This exercise loads a table from a scv file where:

1. the order of the columns in the input file does not match the order of the columns in the database
2. the input file has columns are are not in the database
3. the database has columns that are not in the input file
4. some transformation of data is needed

root@localhost [study\_db] desc load\_data;

+-------+---------------------+------+-----+---------+----------------+

| Field | Type | Null | Key | Default | Extra |

+-------+---------------------+------+-----+---------+----------------+

| id | bigint(20) unsigned | NO | PRI | NULL | auto\_increment |

| i | int(11) | YES | | NULL | |

| d | date | YES | | NULL | |

| t1 | varchar(16) | YES | | NULL | |

| t2 | varchar(16) | YES | | NULL | |

| t1\_t2 | varchar(32) | YES | | NULL | |

+-------+---------------------+------+-----+---------+----------------+

load\_data.csv

-------------

t3,t2,t1,d,i

bmw,sanders,tim,2018-05-01,1

honda,carlin,peter,2018-05-02,2

mercedes,worthy,gail,2018-05-03,3

audi,worthy,allan,2018-05-04,4kia,riordan,ryan,2018-05-04,5

handling file column name database column name sample data notes

--------------------------------------------------------------------------------------

skip t3 N/A bmw this column is not in the database and can be ignored

t2 t2 sanders

t1 t1 tim

d d 2018-05-01

i i 1

concat t2 and t1 N/A t1\_t2 sanders, tim this column will be derived from other columns.

LOAD DATA STATEMENT

-------------------

load data local infile 'load\_data.csv'

into table load\_data

fields terminated by ','[[1]](#footnote-1)

lines terminated by '\r\n'[[2]](#footnote-2)

IGNORE 1 LINES[[3]](#footnote-3)

(@skip, @t2, @t1, @d, @i)[[4]](#footnote-4)

SET i = @i,

d = @d,

t1 = @t1,

t2 = @t2,

t1\_t2 = concat(t1, ', ', t2);[[5]](#footnote-5)

root@localhost [study\_db] select \* from load\_data;

+----+------+------------+-------+---------+---------------+

| id | i | d | t1 | t2 | t1\_t2 |

+----+------+------------+-------+---------+---------------+

| 1 | 1 | 2018-05-01 | tim | sanders | tim, sanders |

| 2 | 2 | 2018-05-02 | peter | carlin | peter, carlin |

| 3 | 3 | 2018-05-03 | gail | worthy | gail, worthy |

| 4 | 4 | 2018-05-04 | allan | worthy | allan, worthy |

| 5 | 5 | 2018-05-04 | ryan | riordan | ryan, riordan |

+----+------+------------+-------+---------+---------------+

5 rows in set (0.00 sec)

This also works:

load data local infile 'load\_data.csv'

into table load\_data

fields terminated by ','

lines terminated by '\r\n'

IGNORE 1 LINES

(@skip, t2, t1, d, i)

SET t1\_t2 = concat(t1, ', ', t2);

1. This was required with the file I used. [↑](#footnote-ref-1)
2. This was not required, but it is probably a good best practice to include it anyway. [↑](#footnote-ref-2)
3. Ignore the header (shaded in yellow). [↑](#footnote-ref-3)
4. The order of the fields here should be the order in which they appear in the input file. To skip over fields, use a local variable such as @skip and do nothing with it; it will then be ignored. Use SET to define how to derive one column from others. [↑](#footnote-ref-4)
5. Derive value of one field from others via SET. [↑](#footnote-ref-5)