

Create table YELP_BUSINESS:

```
CREATE TABLE `sakila`.`yelp_business` (  
  `business_id` VARCHAR(22) NOT NULL,  
  `name` VARCHAR(255) NOT NULL,  
  `neighborhood` VARCHAR(255) DEFAULT "",  
  `address` VARCHAR(255) DEFAULT "",  
  `city` VARCHAR(255) NULL,  
  `state` VARCHAR(100) DEFAULT "",  
  `postal_code` INT DEFAULT 0,  
  `latitude` DECIMAL(12,9) DEFAULT 0.0,  
  `longitude` DECIMAL(12,9) DEFAULT 0.0,  
  `stars` FLOAT(2,1) DEFAULT 0.0,  
  `review_count` INT DEFAULT 0,  
  `is_open` TINYINT(1) DEFAULT 0,  
  PRIMARY KEY (`business_id`));
```

LOAD DATA to YELP_BUSINESS: (excluding 'category' column from original data table)

```
import _mysql  
import csv  
  
filename = 'yelp_business.csv';  
with open(filename, 'r', encoding="utf8") as f:  
    reader = csv.reader(f)  
    your_list = list(reader)  
  
for i in range(0, len(your_list)):  
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")  
    try:  
        #print(len(your_list[0]))  
        #postal_code = int(your_list[i][6])  
        line = "INSERT INTO yelp_business VALUES ( '%s' ,%s, '%s', '%s', '%s', '%s',  
'%s', %s, %s, %s, %s, %s );" % \  
            (your_list[i][0],your_list[i][1], your_list[i][2], your_list[i][3],\  
             your_list[i][4], your_list[i][5], your_list[i][6],  
             your_list[i][7], your_list[i][8], your_list[i][9],\  
             your_list[i][10], your_list[i][11])  
        db.query(line)  
        db.store_result()  
    except Exception as e:
```

```
print(e)
continue
```

YELP_CHECKIN:

```
create table yelp_checkin(
business_id VARCHAR(22) not null,
weekday VARCHAR(10) DEFAULT '',
hour DECIMAL(4,2) DEFAULT 00.00,
checkins INT DEFAULT 0,
constraint fk_checkin_1 foreign key(business_id) references
yelp_business(business_id)
);
```

LOAD DATA TO YELP_CHECKIN:

```
import _mysql
import csv

filename = 'yelp_checkin.csv';
with open(filename, 'r', encoding="utf8") as f:
    reader = csv.reader(f)
    your_list = list(reader)

for i in range(0, len(your_list)):
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")
    try:
        line = "INSERT INTO yelp_checkin VALUES ( '%s' ,'%s',%s, %s);" %
        (your_list[i][0],your_list[i][1],\
        your_list[i][2]), your_list[i][3])
        db.query(line)
        db.store_result()
    except Exception as e:
        print(e)
        continue
```

```
create table yelp_review(
review_id VARCHAR(22) primary key,
user_id VARCHAR(22) not null,
business_id VARCHAR(22) not null,
stars FLOAT(2,1),
date DATE,
text VARCHAR(5000),
useful INT,
funny INT,
cool INT,
constraint fk_review_01 foreign key(user_id) references yelp_user(user_id),
constraint fk_review_02 foreign key(businesss_id) references
yelp_business(business_id)
```

```
import _mysql
import csv

filename = 'yelp_review.csv';
with open(filename, 'r', encoding="utf8") as f:
```

```

reader = csv.reader(f)
your_list = list(reader)

for i in range(0, len(your_list)):
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")
    try:
        line = "INSERT INTO yelp_review VALUES ( '%s' , '%s', '%s', %s, %s, '%s', %s, %s, %s);" % (your_list[i][0],\
            your_list[i][1], your_list[i][2]), your_list[i][3], your_list[i][4],\
            your_list[i][5], your_list[i][6],\
            your_list[i][7], your_list[i][8])
        db.query(line)
        db.store_result()
    except Exception as e:
        print(e)
        continue

```

YELP_TIP:

```

create table yelp_tip(
text VARCHAR(1000),
date DATE not null,
likes INT DEFAULT 0,
business_id VARCHAR(22),
user_id VARCHAR(22),
constraint pk_tip_1 primary key(business_id, user_id),
constraint fk_tip_2 foreign key(business_id) references
yelp_business(business_id),
constraint fk_tip_3 foreign key(user_id) references yelp_user(user_id)
);

```

LOAD DATA TO YELP_TIP:

```

import _mysql
import csv

filename = 'yelp_tip.csv';
with open(filename, 'r', encoding="utf8") as f:
    reader = csv.reader(f)
    your_list = list(reader)

for i in range(0, len(your_list)):
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")
    try:
        line = "INSERT INTO yelp_tip VALUES ( '%s' , '%s', %s, '%s', %s);" %
            (your_list[i][3],\
            your_list[i][4], your_list[i][1]), your_list[i][0], your_list[i][2])
        db.query(line)
        db.store_result()
    except Exception as e:
        print(e)
        continue

```

yelp_business_hours:

```
create table yelp_business_hours(  
business_id VARCHAR(22),  
monday VARCHAR(11) DEFAULT '00:00-00:00',  
tuesday VARCHAR(11) DEFAULT '00:00-00:00',  
wednesday VARCHAR(11) DEFAULT '00:00-00:00',  
thursday VARCHAR(11) DEFAULT '00:00-00:00',  
friday VARCHAR(11) DEFAULT '00:00-00:00',  
saturday VARCHAR(11) DEFAULT '00:00-00:00',  
sunday VARCHAR(11) DEFAULT '00:00-00:00',  
constraint fk_hours_1 foreign key(business_id) references  
yelp_business(business_id)  
);
```

LOAD DATA TO YELP_BUSINESS_HOURS:

```
import _mysql  
import csv  
  
filename = 'yelp_business_hours.csv';  
with open(filename, 'r', encoding="utf8") as f:  
    reader = csv.reader(f)  
    your_list = list(reader)  
  
for i in range(0, len(your_list)):  
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")  
    try:  
        line = "INSERT INTO yelp_business_hours VALUES ( '%s' , '%s' , '%s' , '%s' ,  
'%s' , '%s' , '%s' , '%s');" \br/>        \%(your_list[i][0],your_list[i][1], your_list[i][2]),  
your_list[i][3], your_list[i][4], \br/>        your_list[i][5], your_list[i][6], your_list[i][7])  
        db.query(line)  
        db.store_result()  
    except Exception as e:  
        print(e)  
        continue
```

YELP_BUSINESS_CATEGORY:

Code to split out 'category' column from yelp_business and convert list of multiple categories which is maintained in single field to the cells in columns with atomic value(required for 1NF conversion):

```
def main():  
    csv.field_size_limit(sys.maxsize)  
    filename = 'yelp_business.csv'  
    with open(filename, 'r') as f:  
        reader = csv.reader(f)  
        your_list = list(reader)  
        cols = len(your_list[0])
```

```

total_rows = len(your_list)
output_list = list()
output_list.append([your_list[0][0], your_list[0][cols - 1]])
distinct_c = set()
for i in range(1, total_rows):
    categories = your_list[i][cols - 1]
    all_categories = categories.split(';')
    for c in all_categories:
        output_list.append([your_list[i][0], c])
        distinct_c.add(c)

list_c = list(distinct_c)
print(len(output_list))
with open('out_business.csv', 'w') as fp:
    writer = csv.writer(fp, delimiter=',')
    writer.writerows(output_list)

```

Create new table 'yelp_business_category':

```

create table yelp_business_category(
business_id VARCHAR(22),
category VARCHAR(100),
constraint pk_yelp_bus_cat_1 primary key(business_id, category),
constraint fk_yelp_bus_cat_2 foreign key(business_id) references
yelp_business(business_id)
);

```

Load Data from newly created Excel sheet in 1NF form into data table 'yelp_business_category':

```

import _mysql

import csv

filename = 'yelp_business_category.csv';
with open(filename, 'r', encoding="utf8") as f:
    reader = csv.reader(f)
    your_list = list(reader)

for i in range(0, len(your_list)):
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")
    try:
        line = "INSERT INTO yelp_business_category VALUES ( '%s' , '%s' );" %
(your_list[i][0], your_list[i][1] )
        db.query(line)
        db.store_result()
    except Exception as e:
        print(e)
        continue

```

CREATE TABLE YELP_FRIENDS: (excluding list of friends maintained in original file)

yelp_friends:

PREPARE DATA FOR YELP_FRIENDS:

Query to separate out list of friends maintained in original (one provided by Yelp) yelp_user table and create a new file that will maintain list of users and friends with one-to-one format (to reduce table to 1NF)

```
def prepareUser():
    filename = 'yelp_user.csv'
    with open(filename, 'r') as f:
        reader = csv.reader(f)
        your_list = list(reader)
        cols = len(your_list[0])
        total_rows = len(your_list)
        output_list = list()
        output_list.append([your_list[0][0], your_list[0][4]])
        for i in range(1, total_rows):
            friends = your_list[i][4]
            all = friends.split(',')
            for f in all:
                output_list.append([your_list[i][0], f])

    # Write CSV file
    with open('yelp_friends.csv', 'w') as fp:
        writer = csv.writer(fp, delimiter=',')
        writer.writerows(output_list)
```

CREATE TABLE YELP_FRIENDS:

```
create table yelp_friends(
user_id VARCHAR(22),
friend_id VARCHAR(22),
constraint fk_friend_1 foreign key(user_id) references yelp_user(user_id),
constraint fk_friend_2 foreign key(friend_id) references yelp_user(user_id)
);
```

LOAD DATA TO YELP_FRIENDS:

```
import _mysql
import csv

filename = 'yelp_friends.csv';
with open(filename, 'r', encoding="utf8") as f:
    reader = csv.reader(f)
    your_list = list(reader)

for i in range(0, len(your_list)):
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")
    try:
        line = "INSERT INTO yelp_friends VALUES ( '%s' , '%s');" % (your_list[i][0],
your_list[i][1])
        db.query(line)
        db.store_result()
    except Exception as e:
        print(e)
        continue
```

CREATE TABLE YELP_BUSINESS_ATTRIBUTES:

```
create table yelp_business_attributes(  
business_id VARCHAR(22),  
AcceptsInsurance VARCHAR(5) DEFAULT="NA",  
ByAppointmentOnly VARCHAR(5) DEFAULT="NA",  
BusinessAcceptsCreditCards VARCHAR(5) DEFAULT="NA",  
BusinessParking_garage VARCHAR(5) DEFAULT="NA",  
BusinessParking_street VARCHAR(5) DEFAULT="NA",  
BusinessParking_validated VARCHAR(5) DEFAULT="NA",  
BusinessParking_lot VARCHAR(5) DEFAULT="NA",  
BusinessParking_valet VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_coloring VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_africanamerican VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_curly VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_permes VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_kids VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_extensions VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_asian VARCHAR(5) DEFAULT="NA",  
HairSpecializesIn_straightperms VARCHAR(5) DEFAULT="NA",  
RestaurantsPriceRange2 VARCHAR(5) DEFAULT="NA",  
GoodForKids VARCHAR(5) DEFAULT="NA",  
WheelchairAccessible VARCHAR(5) DEFAULT="NA",  
BikeParking VARCHAR(5) DEFAULT="NA",  
Alcohol VARCHAR(5) DEFAULT="NA",  
HasTV VARCHAR(5) DEFAULT="NA",  
NoiseLevel VARCHAR(5) DEFAULT="NA",  
RestaurantsAttire VARCHAR(5) DEFAULT="NA",  
Music_dj VARCHAR(5) DEFAULT="NA",  
Music_background_music VARCHAR(5) DEFAULT="NA",  
Music_no_music VARCHAR(5) DEFAULT="NA",  
Music_karaoke VARCHAR(5) DEFAULT="NA",  
Music_live VARCHAR(5) DEFAULT="NA",  
Music_video VARCHAR(5) DEFAULT="NA",  
Music_jukebox VARCHAR(5) DEFAULT="NA",  
Ambience_romantic VARCHAR(5) DEFAULT="NA",  
Ambience_intimate VARCHAR(5) DEFAULT="NA",  
Ambience_classy VARCHAR(5) DEFAULT="NA",  
Ambience_hipster VARCHAR(5) DEFAULT="NA",  
Ambience_divey VARCHAR(5) DEFAULT="NA",  
Ambience_touristy VARCHAR(5) DEFAULT="NA",  
Ambience_trendy VARCHAR(5) DEFAULT="NA",  
Ambience_upscale VARCHAR(5) DEFAULT="NA",  
Ambience_casual VARCHAR(5) DEFAULT="NA",  
RestaurantsGoodForGroups VARCHAR(5) DEFAULT="NA",  
Caters VARCHAR(5) DEFAULT="NA",  
WiFi VARCHAR(5) DEFAULT="NA",  
RestaurantsReservations VARCHAR(5) DEFAULT="NA",  
RestaurantsTakeOut VARCHAR(5) DEFAULT="NA",  
HappyHour VARCHAR(5) DEFAULT="NA",  
GoodForDancing VARCHAR(5) DEFAULT="NA",  
RestaurantsTableService VARCHAR(5) DEFAULT="NA",  
OutdoorSeating VARCHAR(5) DEFAULT="NA",  
RestaurantsDelivery VARCHAR(5) DEFAULT="NA",
```



```

BestNights_monday VARCHAR(5) DEFAULT="NA",
BestNights_tuesday VARCHAR(5) DEFAULT="NA",
BestNights_friday VARCHAR(5) DEFAULT="NA",
BestNights_wednesday VARCHAR(5) DEFAULT="NA",
BestNights_thursday VARCHAR(5) DEFAULT="NA",
BestNights_sunday VARCHAR(5) DEFAULT="NA",
BestNights_saturday VARCHAR(5) DEFAULT="NA",
GoodForMeal_dessert VARCHAR(5) DEFAULT="NA",
GoodForMeal_latenight VARCHAR(5) DEFAULT="NA",
GoodForMeal_lunch VARCHAR(5) DEFAULT="NA",
GoodForMeal_dinner VARCHAR(5) DEFAULT="NA",
GoodForMeal_breakfast VARCHAR(5) DEFAULT="NA",
GoodForMeal_brunch VARCHAR(5) DEFAULT="NA",
CoatCheck VARCHAR(5) DEFAULT="NA",
Smoking VARCHAR(5) DEFAULT="NA",
DriveThru VARCHAR(5) DEFAULT="NA",
DogsAllowed VARCHAR(5) DEFAULT="NA",
BusinessAcceptsBitcoin VARCHAR(5) DEFAULT="NA",
Open24Hours VARCHAR(5) DEFAULT="NA",
BYOBCorkage VARCHAR(5) DEFAULT="NA",
BYOB VARCHAR(5) DEFAULT="NA",
Corkage VARCHAR(5) DEFAULT="NA",
DietaryRestrictions_dairy-free VARCHAR(5) DEFAULT="NA",
DietaryRestrictions_gluten-free VARCHAR(5) DEFAULT="NA",
DietaryRestrictions_vegan VARCHAR(5) DEFAULT="NA",
DietaryRestrictions_kosher VARCHAR(5) DEFAULT="NA",
DietaryRestrictions_halal VARCHAR(5) DEFAULT="NA",
DietaryRestrictions_soy-free VARCHAR(5) DEFAULT="NA",
DietaryRestrictions_vegetarian VARCHAR(5) DEFAULT="NA",
AgesAllowed VARCHAR(5) DEFAULT="NA",
RestaurantsCounterService VARCHAR(5) DEFAULT="NA",
constraint fk_attr_1 foreign key(business_id) references
yelp_business(business_id)
);

```

LOAD DATA TO YELP_BUSINESS_ATTRIBUTES:

```

import _mysql
import csv

filename = 'yelp_business_attributes.csv';
with open(filename, 'r', encoding='utf8') as f:
    reader = csv.reader(f)
    your_list = list(reader)

for i in range(0, len(your_list)):
    db = _mysql.connect("127.0.0.1", "root", "Firework@1234", "sakila")
    try:
        line = "INSERT INTO yelp_business_attributes VALUES ( '%s' , '%s', '%s', \
'%s' , '%s', '%s', '%s', '%s', '%s', '%s', '%s', \
'%s' , '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', \
'%s' , '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', \
'%s' , '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', \
"

```

```

        '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', \
        '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', \
        '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', '%s', \
    );" % (your_list[i][0], your_list[i][1], your_list[i][2]
        your_list[i][3], your_list[i][4], your_list[i][5], your_list[i][6],
your_list[i][7], your_list[i][8], \
        your_list[i][9], your_list[i][10], your_list[i][11], your_list[i][12],
your_list[i][13], your_list[i][14], \
        your_list[i][15], your_list[i][16], your_list[i][17], your_list[i][18],
your_list[i][19], your_list[i][20],
        your_list[i][21], your_list[i][22], your_list[i][23], your_list[i][24],
your_list[i][25], your_list[i][26], \
        your_list[i][27], your_list[i][28], your_list[i][29], your_list[i][30],
your_list[i][31], your_list[i][32], \
        your_list[i][33], your_list[i][34], your_list[i][35], your_list[i][36],
your_list[i][37], your_list[i][38], \
        your_list[i][39], your_list[i][40], your_list[i][41], your_list[i][42],
your_list[i][43], your_list[i][44], \
        your_list[i][45], your_list[i][45], your_list[i][46], your_list[i][47],
your_list[i][48], your_list[i][49], \
        your_list[i][50], your_list[i][51], your_list[i][52], your_list[i][53],
your_list[i][54], your_list[i][55], \
        your_list[i][56], your_list[i][57], your_list[i][58], your_list[i][59],
your_list[i][60], your_list[i][61], \
        your_list[i][62], your_list[i][63], your_list[i][64], your_list[i][65],
your_list[i][66], your_list[i][67], \
        your_list[i][68], your_list[i][69], your_list[i][70], your_list[i][71],
your_list[i][72], your_list[i][73], \
        your_list[i][74], your_list[i][75], your_list[i][76], your_list[i][77],
your_list[i][78], your_list[i][79], \
        your_list[i][80], your_list[i][81], your_list[i][82]
    )
    db.query(line)
    db.store_result()
except Exception as e:
    print(e)
    continue

```

MERGE TABLES YELP_BUSINESS, YELP_BUSINESS_ATTRIBUTES, YELP_BUSINESS_HOURS to create a single table:

```

create table yelp_business_master as (
select yb.business_id, yb.name, yb.neighborhood, yb.address, yb.city,
yb.city, yb.postal_code, yb.latitude,
yb.longitude, yb.stars, yb.review_count, yb.is_open, yba.AcceptsInsurance,
yba.ByAppointmentOnly,
yba.BusinessAcceptsCreditCards, yba.BusinessParking_garage,
yba.BusinessParking_street, yba.BusinessParking_validated,
yba.BusinessParking_lot, yba.BusinessParking_valet,
yba.HairSpecializesIn_coloring, yba.HairSpecializesIn_africanamerican,

```

```

yba.HairSpecializesIn_curly, yba.HairSpecializesIn_perms,
yba.HairSpecializesIn_kids, yba.HairSpecializesIn_extensions,
yba.HairSpecializesIn_asian, yba.HairSpecializesIn_straightperms,
yba.RestaurantsPriceRange2, yba.GoodForKids,
yba.WheelchairAccessible, yba.BikeParking, yba.Alcohol, yba.HasTV,
yba.NoiseLevel, yba.RestaurantsAttire,
yba.Music_dj, yba.Music_background_music, yba.Music_no_music,
yba.Music_karaoke, yba.Music_live, yba.Music_video,
yba.Music_jukebox, yba.Ambience_romantic, yba.Ambience_intimate,
yba.Ambience_classy, yba.Ambience_hipster,
yba.Ambience_divey, yba.Ambience_touristy, yba.Ambience_trendy,
yba.Ambience_upscale, yba.Ambience_casual,
yba.RestaurantsGoodForGroups, yba.Caters, yba.WiFi,
yba.RestaurantsReservations, yba.RestaurantsTakeOut,
yba.HappyHour, yba.GoodForDancing, yba.RestaurantsTableService
information_schema, yba.OutdoorSeating,
yba.RestaurantsDelivery, yba.BestNights_monday, yba.BestNights_tuesday,
yba.BestNights_friday, yba.BestNights_wednesday,
yba.BestNights_thursday, yba.BestNights_sunday, yba.BestNights_saturday,
yba.GoodForMeal_dessert, yba.GoodForMeal_latenight,
yba.GoodForMeal_lunch, yba.GoodForMeal_dinner, yba.GoodForMeal_breakfast,
yba.GoodForMeal_brunch, yba.CoatCheck,
yba.Smoking, yba.DriveThru, yba.DogsAllowed, yba.BusinessAcceptsBitcoin,
yba.Open24Hours, yba.BYOBcorkage, yba.BYOB,
yba.Corkage, yba.DietaryRestrictions_dairy-free,
yba.DietaryRestrictions_gluten-free, yba.DietaryRestrictions_vegan,
yba.DietaryRestrictions_kosher, yba.DietaryRestrictions_halal,
yba.DietaryRestrictions_soy-free,
yba.DietaryRestrictions_vegetarian, yba.AgesAllowed,
yba.RestaurantsCounterService, ybh.monday, ybh.tuesday,
ybh.wednesday, ybh.thursday, ybh.friday, ybh.saturday, ybh.sunday
from yelp_business as yb, yelp_business_attributes as yba,
yelp_business_hours as ybh
where yb.business_id.business_id = yba.business_id and yb.business_id =
ybh.business_id
);

```