Lecture 3: Relational databases

collection of tables + DBMS

1 "personal" databases vs commercial databases

user says what to do, and how

intelligence in the code

structure query language, SQL

2 obtaining the code

git { tutorialspoint/git
revision control system

configure
make
make install

git clone https://github.com/dacase/rdbstat.git
cd src & make install → compiles the code
put into your path

3 Tables

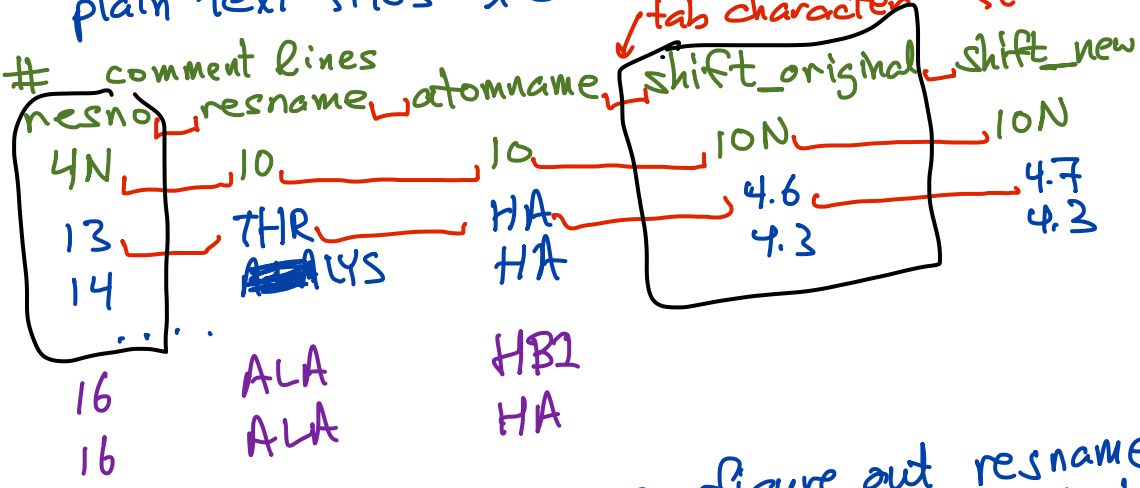
plain text files : each line is a row after the header

field, or column headers →

format →

rows

tab character \t



if you know resno, you can figure out resname

redundant, and could become inconsistent

4 Operators

input tables on stdin
output tables on stdout

column < table1.rdb | headchg -del | pair > summary.txt
↳ compares two sets of number

row < table1.rdb atomname eq "HA" | headchg -del | wc
tells me the number of HA shifts

compute < table1.rdb resno = resno + 5 > new_sequence_no.rdb
beware of special characters '(')
shift_new = shift_new '*' 0.985

sorttbl
sorts the row

jointbl
two tables as input
one table as output

5 Simple statistics

table2.rdb keys ... shift_jan26 ...

jointbl < table1.rdb resno atomname table2.rdb > result.rdb
beware! both tables must be sorted on keys
just rows where the keys match

more useful example

seq.rdb
resno resname

shifts.rdb
resno atomname shift

jointbl < seq.rdb resno shifts.rdb > prettier.rdb
contains resnames

key employee_id
name_table
id firstname lastname address

making the "best" of tables for complex data "normalization"

pay_rate table
id payrate

6 Some examples