

Bash Scripting

It is often handy to create a script if one has to do a task a number of times. This ensures that the task is done the same way every time and reduces repetitive typing.

An example is a script I use at the beginning of the semester to generate a clean list of student names and their A number. I often need this when generating quizzes or exams.

The Pima page where this information is listed also has a lot of other “garbage” that I don’t need such as when registered, class hours, etc.

So I spent some time developing a simple bash script using the stream editor SED to strip out all of the extraneous material and just output a file with the First and Last names and A numbers of students.

I first just “select all” on the Pima page and paste to a file. Then I invoke the script and run it against that filename and it outputs the cleaned up text in a second filename that I specify.

An example of how this is run would be `./sedscrip.sh CIS225 Students225`
This takes an input file of CIS225 and outputs a file called Student225

Here is what the script looks like:

```
#!/bin/bash
```

```
filename1=$1
```

```
filename2=$2
```

```
sed -i 's/,/ /g' $filename1
```

```
sed -i 's/\. /g' $filename1
```

```
sed -i 's/**Web Registered**/ /g' $filename1
```

```
sed -i 's/**Registered**/ /g' $filename1
```

```
sed -i 's/Confidential/ /g' $filename1
```

```
sed -i 's/Credit/ /g' $filename1
```

```
sed -i 's/4.000/ /g' $filename1
```

```
sed -i 's/E-mail/ /g' $filename1
```

```
awk '{print $4" "$3" "$5" "$6" "$7" "$8}' $filename1 > $filename2
```

```
# Usage, sedscrip filename1 filename2
```

```
# Outputs cleaned up file consisting of first and last name,
```

```
# initial if present and A number.
```

Microsoft has recognized how useful this is and has introduced a scripting language in their Windows OS's called powershell. It consists of many cmdlets that allow you to do many things. They also took the time to alias most of the Windows command prompts and the Linux command prompts to these cmdlets. This allows a Linux admin or a Windows admin familiar with the command prompt to generate useful scripts without a long learning curve.