Bash <3's CSVs
Command line data analysis

Nick Canzoneri
csv,conf 2019

Who am I?

- Database infrastructure @GitHub

- github.com/nickcanz

- twitter.com/nick_canz

- nickcanzoneri.com

Who am I?

Bash isn't great for *everything*

@nick_canz

Why use bash?

"Command-line Tools can be 235x Faster than your Hadoop Cluster"
https://adamdrake.com/command-line-tools-can-be-235x-faster-than-your-hadoop-cluster.html

> Since the data volume was **only about 1.75GB** containing around 2 million chess games,
>
> for the same amount of data I was able to use my laptop to get the results in about **12 seconds** while the Hadoop processing took about **26 minutes**

@nick_canz

When is Bash a good choice?

➡️ `New York city,New York,8537673`
➡️ `Los Angeles city,California,3976322`
➡️ `Chicago city,Illinois,2704958`

When is Bash a good choice?

```
New York city,New York,8537673
Los Angeles city,California,3976322
Chicago city,Illinois,2704958
```

First step: peeking at the data

```
$ head -n2 populations.csv
New York city,New York,8537673
Los Angeles city,California,3976322
```

…sort of like

```
SELECT * FROM populations
LIMIT 2
```

First step: peeking at the data

```
$ tail -n2 populations.csv
Saginaw city,Michigan,48984
Niagara Falls city,New York,48632
```

…sort of like

```
SELECT * FROM populations
ORDER BY DESC
LIMIT 2
```

First step: peeking at the data

```
$ wc -l populations.csv
    761 populations.csv
```

...sort of like

```
SELECT count(*)
FROM populations
```

First step: peeking at the data

```
$ wc -l populations.csv          Count lines
    761 populations.csv


$ wc -m populations.csv          Count characters
  23643 populations.csv


$ wc -w populations.csv          Count words
   1849 populations.csv
```

@nick_canz

First step: peeking at the data

```
$ head -n2 populations.csv
New York city,New York,8537673
Los Angeles city,California,3976322

$ tail -n2 populations.csv
Saginaw city,Michigan,48984
Niagara Falls city,New York,48632

$ wc -l populations.csv
     761 populations.csv
```

@nick_canz

Getting the lines you want

```
$ grep Pennsylvania populations.csv
Philadelphia city,Pennsylvania,1567872
Pittsburgh city,Pennsylvania,303625
Allentown city,Pennsylvania,120443
```

…sort of like

```
SELECT * FROM populations
WHERE row like 'Pennsylvania'
```

@nick_canz

Excluding the lines you don't want

```
$ grep -v city populations.csv
Urban Honolulu CDP,Hawaii,351792
Lexington-Fayette urban county,Kentucky,318449
Anchorage municipality,Alaska,298192
```

...sort of like

```
SELECT * FROM populations
WHERE row not like 'city'
```

@nick_canz

Get the fields you want

```
New York city,New York,8537673
Los Angeles city,California,3976322

$ cut -d',' -f1,3 populations.csv
New York city,8537673
Los Angeles city,3976322
```

Get the fields you want

```
[17/May/2015:08:05:32 +0000] "GET /downloads/product_1 HTTP/1.1" 304

$ cut -d' ' -f1 nginx.log

[17/May/2015:08:05:32

$ cut -d' ' -f1 nginx.log | cut -d':' -f2-

08:05:32
```

@nick_canz

Organizing your data

```
$ sort populations.csv | head -n 5

Abilene city,Texas,122225
Akron city,Ohio,197633
Alameda city,California,78906
Albany city,Georgia,73801
Albany city,New York,98111
```

@nick_canz

Organizing your data

```
$ sort -t',' -k2,2 populations.csv

Auburn city,Alabama,63118
Birmingham city,Alabama,212157
Decatur city,Alabama,55072
Dothan city,Alabama,68468
Hoover city,Alabama,84978
```

Organizing your data

```
$ sort -t',' -k2,2 -k3,3nr populations.csv

Birmingham city,Alabama,212157
Montgomery city,Alabama,200022
Huntsville city,Alabama,193079
Mobile city,Alabama,192904
Tuscaloosa city,Alabama,99543
```

What are the biggest cities by state?

```
$ sort -t',' -k2,2 -k3,3nr populations.csv |
  sort -u -t',' -k2,2
```

```
Birmingham city,Alabama,212157
Anchorage municipality,Alaska,298192
Phoenix city,Arizona,1615017
Little Rock city,Arkansas,198541
Los Angeles city,California,3976322
Denver city,Colorado,693060
```

How many cities per state?

```
$ cut -d',' -f2 populations.csv |   Get only the state field
  sort |          Sort it
  uniq -c |       Get the unique values and count
  sort -nr        Sort by number descending
```

```
178 California
 65 Texas
 57 Florida
 29 Illinois
 25 Michigan
```

@nick_canz

Where does Portland rank?

```
$ sort -t',' -k3,3nr populations.csv |
  grep -n Portland
```
Find Portland and it's number in the list

```
26:Portland city,Oregon,639863
537:Portland city,Maine,66937
```

@nick_canz

Working with multiple files

```
$ cat fips_and_city.csv
06,Los Angeles
06,San Francisco
12,Miami
12,Orlando
42,Philadelphia
42,Pittsburgh
```

```
$ cat fips_and_state.csv
06,California
12,Florida
42,Pennsylvania
```

@nick_canz

Working with multiple files

```
$ cat fips_and_city.csv
06,Los Angeles
06,San Francisco
12,Miami
12,Orlando
42,Philadelphia
42,Pittsburgh
```

```
$ cat fips_and_state.csv
06,California
12,Florida
42,Pennsylvania
```

```
$ join -t',' fips_and_city.csv fips_and_state.csv
06,Los Angeles,California
06,San Francisco,California
12,Miami,Florida
12,Orlando,Florida
42,Philadelphia,Pennsylvania
42,Pittsburgh,Pennsylvania
```

@nick_canz

Working with multiple files

```
$ cat cities.csv
Philadelphia
New York
Austin
```

```
$ cat foods.csv
Cheesesteak
Pizza
Brisket
```

```
$ paste -d',' cities.csv foods.csv
Philadelphia,Cheesesteak
New York,Pizza
Austin,Brisket
```

```
$ paste -s -d',' cities.csv foods.csv
Philadelphia,New York,Austin
Cheesesteak,Pizza,Brisket
```

@nick_canz

Working with multiple files

```
$ seq 1 5
1
2
3
4
5


$ paste -d, <(seq 1 5) <(head -n5 populations.csv)
1,New York city,New York,8537673
2,Los Angeles city,California,3976322
3,Chicago city,Illinois,2704958
4,Houston city,Texas,2303482
5,Phoenix city,Arizona,1615017
```

Comparing files

```
$ comm ohio_cities.csv missouri_cities.csv
Akron city
Cincinnati city
Cleveland city
Columbus city
  Joplin city
  Kansas City city
  O'Fallon city
    Springfield city
St. Louis city
```
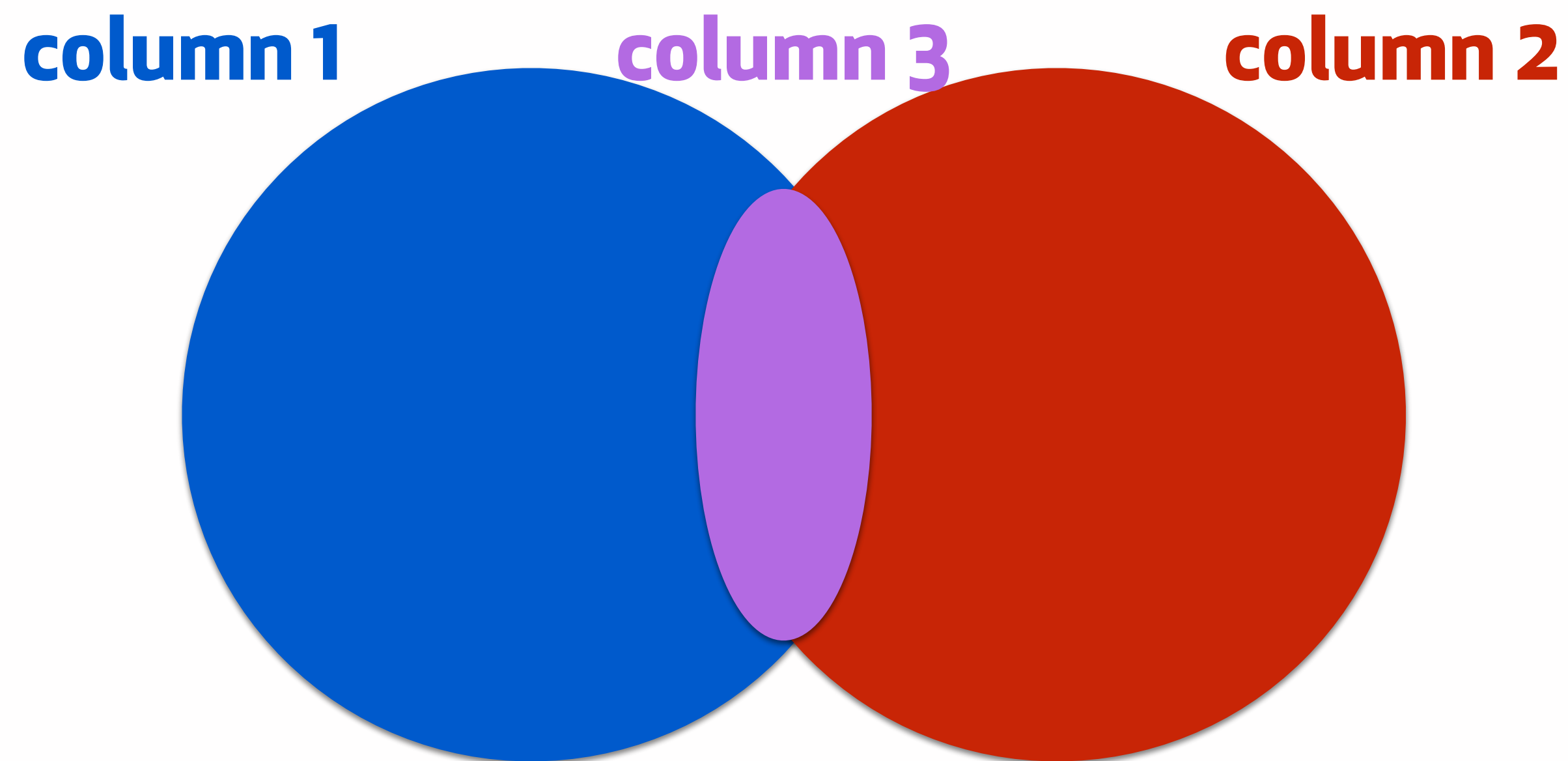
column 1   **column 3**   **column 2**

@nick_canz
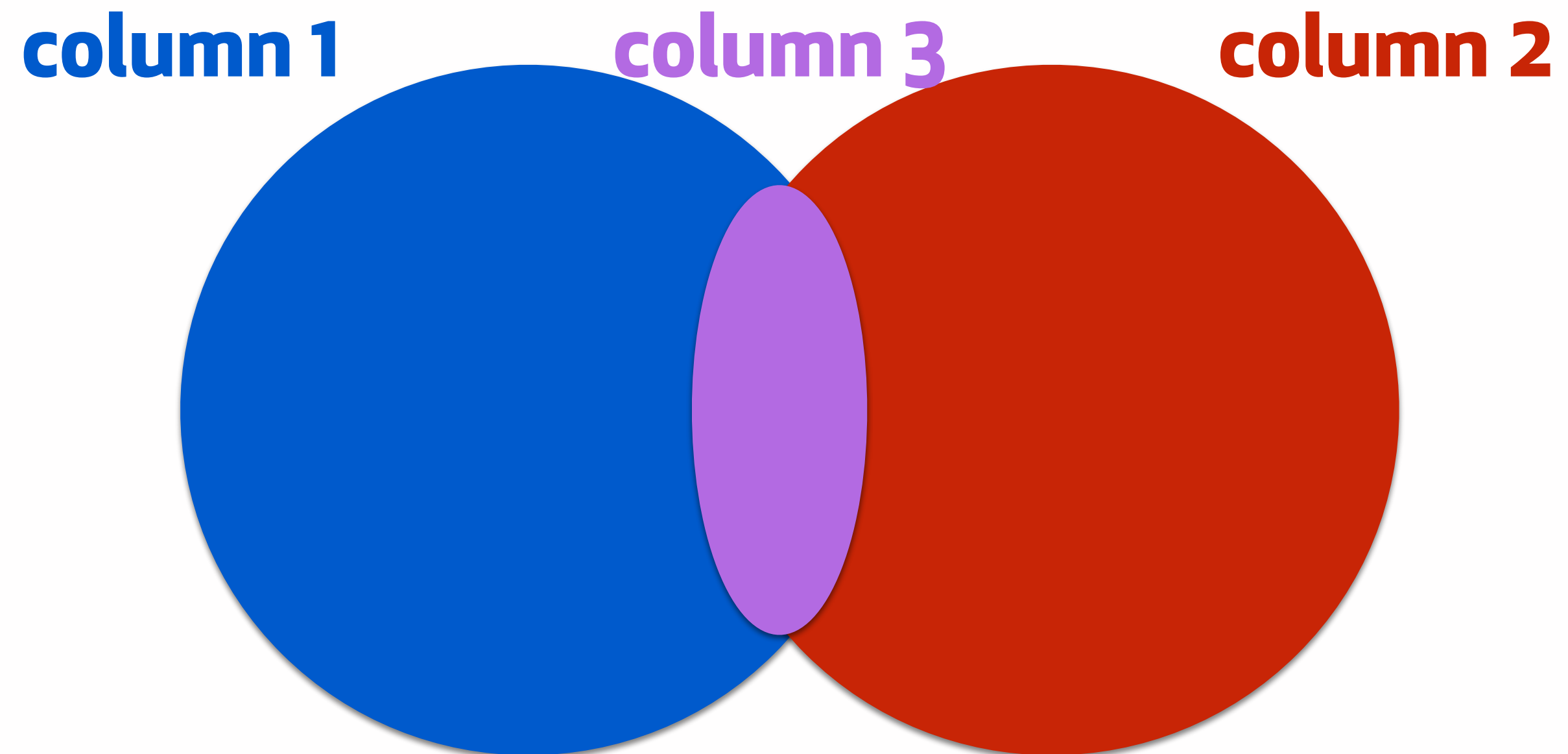
Comparing files

```
$ comm -23 ohio_cities.csv missouri_cities.csv
```

```
$ comm -13 ohio_cities.csv missouri_cities.csv
```
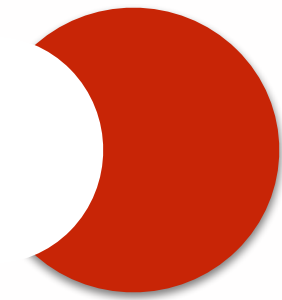
```
$ comm -12 ohio_cities.csv missouri_cities.csv
```

column 1   column 3   column 2

@nick_canz

Dealing with cleanup

```
Brookhaven city,Georgia,52444
Minnetonka city,Minnesota,52369
Palm Desert city,California,52231

$ sed 's/ city//' populations.csv

Brookhaven,Georgia,52444
Minnetonka,Minnesota,52369
Palm Desert,California,52231
```

@nick_canz

Dealing with cleanup

```
$ echo 'A B C' | tr ' ' '\n'
A
B
C

$ echo 'APPLE PEAR' | tr '[:upper:]' '[:lower:]'
apple pear

$ echo '0 1 2' | tr '[0-8]' '[1-9]'
1 2 3
```

Practical example time

A nice dataset

```
$ head populations.csv
New York city,New York,8537673
Los Angeles city,California,3976322
Chicago city,Illinois,2704958
Houston city,Texas,2303482
Phoenix city,Arizona,1615017
Philadelphia city,Pennsylvania,1567872
San Antonio city,Texas,1492510
San Diego city,California,1406630
Dallas city,Texas,1317929
San Jose city,California,1025350
```

Where to get the data?

A not nice dataset

```
$ head PEP_2016_PEPANNRSIP.US12A_with_ann.csv
GEO.id,GEO.id2,GEO.display-label,GC_RANK.target-geo-id,GC_RANK.target-geo-id2,GC_RANK.rank-label,GC_RANK
label,rescensus42010,resbase42010,respop72010,respop72011,respop72012,respop72013,respop72014,respop7201
Id,Id2,Geography,Target Geo Id,Target Geo Id2,Rank,Geography,Geography,"April 1, 2010 - Census","April 1
2011,Population Estimate (as of July 1) - 2012,Population Estimate (as of July 1) - 2013,Population Esti
0100000US,,United States,1620000US3651000,3651000,1,"United States - New York city, New York","New York
0100000US,,United States,1620000US0644000,0644000,2,"United States - Los Angeles city, California","Los
0100000US,,United States,1620000US1714000,1714000,3,"United States - Chicago city, Illinois","Chicago ci
```

Making our nice dataset

```
$ cat PEP_2016_PEPANNRSIP.US12A_with_ann.csv |
      tail -n +3   Remove the two(?) header lines
```

```
0100000US,,United States,1620000US3651000,3651000,1,"United States - New York city, New York","Ne
0100000US,,United States,1620000US0644000,0644000,2,"United States - Los Angeles city, California
0100000US,,United States,1620000US1714000,1714000,3,"United States - Chicago city, Illinois","Chi
0100000US,,United States,1620000US4835000,4835000,4,"United States - Houston city, Texas","Housto
```

Making our nice dataset

```
$ cat PEP_2016_PEPANNRSIP.US12A_with_ann.csv |
    tail -n +3 |
    cut -d',' -f9,10,19    Get the fields we want


"New York city, New York",8537673
"Los Angeles city, California",3976322
"Chicago city, Illinois",2704958
"Houston city, Texas",2303482
"Phoenix city, Arizona",1615017
```

Making our nice dataset

```
$ cat PEP_2016_PEPANNRSIP.US12A_with_ann.csv |
    tail -n +3 |
    cut -d',' -f9,10,19 |
    sed 's/"//g'    Clean up the quotes


New York city, New York,8537673
Los Angeles city, California,3976322
Chicago city, Illinois,2704958
Houston city, Texas,2303482
Phoenix city, Arizona,1615017
```

@nick_canz

Making our nice dataset

```
$ cat PEP_2016_PEPANNRSIP.US12A_with_ann.csv |
    tail -n +3 |
    cut -d',' -f9,10,19 |
    sed 's/"//g' |
    sed 's/, /,/'
```
Remove leading space in state column

```
New York city,New York,8537673
Los Angeles city,California,3976322
Chicago city,Illinois,2704958
Houston city,Texas,2303482
Phoenix city,Arizona,1615017
```

Making our nice dataset

```
$ cat PEP_2016_PEPANNRSIP.US12A_with_ann.csv |
    tail -n +3 |
    cut -d',' -f9,10,19 |
    sed 's/"//g' |
    sed 's/, //' > populations.csv
```

Save it all to a file!

@nick_canz

Shakespeare word count

```
$ cat shakespeare.txt |
tr -d -c '[:alpha:][:space:]' |
tr -s '[:space:]' ' ' |
tr ' ' '\n' |
tr '[:upper:]' '[:lower:]' |
sort | uniq -c |
sort -nr
```

```
27825 the
26791 and
20681 i
19261 to
18289 of
14668 a
13716 you
12481 my
11135 that
11027 in
9621 is
8745 not
8261 for
8046 with
7769 me
7703 it
7106 be
```

Thanks!

Talk materials
github.com/nickcanz/csvconf2019

@nick_canz on twitter