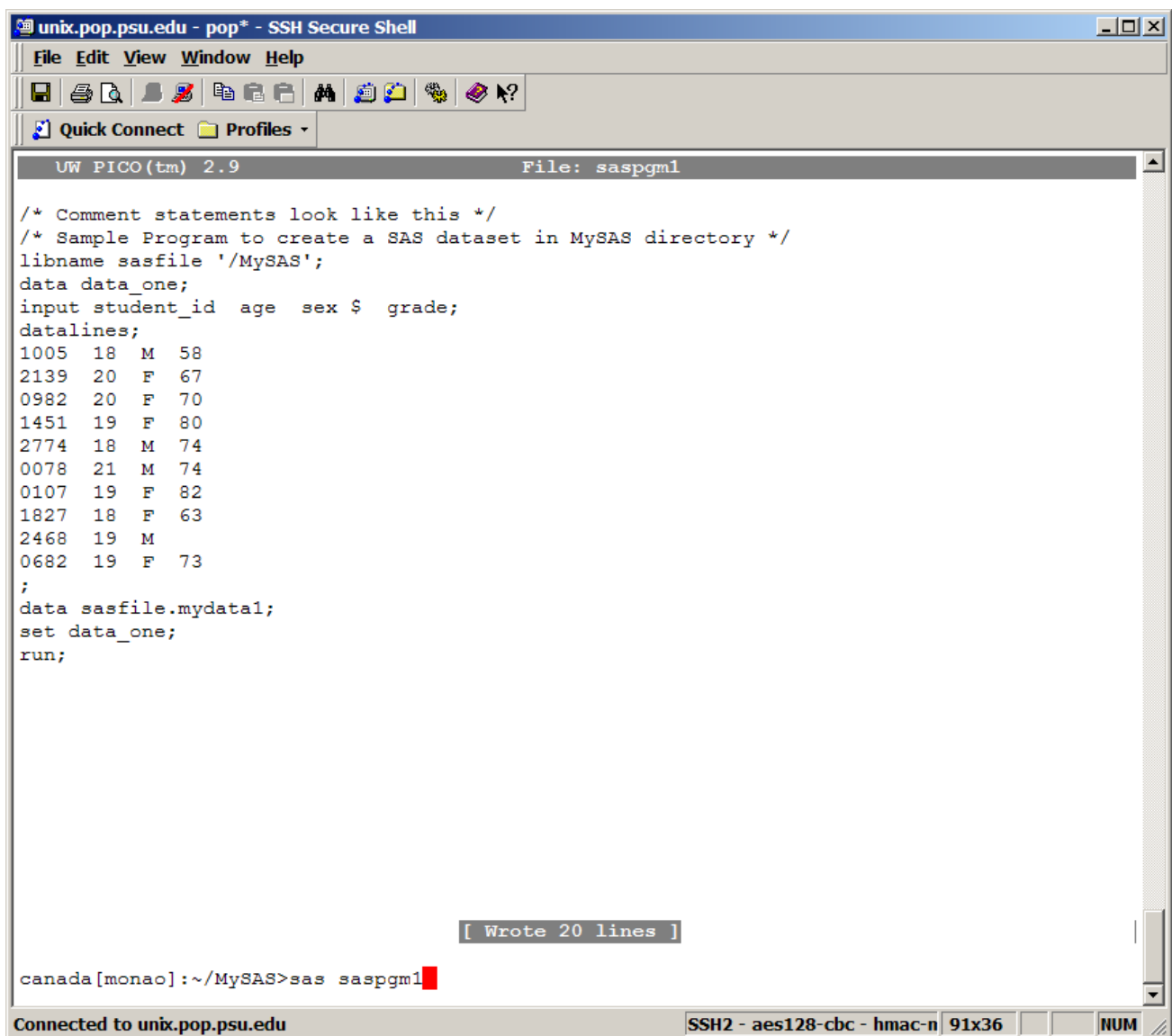
There are 2 modes in **vi**:

input (for text/data entry) and
command.

Esc (*escape key*) will put you into the command mode
while typing **i** will put you back into the input mode.

Both editors are available in the PopNet servers.

Writing a SAS program in UNIX



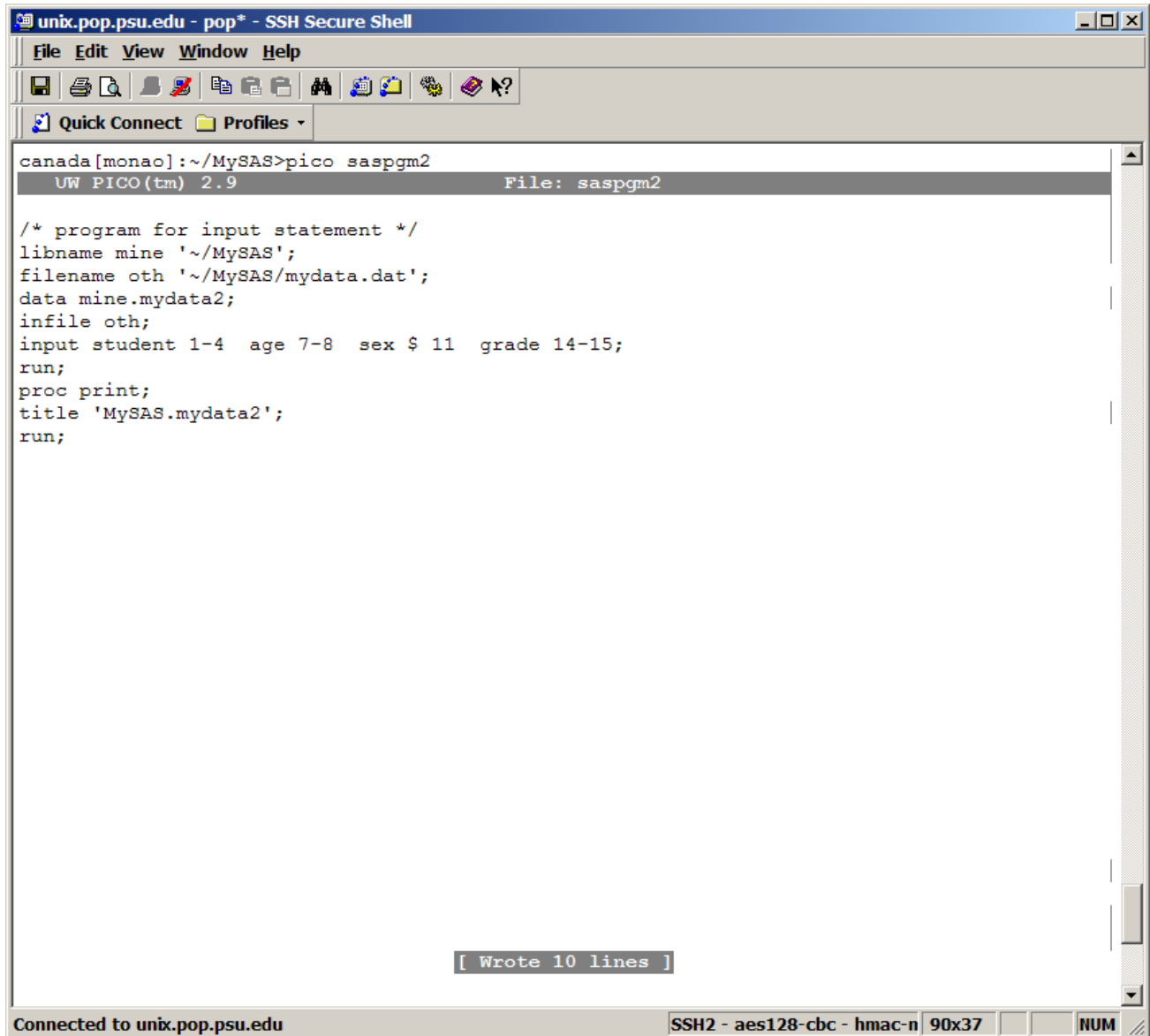
```
unix.pop.psu.edu - pop* - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
UW PICO (tm) 2.9 File: saspgm1
/* Comment statements look like this */
/* Sample Program to create a SAS dataset in MySAS directory */
libname sasfile '/MySAS';
data data_one;
input student_id age sex $ grade;
datalines;
1005 18 M 58
2139 20 F 67
0982 20 F 70
1451 19 F 80
2774 18 M 74
0078 21 M 74
0107 19 F 82
1827 18 F 63
2468 19 M
0682 19 F 73
;
data sasfile.mydata1;
set data_one;
run;

[ Wrote 20 lines ]

canada[monao]:~/MySAS>sas saspgm1
```

Connected to unix.pop.psu.edu SSH2 - aes128-cbc - hmac-n 91x36 NUM

Writing a SAS program in UNIX



The screenshot shows a terminal window titled "unix.pop.psu.edu - pop* - SSH Secure Shell". The user is in the directory ~/MySAS and has just run the command "pico sasp2gm2". The pico editor interface shows the following SAS program code:

```
/* program for input statement */
libname mine '~/MySAS';
filename oth '~/MySAS/mydata.dat';
data mine.mydata2;
infile oth;
input student 1-4 age 7-8 sex $ 11 grade 14-15;
run;
proc print;
title 'MySAS.mydata2';
run;
```

At the bottom of the editor window, a status bar indicates "[Wrote 10 lines]". The terminal window footer shows "Connected to unix.pop.psu.edu" and "SSH2 - aes128-cbc - hmac-n 90x37 NUM".

Now that a program has been created and saved as **sasp2gm2**, the program can be run.

Running a SAS program/job (batch mode)

```
[monao@italy ~]$ cd MySAS
[monao@italy ~/MySAS]$ ls
content5p      mydata2.sas7bdat  sasp2gm2      sasp2gm3.log  sasp2gm4.lst
content5p.log  mydata3.sas7bdat  sasp2gm3.lst
mydata.dat     sasp2gm1          sasp2gm4
mydata1.sas7bdat sasp2gm1.log     sasp2gm3      sasp2gm4.log
[monao@italy ~/MySAS]$ sas sasp2gm2 &
[1] 449
[monao@italy ~/MySAS]$ ps -fu monao
UID    PID  PPID  C  STIME TTY      TIME CMD
monao   381   377  0  09:28 ?        00:00:00 sshd: monao@pts/2
monao   382   381  0  09:28 pts/2    00:00:00 -tcsh
monao   505   382  0  09:36 pts/2    00:00:00 ps -fu monao
[1] + Done                  sas sasp2gm2
[monao@italy ~/MySAS]$ ls
content5p      mydata2.sas7bdat  sasp2gm2      sasp2gm3.log  sasp2gm4.lst
content5p.log  mydata3.sas7bdat  sasp2gm2.log  sasp2gm3.lst
mydata.dat     sasp2gm1          sasp2gm2.lst  sasp2gm4
mydata1.sas7bdat sasp2gm1.log     sasp2gm3      sasp2gm4.log
```

SAS Output

```
[monao@italy ~/MySAS]$ cat sas2pgm.log
cat: sas2pgm.log: No such file or directory
[monao@italy ~/MySAS]$ cat sasp2gm2.log
1
                                The SAS System
28, 2006                                09:32 Thursday, September
```

NOTE: Copyright (c) 2002-2003 by SAS Institute Inc., Cary, NC, USA.

NOTE: SAS (r) 9.1 (TS1M3)

Licensed to PSU-STATISTICS,BUSINESS,HHD,POPS DEPTS-T/R ONLY, Site 0016309002.

NOTE: This session is executing on the Linux 2.6.9-42.0.2.ELsmp platform.

NOTE: SAS 9.1.3 Service Pack

You are running SAS 9. Some SAS 8 files will be automatically converted by the V9 engine; others are incompatible. Please see <http://support.sas.com/rnd/migration/planning/platform/64bit.html>

PROC MIGRATE will preserve current SAS file attributes and is recommended for converting all your SAS libraries from any SAS 8 release to SAS 9. For details and examples, please see <http://support.sas.com/rnd/migration/index.html>

This message is contained in the SAS news file, and is presented upon

initialization. Edit the file "news" in the "misc/base" directory to display site-specific news and information in the program log. The command line option "-nonews" will prevent this display.

NOTE: SAS initialization used:
real time 0.21 seconds
cpu time 0.06 seconds

NOTE: AUTOEXEC processing beginning; file is /usr/local/SAS/SAS_9.1/autoexec.sas.

```
1 /* program for input statement */  
2 libname mine '~/MySAS';
```

NOTE: Libref MINE was successfully assigned as follows:

```
Engine: V9  
Physical Name: /home/users/monao/MySAS  
3 filename oth '~/MySAS/mydata.dat';  
4 data mine.mydata2;  
5 infile oth;  
6 input student 1-4 age 7-8 sex $ 11 grade 14-15;  
7 run;
```

NOTE: The infile OTH is:

```
File Name=/home/users/monao/MySAS/mydata.dat,  
Owner Name=monao,Group Name=ccore,  
Access Permission=rw-r--r--,  
File Size (bytes)=160
```

NOTE: 10 records were read from the infile OTH.

2
28, 2006

The SAS System

09:32 Thursday, September

The minimum record length was 15.

The maximum record length was 15.

NOTE: The data set MINE.MYDATA2 has 10 observations and 4 variables.

NOTE: DATA statement used (Total process time):

```
real time 0.12 seconds  
cpu time 0.03 seconds
```

```
8 proc print;  
9 title 'MySAS.mydata2';  
10 run;
```

NOTE: There were 10 observations read from the data set MINE.MYDATA2.

NOTE: The PROCEDURE PRINT printed page 1.

NOTE: PROCEDURE PRINT used (Total process time):

```
real time 0.09 seconds  
cpu time 0.05 seconds
```

NOTE: SAS Institute Inc., SAS Campus Drive, Cary, NC USA 27513-2414

NOTE: The SAS System used:

real time 0.46 seconds

cpu time 0.14 seconds

[monao@italy ~/MySAS]\$

SAS Output

[monao@italy ~/MySAS]\$ **cat saspgm2.lst**

MySAS.mydata2

09:32 Thursday, September 28,

2006 1

Obs	student	age	sex	grade
1	1005	18	M	58
2	2139	20	M	67
3	982	20	F	70
4	1451	19	M	80
5	2774	18	M	74
6	78	21	M	59
7	107	19	F	82
8	1828	17	F	63
9	2468	19	M	.
10	682	19	F	73

[monao@italy ~/MySAS]\$

To Logoff & Disconnect from SSH:

Type in **logout**

ex: [monao@italy ~/MySAS]\$ **logout**

PC SAS:

PC SAS can be found on the PC in Oswald's 8th floor Computer Lab (rm # 811).

Starting the program-

If there is a SAS icon on your desk top, simply double click the icon to start up the SAS program on your PC.

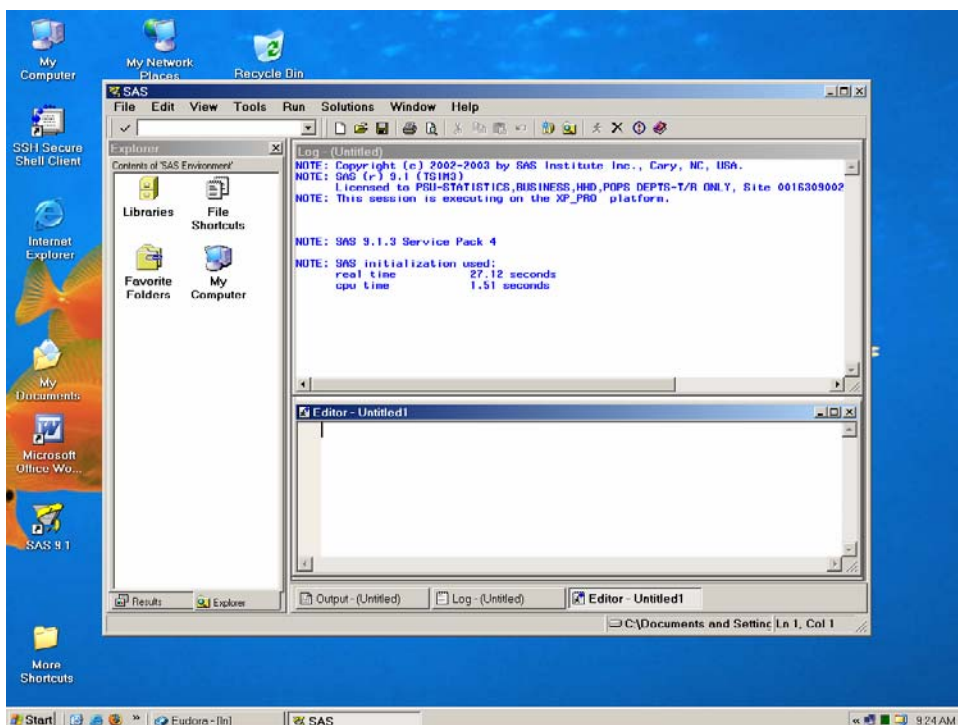
Otherwise click the **Start** button on the bottom menu. In **Programs** locate **SAS** and mouse over it so the drop down menu appears. Click on **SAS 9.1**. If the program is not found in the above locations, the SAS program might not be on the machine you are using. Contact Pop Help to see if SAS can be installed onto the machine.

Upon initiating PC SAS a window opens in which there are 3 sub windows.

On the left is the **Explorer** window which will list the libraries/directories you have assigned. You will always have a file called Work, where temporary files can be created and used. These files are only there for the SAS session – i.e. once you exit SAS they are gone completely.

On the right the top window is the **SAS Log** window, where SAS displays it's actions and comments. It is here you will find out whether your program runs correctly or gross syntax errors in your program.

On the right bottom window is the **Editor** window. It is here you can write your program code or open a sas program file to view/modify/run all or part of the code.



BASIC SAS Rules:

- SAS statements ALWAYS end with a semi-colon (;)
- Almost all SAS statements begin with a SAS keyword: **libname**, **proc**, **data**, **infile**, **input**, **output**, **set**, **if**, **else**, **merge**, **run**, etc. The exception are assignments statements: **sex='F'**;
- Quotes must match in style: **'xyz'** or **"xyz"**
- Comments are denoted by a preceding **/*** or surrounded by **/*** and ***/**
- A SAS program must end with a **RUN**;
- SAS can be free form but structure the code so it can be
 1. Easily read
 2. The logical sequence can be understood
 3. Use comments to describe what a series of code/program is doing

Data Types

Be sure you have **permission to read the data files** you intend to use

Generally non SAS data or raw data, like **.dat**, use the **filename** statement to point to the location and assign an alias.

Then to bring in variables an **input** statement is used.

SAS data, like **.sas7bdat**, can require a **libname** statement to point to the location and assign an alias. The SAS keyword, **data**, indicates you are going to creating a sas data set. a **.sas7bdat** with the statements that follow.

Library/Directory Assignment

In PC SAS, the work library is automatically assigned as your SAS session's work library.

All other library names are assigned using the SAS keyword **libname**.

Example:

1. There is a SAS dataset called **MO.sas7bdat** in a directory called **SAS_data** located in the PC's **C** drive.
2. The library name I want to assign is **mydata**.

The SAS statement is written as:

```
libname mydata 'C:/SAS_data/';  
run;
```

***note:** PC SAS colorizes the words based upon their function;
Red is bad!*

(to run this 2 statement program: in the toolbar you will see a running man figure; mouse over it and click to run the program;)

The **SAS Log** will display:

```
1  libname mydata 'C:/SAS_data/';
NOTE: Libref MYDATA was successfully assigned as follows:
      Engine:          V9
      Physical Name:  C:\SAS_data
2  run;
```

In the **SAS Explorer** double click on **Libraries** which opens the **Active Libraries** window in which you will find **MYDATA**; Double clicking **MYDATA** will open a window with the contents of the directory's SAS datasets where **MO** is listed.

Data such as .dat or .txt files can be read into a SAS data set with a program.

Example:

In this program you type in your data, which follows the **datalines** statement.

```
/* Comment statements look like this */
/* Sample Program to create a SAS dataset in MySAS directory */

libname sasfile 'C:/SAS_data/';
/* library called sasfile is created */

data data_one;
input student_id 1-4 age 7-8 sex $ 11 grade 14-15;
datalines;
1005 18 M 58
2139 20 F 67
1927 18 F 53
1857 17 F 63
2368 18 F
0982 20 F 70
1451 19 F 80
2774 18 M 74
1465 21 M
0078 21 M 74
0107 19 F 82
1827 18 F 63
2468 19 M
1774 18 M 94
2004 16 M 84
1438 17 F
0682 19 F 73
;
/* data_one is in the WORK library */
```

```
data sasfile.mydata1;
set data_one;
run;
/* SAS_data/mydata1.sas7bdat is a copy of data_one */
```

Some More SAS keywords:

data – creates a SAS data set;
set – copies the data set(s) named after keyword;
merge – combine data (2 or more) into one;
format – tell the type and appearance of the data;
retain – keeps the value of data variable(s) to the next observation;
attrib – assign labels, formats etc. to variables;
if – conditions data should satisfy;
else – when the data conditions are not satisfied;
do – statements that are done for data conditions;
end – indicates the point the do statements come to an end;
proc – a procedure that performs a task such as print, statistical analysis, etc.;
by – when data is sorted based upon variable(s), follows set or merge or proc;

Examples of SAS programs:

Extract and IF statements:

```
libname sasfile 'C:/SAS_data/';
/* library called sasfile is created */
data sasfile.female;
set data_one;
if sex='F' then output;
run;
/* C:/SAS_data/female.sas7bdat is a subset of data_one composed of the females only */
```

```
data no_score;
set data_one;

/* when missing a grade */
if grade=. then do;
  /* only want students 17 & older */
  if age lt 17 then delete;
  output;
end;
run;
/* a data set of students over 17 whose grade is missing is created in the work library called
no_score */
```

Print Procedure:

```
/* print out the students who are missing a grade */  
proc print data=no_score;  
title 'Students Missing a Grade';  
run;
```

Resulting OUTPUT window for Proc Print:

Students Missing a Grade 10:18 Thursday, September 28, 2006 4

Obs	student_ id	age	sex	grade
1	2368	18	F	.
2	1465	21	M	.
3	2468	19	M	.
4	1438	17	F	.

Frequency Procedure:

```
/* to find the frequency of the variables sex & age */  
proc freq data=data_one;  
table sex age;  
run;
```

Resulting OUTPUT window for Proc Freq:

The SAS System 10:18 Thursday, September 28, 2006 1

The FREQ Procedure

sex	Frequency	Percent	Cumulative Frequency	Cumulative Percent
F	10	58.82	10	58.82
M	7	41.18	17	100.00

age	Frequency	Percent	Cumulative Frequency	Cumulative Percent
16	1	5.88	1	5.88
17	2	11.76	3	17.65
18	6	35.29	9	52.94
19	4	23.53	13	76.47
20	2	11.76	15	88.24
21	2	11.76	17	100.00

Sort Procedure:

```
/* to sort the data called data_one by sex and age to create data_two */  
proc sort data=data_one out=data_two;  
by sex age;  
  
proc print data=data_two;  
run;
```

Resulting OUTPUT window for Proc Sort:

The SAS System 10:18 Thursday, September 28, 2006 2

Obs	student_ id	age	sex	grade
1	1857	17	F	63
2	1438	17	F	.
3	1927	18	F	53
4	2368	18	F	.
5	1827	18	F	63
6	1451	19	F	80
7	107	19	F	82
8	682	19	F	73
9	2139	20	F	67
10	982	20	F	70
11	2004	16	M	84
12	1005	18	M	58
13	2774	18	M	74
14	1774	18	M	94
15	2468	19	M	.
16	1465	21	M	.
17	78	21	M	74

Proc Means:

```
/* to find the means median maximum and minimum values for the age and grade */  
proc means data=data_one n mean median max min ;  
var age grade ;  
run;
```

Resulting OUTPUT window for Proc Means:

10:18 Thursday, September 28, 2006 8

The MEANS Procedure

Variable	N	Mean	Median	Maximum	Minimum
age	17	18.5882353	18.0000000	21.0000000	16.0000000
grade	13	71.9230769	73.0000000	94.0000000	53.0000000

```

/* to find the number means median maximum and minimum values for the age and grade */
proc means data=data_two n mean median max min ;
by sex;
var age grade ;
run;

```

Resulting OUTPUT window for Proc Means by a variable

10:18 Thursday, September 28, 2006 7

----- sex=F -----

The MEANS Procedure

Variable	N	Mean	Median	Maximum	Minimum
age	10	18.5000000	18.5000000	20.0000000	17.0000000
grade	8	68.8750000	68.5000000	82.0000000	53.0000000

----- sex=M -----

Variable	N	Mean	Median	Maximum	Minimum
age	7	18.7142857	18.0000000	21.0000000	16.0000000
grade	5	76.8000000	74.0000000	94.0000000	58.0000000

Using Labels and Formats:

```

/* applying labels to data & formatting the data */
data class;

set data_two;
format grade 4.2;
attrib student_id label='Student Id'
        grade label='Test One';
proc print;
var student_id grade;
run;

```

Resulting OUTPUT window

10:18 Thursday, September 28, 2006 15

Obs	student_id	grade
1	1857	63.0
2	1438	.
3	1927	53.0
4	2368	.
5	1827	63.0
6	1451	80.0

7	107	82.0
8	682	73.0
9	2139	67.0
10	982	70.0
11	2004	84.0
12	1005	58.0
13	2774	74.0
14	1774	94.0
15	2468	.
16	1465	.
17	78	74.0

Questions???

I am happy to answer questions on SAS.

If you need extra help please contact me. I can coach you individually.

Stop by my office, call or email.

Helpful Links

UNIX:

<http://www.ee.surrey.ac.uk/Teaching/Unix/>

<http://webmonkey.wired.com/webmonkey/html/97/02/index3a.html>

<http://www.sims.berkeley.edu/~kevin/unix-tutorial/>

<http://www.its.caltech.edu/its/facilities/labsclusters/unix/unixtutorial.shtml>

SAS:

PRI SAS Help Webpage:

<http://www.pop.psu.edu/help/cacpri/index.htm>

SAS Tutorial Links:

http://www.columbia.edu/acis/eds/stat_pak/sas/sas-write.html

http://cirt.unm.edu/introductions/Sas_tutorial/

PC:

http://www-unix.oit.umass.edu/~statdata/software/handouts/sas_online/

<http://www.indiana.edu/~statmath/stat/sas/win/>

SAS Documentation:

<http://v8doc.sas.com/sashtml/>

Samples of SAS Programs:

http://support.sas.com/techsup/sample/sample_library.html

PICO:

<http://www.indiana.edu/~uitspubs/b103/>

http://www.ncsu.edu/it/essentials/managing_files/text_editors/pico_tutor/

VI:

http://math.la.asu.edu/vi_tutorial/vicontents.html

<http://www.unix-manuals.com/tutorials/vi/vi-in-10-1.html>