CREATE SCHEMA accidents;

USE accidents;

/\* -------------------------------- \*/

/\* Create Tables \*/

CREATE TABLE accident(

accident\_index VARCHAR(13),

accident\_severity INT

);

CREATE TABLE vehicles(

accident\_index VARCHAR(13),

vehicle\_type VARCHAR(50)

);

/\* First: for vehicle types, create new csv by extracting data from Vehicle Type sheet from Road-Accident-Safety-Data-Guide.xls \*/

CREATE TABLE vehicle\_types(

vehicle\_code INT,

vehicle\_type VARCHAR(10)

);

/\* -------------------------------- \*/

/\* Load Data \*/

LOAD DATA LOCAL INFILE 'C:\\Users\\Accidents\_2015.csv'

INTO TABLE accident

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(@col1, @dummy, @dummy, @dummy, @dummy, @dummy, @col2, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy)

SET accident\_index=@col1, accident\_severity=@col2;

LOAD DATA LOCAL INFILE 'C:\\Users\\Vehicles\_2015.csv'

INTO TABLE vehicles

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

IGNORE 1 LINES

(@col1, @dummy, @col2, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy, @dummy)

SET accident\_index=@col1, vehicle\_type=@col2;

LOAD DATA LOCAL INFILE 'C:\\Users\\vehicle\_types.csv'

INTO TABLE vehicle\_types

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

IGNORE 1 LINES;

/\* -------------------------------- \*/

CREATE TABLE accidents\_median(

vehicle\_types VARCHAR(100),

severity INT

);

# UK Road Safty 2015 data

import pymysql

myConnection = pymysql.connect(

host="localhost", user="root", password="root", db="accidents")

cur = myConnection.cursor()

cur.execute(

"SELECT vehicle\_type FROM vehicle\_types WHERE vehicle\_type LIKE '%torcycle%';")

cycle\_list = cur.fetchall()

selectSQL = ('''

SELECT vt.vehicle\_type, a.accident\_severity

FROM accident a

JOIN vehicles v ON a.accident\_index = v.accident\_index

JOIN vehicle\_types vt ON v.vehicle\_type = vt.vehicle\_code

WHERE vt.vehicle\_type LIKE %s

ORDER BY a.accident\_severity;

''')

insert\_SQL = ('''INSERT INTO accidents\_median

VALUES(%s, %s);''')

for cycle in cycle\_list:

cur.execute(selectSQL, cycle[0])

accidents = cur.fetchall()

# calculate median severity

# divide the length of accidents /2 to find the median of accdients list

quotient, remainder = divmod(len(accidents), 2)

if remainder:

# meaning odds number of items in accidents list

median\_severity = accidents[quotient][1]

else:

# even numbers of items in accidents list

median\_severity = (accidents[quotient]

[1] + accidents[quotient + 2][1]) / 2

print("finding Median Severity for ", cycle[0])

# insert the calculated median severity into table

cur.execute(insert\_SQL, (cycle[0], median\_severity))

myConnection.commit()

myConnection.close

/\* SIDE NOTE: Compare performance of the query rows by using Explain Icon first (Before Indexing and After Indexing)\*/

/\* Create index on accident\_index as it is using in both vehicles and accident tables and join clauses using indexes will perform faster \*/

CREATE INDEX accident\_index

ON accident(accident\_index);

CREATE INDEX accident\_index

ON vehicles(accident\_index);

/\* get Accident Severity and Total Accidents per Vehicle Type \*/

SELECT vt.vehicle\_type AS 'Vehicle Type', a.accident\_severity AS 'Severity', COUNT(vt.vehicle\_type) AS 'Number of Accidents'

FROM accident a

JOIN vehicles v ON a.accident\_index = v.accident\_index

JOIN vehicle\_types vt ON v.vehicle\_type = vt.vehicle\_code

GROUP BY 1

ORDER BY 2,3;

/\* Average Severity by vehicle type \*/

SELECT vt.vehicle\_type AS 'Vehicle Type', AVG(a.accident\_severity) AS 'Average Severity', COUNT(vt.vehicle\_type) AS 'Number of Accidents'

FROM accident a

JOIN vehicles v ON a.accident\_index = v.accident\_index

JOIN vehicle\_types vt ON v.vehicle\_type = vt.vehicle\_code

GROUP BY 1

ORDER BY 2,3;

/\* Average Severity and Total Accidents by Motorcyle \*/

SELECT vt.vehicle\_type AS 'Vehicle Type', AVG(a.accident\_severity) AS 'Average Severity', COUNT(vt.vehicle\_type) AS 'Number of Accidents'

FROM accident a

JOIN vehicles v ON a.accident\_index = v.accident\_index

JOIN vehicle\_types vt ON v.vehicle\_type = vt.vehicle\_code

WHERE vt.vehicle\_type LIKE '%otorcycle%'

GROUP BY 1

ORDER BY 2,3;