

# Learn how to use the csv kit

## Resources

[link to tutorial page](#)

[link to references and full manual for functions](#)

## 1.2 installing

```
sudo pip install csvkit
```

This worked, but also recomend using virtualenv to keep the python packages in better order.

## 1.3 Getting the data

```
mkdir csvkit_tutorial cd csvkit_tutorial
```

```
curl -L -O https://raw.githubusercontent.com/wireservice/csvkit/master/examples/realdata/ne_1033_data.xls
```

## 1.4 in2csv the excel killer

*transform to csv and show in terminal*

```
in2csv ne_1033_data.xlsx
```

*transform to csv and save as data.csv*

```
in2csv ne_1033_data.xlsx > data.csv
```

## 1.5 csvlook: dataperiscope

*to get the rough idea of the data use csvlook*

```
csvlook data.csv
```

*to get it paged, separated in columns and able to navigate through, use less*

```
csvlook data.csv | less -S
```

## 1.6 csvcut: datascalpel

*to select and reorder columns*

\* to display the numbers and names of the columns use -n flag\*

```
csvcut -n data.csv
```

*to display just columns of given numbers use -c flag*

```
csvcut -c 2,5,6 data.csv
```

*to see the selected columns in the nice less -S format combine with csvlook*

```
csvcut -c 2,5,6 data.csv | csvlook | less -S
```

*csvcut also understands the names of the columns as long as there are no spaces after commas*

*the commas are tricky, if in the name double quote (even in beginning in the column name)*

```
csvcut -c county,item_name,quantity data.csv
```

## 1.7 connecting commands through pipes

easy, not much to add here

## 2. Examining the data

### 2.1 csvstat: statistics without code

*inspired by the summary from, pick columns and and pipe to stat*

```
csvcut -c county,acquisition_cost,ship_date data.csv | csvstat
```

### csvgrep: find the data you need

*csvgrep with -c flag sets columns where to search, -m flag what to search*

```
csvcut -c county,item_name,total_cost data.csv | csvgrep -c county -m LAN-CASTER | csvlook
```

*the -r flag gives you the chance to go for regular expressions“*

*in case of -r flag the pattern needs to be doublequoted*

*case insensitivity is achieved by“(i)pattern”*

```
csvgrep -c 2 -r “(i)lancaster” data.csv | csvcut -c 1,2,5 | csvlook | less -S
```

*The insensitivity flag applies to all the expressions afterwards*

```
csvgrep -c 5 -r “(i)ri|li” data.csv | csvcut -c 1,2,5 | csvlook | less -S
```