Final

Raein Hashemi

December 7, 2016

My idea for this project is to build a database consisting of all kinds of local businesses with their attributes like their names, the price for their services, their review count, telephone number and etc. The focus for me is to be able to search for these businesses with broader domain, without the focus on just one attribute. For example, Yelp has this kind of data but users have to search for a specific type of business to get results. What if they wanted to combine two or more kinds of local businesses with for example a low price range? They also have to specify a city or neighborhood to get results from, when with this database they can combine results with different cities or neighborhoods. These restrictions can be averted by collecting the data from websites like Yelp and putting them in a database and retrieve data with whatever filters we want.

I will parse the data in YELP search pages. It provides some attributes such as Name, Type, Price, number of reviews and the Tell number for each element. The Elements could be restaurants, bars, coffee-shops or so many other places. Then the user is able to choose which information he/she wants as an output. For different cities and neighborhoods, I have to send multiple requests to Yelp and combine the results. Each time the user sends a request, the collected data is stored in our local database. The next time we want to search within this domain, the program won't send a request to Yelp, instead our database will be used. It will work as a cache. If the data is within our database, it will be retrieved immediately. But if it's not there, the database will be updated with new data from the Yelp website. Collect, Store and Retrieve.

```
cat("\014")       #This clears the Consol

rm(list=ls())     #This removes all the variables previously existed in Global Environment.

library(RCurl)
```

```
## Warning: package 'RCurl' was built under R version 3.2.5
```

```
## Loading required package: bitops
```

```
library(XML)
```

```
## Warning: package 'XML' was built under R version 3.2.5
```

```
library(httr)
```

```
## Warning: package 'httr' was built under R version 3.2.4
```

```
library(sqldf)
```

```
## Warning: package 'sqldf' was built under R version 3.2.5
```

```
## Loading required package: gsubfn

## Warning: package 'gsubfn' was built under R version 3.2.5

## Loading required package: proto

## Loading required package: RSQLite

## Warning: package 'RSQLite' was built under R version 3.2.5

## Loading required package: DBI

## Warning: package 'DBI' was built under R version 3.2.5
```

```
db <- dbConnect(SQLite(), dbname="SBDB.sqlite")

tableSize <- dbListTables(conn=db)

# If file didn't exist or it didn't have any tables, create a new one
if(length(tableSize) == 0) {

  sqldf("attach 'SBDB.sqlite' as new")

  dbSendQuery(conn = db,
              "CREATE TABLE SmallB(DataID INTEGER primary key autoincrement,
              ID INTEGER,
              Name TEXT,
              Type TEXT,
              Price INTEGER,
              Score INTEGER,
              Reviews INTEGER,
              Phone TEXT,
              Address TEXT,
              Business TEXT,
              Location TEXT)")
}

# YelpParse Function parses data in the YELP search pages. It provides Name, Type, Price,
# Score, number of Reviews, Phone number and the Address for each element.
# The Elements could be restaurants or bars or coffee-shops or so many other places.
# User is able to choose which information he/she wants as output by putting the names of
# the Business and their Location in the correponding places for input.

YelpParse <- function(page, business, location) {

          ######################################################################

          # This is the URL of the website we need to scrape and get information

          link <- paste("http://www.yelp.com/search?find_desc=", business,
                        "&find_loc=", location, "&start=", as.character(page*10),
                        "&sortby=review_count", sep="")
```

```
theurl <- link
theurl <- gsub(" ","",theurl)   #No extra spaces

webpage <- getURL(theurl, .opts=curlOptions(followlocation = TRUE))

# convert the page into a line-by-line format rather than a single string
tc <- textConnection(webpage)
#webpage is now a vector of string each elament is a line of string
webpage <- readLines(tc)
close(tc)

pagetree <- htmlTreeParse(webpage, useInternalNodes = TRUE)  #pagetree is now in html format



#########################  NAME  ################################

business.name <- unlist(xpathApply(pagetree,
                          "//*/span[@class='indexed-biz-name']/a[@*][@*][@*]", xmlValue))

if(length(business.name) == 11) # Sometimes it gives 11 elements and the first one is wrong
    business.name <- business.name[2:11]

business.name <- as.character(business.name)
business.name <- gsub("\u0092", "", business.name)
business.name

############################   REVIEW SCORE  ############################

review.score <- unlist(xpathApply(pagetree,
                        "//*/img[@class='offscreen']", xmlGetAttr, "alt"))
review.score

if(length(review.score) < 10)
  review.score <- rep.int("0", 10)

review.score <- gsub(" star rating", "", review.score) #Removing extra characters"
review.score <- gsub("\n            ", "", review.score)
review.score <- gsub("\n    ", "", review.score)

if(length(review.score) == 11)
    review.score <- review.score[2:11]

review.score <- as.numeric(review.score)
review.score

############################   REVIEW COUNT  ############################

review.count <- unlist(xpathApply(pagetree,
                        "//*/span[@class='review-count rating-qualifier']",xmlValue))
review.count

if(length(review.count) < 10)
```

```
  review.count <- rep.int("0", 10)

review.count <- gsub("\n            ", "", review.count) #Removing extra characters"
review.count <- gsub("\n    ", "", review.count)
review.count <- gsub(" reviews", "", review.count)
review.count <- gsub(" review", "", review.count)

if(length(review.count) == 11)
    review.count <- review.count[2:11]

review.count <- as.numeric(review.count)
review.count

############################## PRICE ##############################

business.price <- unlist(xpathApply(pagetree,
                        "//*/span[@class='business-attribute price-range']",xmlValue))
business.price

if(length(business.price) != 0)
  for (i in 1:length(business.price)) # Scaling price notations to 1,2,3,4
  {
      if (business.price[i]=="$")  {business.price[i]="1" }
      if (business.price[i]=="$$")  {business.price[i]="2" }
      if (business.price[i]=="$$$")  {business.price[i]="3" }
      if (business.price[i]=="$$$$")  {business.price[i]="4" }
  }

else business.price <- "0"

business.price

if(length(business.price) == 11)
    business.price<-business.price[2:11]

business.price <- as.numeric(business.price)
business.price


########################  TYPE  ############################

# Some times there are several <a> tags. We need the first one.
business.type <- unlist(xpathApply(pagetree,
                                "//*/span[@class='category-str-list']/a[@*][1]",xmlValue))

if(length(business.type) == 11)
    business.type <- business.type[2:11]

business.type <- as.character(business.type)
business.type

#######################   ADDRESS  ##############################
```

```
          business.address <- unlist(xpathApply(pagetree, "//*/address",xmlValue))

          business.address <- gsub("\n    ", "", business.address)
          business.address <- gsub("\n", "", business.address)
          business.address <- gsub("'", "", business.address)

          if(length(business.address) == 11)
              business.address<-business.address[2:11]

          business.address <- as.character(business.address)
          business.address

          ############################  TELL  ##############################

          business.tell <- unlist(xpathApply(pagetree,
                           "//*/div[@class='secondary-attributes']/span[@class='biz-phone']", xmlValue)

          business.tell <- gsub("\n        ", "", business.tell)
          business.tell <- gsub("\n    ", "", business.tell)

          if(length(business.tell) == 11)
              business.tell<-business.tell[2:11]

          business.tell <- as.character(business.tell)
          business.tell


          ################################ Putting ALL in a DATA FRAME ############################


          business.data <- data.frame(NAME=business.name,
                                      TYPE=business.type,
                                      PRICE= business.price,
                                      SCORE= review.score,
                                      REVIEW_COUNT=review.count,
                                      TELL=business.tell,
                                      ADD=business.address,
                                      BUS=business,
                                      LOC=location)

          return (business.data);
}

#### Beginning of Data Collection, Storage and Retrieval

# empty result set
business.data <- data.frame();

# 150 results (15 pages of 10 results) for Restaurants in Boston
for (i in 0:14) { # 15 pages

  # Use the function to fetch data, one page at a time (10 results per page)
  business.data <- YelpParse(i, "Restaurants", "Boston")
```

```
  # Collect the data, and set the ID attribute based on whether it's been in our database or not
  for(j in 1:10) # 10 results per page
    dbSendQuery(conn = db,
            paste("INSERT OR REPLACE INTO SmallB
                  (ID, Name, Type, Price, Score, Reviews, Phone, Address, Business, Location)
                  VALUES (COALESCE((select ID from SmallB where Name = '",
              as.character(business.data[j,]$NAME), "' and TYPE = '",
              as.character(business.data[j,]$TYPE), "'),", as.character(i*10+j) ,"), '",

              as.character(business.data[j,]$NAME), "', '",
              as.character(business.data[j,]$TYPE), "', ",
              as.character(business.data[j,]$PRICE), ", ",
              as.character(business.data[j,]$SCORE), ", ",
              as.character(business.data[j,]$REVIEW_COUNT), ", '",
              as.character(business.data[j,]$TELL), "', '",
              as.character(business.data[j,]$ADD), "', '",
              as.character(business.data[j,]$BUS), "', '",
              as.character(business.data[j,]$LOC), "')"))

}

# 150 results (15 pages of 10 results) for Restaurants in San Francisco
for (i in 0:14) { # 15 pages

  # Use the function to fetch data, one page at a time (10 results per page)
  business.data <- YelpParse(i, "Restaurants", "San Francisco")

  # Collect the data, and set the ID attribute based on whether it's been in our database or not
  for(j in 1:10) # 10 results per page
    dbSendQuery(conn = db,
            paste("INSERT OR REPLACE INTO SmallB
                  (ID, Name, Type, Price, Score, Reviews, Phone, Address, Business, Location)
                  VALUES (COALESCE((select ID from SmallB where Name = '",
              as.character(business.data[j,]$NAME), "' and TYPE = '",
              as.character(business.data[j,]$TYPE), "'),", as.character(i*10+j) ,"), '",

              as.character(business.data[j,]$NAME), "', '",
              as.character(business.data[j,]$TYPE), "', ",
              as.character(business.data[j,]$PRICE), ", ",
              as.character(business.data[j,]$SCORE), ", ",
              as.character(business.data[j,]$REVIEW_COUNT), ", '",
              as.character(business.data[j,]$TELL), "', '",
              as.character(business.data[j,]$ADD), "', '",
              as.character(business.data[j,]$BUS), "', '",
              as.character(business.data[j,]$LOC), "')"))

}

# Delete duplicate results
dbSendQuery(conn = db,
            "DELETE FROM SmallB WHERE DataID NOT IN (
                SELECT min(DataID)
                FROM SmallB
```

```
                GROUP BY ID, Business, Location
              )")
```

```
## <SQLiteResult>
```

Now we can answer our questions:

1) Show the names and ratings of high cost Restaurants in both Boston and San Francisco.

```
dbGetQuery(conn = db, paste("SELECT Name,Type,Score,Price,Location FROM SmallB
                        WHERE (Location = ", "\"", "Boston", "\"",
                        " OR Location = ", "\"", "San Francisco", "\")",
                        " AND Business = ", "\"", "Restaurants", "\"",
                        " AND Price = '4'"))
```

```
##             Name             Type Score Price        Location
## 1          O Ya         Japanese    4.5     4         Boston
## 2         Ostra          Seafood    4.5     4         Boston
## 3          Asta   American (New)    4.5     4         Boston
## 4    Gary Danko   American (New)    4.5     4  San Francisco
## 5  The Progress   American (New)    4.0     4  San Francisco
## 6   Californios          Mexican    4.5     4  San Francisco
```

2) What are the top rated hospitals with +100 reviews in Boston?

```
# 100 results (10 pages of 10 results) for Hospitals in Boston
for (i in 0:9) { # 10 pages

  # Use the function to fetch data, one page at a time (10 results per page)
  business.data <- YelpParse(i, "Hospitals", "Boston")

  # Collect the data, and set the ID attribute based on whether it's been in our database or not
  for(j in 1:10) # 10 results per page
    dbSendQuery(conn = db,
            paste("INSERT OR REPLACE INTO SmallB
                  (ID, Name, Type, Price, Score, Reviews, Phone, Address, Business, Location)
                  VALUES (COALESCE((select ID from SmallB where Name = '",
              as.character(business.data[j,]$NAME), "' and Type = '",
              as.character(business.data[j,]$TYPE), "' and Address = '",
              as.character(business.data[j,]$ADD), "'),", as.character(i*10+j) ,"), '",

              as.character(business.data[j,]$NAME), "', '",
              as.character(business.data[j,]$TYPE), "', ",
              as.character(business.data[j,]$PRICE), ", ",
              as.character(business.data[j,]$SCORE), ", ",
              as.character(business.data[j,]$REVIEW_COUNT), ", '",
              as.character(business.data[j,]$TELL), "', '",
              as.character(business.data[j,]$ADD), "', '",
              as.character(business.data[j,]$BUS), "', '",
```

```
              as.character(business.data[j,]$LOC), "')"))

}

# Delete duplicate results
dbSendQuery(conn = db,
            "DELETE FROM SmallB WHERE DataID NOT IN (
               SELECT min(DataID)
               FROM SmallB
               GROUP BY ID, Business, Location
             )")
```

```
## <SQLiteResult>
```

```
dbGetQuery(conn = db, paste("SELECT Name,Type,Reviews,Score,Location FROM SmallB
                        WHERE Location = ", "\"", "Boston", "\"",
                        " AND Business = ", "\"", "Hospitals", "\"",
                        " AND Reviews > '100'",
                        " AND Score >= '3'"))
```

```
##                                     Name              Type Reviews Score
## 1        Massachusetts General Hospital         Hospitals      151   3.5
## 2             Brigham & Womens Hospital         Hospitals      121   3.5
## 3  Beth Israel Deaconess Medical Center   Medical Centers      115   3.0
##   Location
## 1  Boston
## 2  Boston
## 3  Boston
```

3) What types of restaurants are least and most expensive in San Francisco?

```
dbGetQuery(conn = db, paste("SELECT Type,Price,Location FROM SmallB
                        WHERE Location = ", "\"", "San Francisco", "\"",
                        " AND Business = ", "\"", "Restaurants", "\"",
                        " AND (Price = '1'",
                        " OR Price = '4')",
                        " GROUP BY Type",
                        " ORDER BY Price Desc"))
```

```
##                        Type Price        Location
## 1           American (New)      4  San Francisco
## 2                  Mexican      4  San Francisco
## 3   American (Traditional)      1  San Francisco
## 4             Asian Fusion      1  San Francisco
## 5                 Bakeries      1  San Francisco
## 6       Breakfast & Brunch      1  San Francisco
## 7                  British      1  San Francisco
## 8            Chicken Wings      1  San Francisco
## 9                  Chinese      1  San Francisco
## 10            Coffee & Tea      1  San Francisco
```

```
## 11                   Delis      1  San Francisco
## 12                 Falafel      1  San Francisco
## 13             Food Stands      1  San Francisco
## 14           Mediterranean      1  San Francisco
## 15              Sandwiches      1  San Francisco
## 16                 Seafood      1  San Francisco
## 17                Taiwanese      1  San Francisco
## 18              Vietnamese      1  San Francisco
```

4) List the best restaurants in San Francisco? (lower price, higher score, higher review count)

```
dbGetQuery(conn = db, paste("SELECT Name,Price,Score,Reviews,Location FROM SmallB
                        WHERE Location = ", "\"", "San Francisco", "\"",
                        " AND Business = ", "\"", "Restaurants", "\"",
                        " ORDER BY Price ASC,",
                        " Score DESC, Reviews DESC"))
```

```
##                                Name Price Score Reviews        Location
## 1                    Saigon Sandwich     1   4.5    2800  San Francisco
## 2       The Codmother Fish and Chips     1   4.5    2046  San Francisco
## 3                               HRD     1   4.5    1953  San Francisco
## 4                         Lous Cafe     1   4.5    1233  San Francisco
## 5                Hot Sauce and Panko     1   4.5     788  San Francisco
## 6                 The Flying Falafel     1   4.5     568  San Francisco
## 7                               Bite     1   4.5     542  San Francisco
## 8                     Wooly Pig Cafe     1   4.5     486  San Francisco
## 9              Basa Seafood Express      1   4.5     431  San Francisco
## 10                      Box Kitchen     1   4.5     431  San Francisco
## 11                     The Chairman     1   4.5     380  San Francisco
## 12                      Street Taco     1   4.5     370  San Francisco
## 13                          Tacorea     1   4.5     358  San Francisco
## 14                      El Farolito     1   4.0    4001  San Francisco
## 15                      La Taqueria     1   4.0    2814  San Francisco
## 16                     Tommys Joynt     1   4.0    2401  San Francisco
## 17             Good Mong Kok Bakery     1   4.0    1501  San Francisco
## 18                     farm : table     1   4.0    1228  San Francisco
## 19                      Cordon Bleu     1   4.0     899  San Francisco
## 20                        Dinosaurs     1   4.0     746  San Francisco
## 21                          Palmyra     1   4.0     639  San Francisco
## 22                Velo Rouge Cafe     1   4.0     618  San Francisco
## 23                          Tempest     1   4.0     550  San Francisco
## 24                     Bacon Bacon      1   4.0     281  San Francisco
## 25                       Kung Food     1   4.0     243  San Francisco
## 26                      DragonEats     1   4.0     238  San Francisco
## 27            Flying Pig Bistro Pub     1   4.0     213  San Francisco
## 28             Hog Island Oyster Co     2   4.5    3998  San Francisco
## 29                     Sotto Mare      2   4.5    2535  San Francisco
## 30                     Chez Maman      2   4.5    1677  San Francisco
## 31                   B Patisserie      2   4.5    1502  San Francisco
## 32                     Don Pistos      2   4.5    1500  San Francisco
## 33                         Garaje      2   4.5    1071  San Francisco
## 34                     La FusiÃ³n      2   4.5     881  San Francisco
```

```
## 35                        Leopolds      2   4.5     874  San Francisco
## 36          Italian Homemade Company      2   4.5     650  San Francisco
## 37                          Hogwash      2   4.5     535  San Francisco
## 38                        Eight AM       2   4.5     501  San Francisco
## 39             Hopwater Distribution     2   4.5     494  San Francisco
## 40                    Old Skool Cafe      2   4.5     421  San Francisco
## 41                  Rustys Southern       2   4.5     389  San Francisco
## 42                    Rove Kitchen        2   4.5     321  San Francisco
## 43          De Afghanan Kabob House       2   4.5     298  San Francisco
## 44                  Cafe du Soleil        2   4.5     285  San Francisco
## 45      The Italian Homemade Company      2   4.5     283  San Francisco
## 46                Myriad Gastropub        2   4.5     276  San Francisco
## 47                Kitchen Istanbul        2   4.5     275  San Francisco
## 48       Montesacro Pinseria-Enoteca      2   4.5     271  San Francisco
## 49                          Paprika       2   4.5     255  San Francisco
## 50           Marcellas Lasagneria         2   4.5     247  San Francisco
## 51                       Lite Bite       2   4.5     235  San Francisco
## 52                  Red Hill Station     2   4.5     201  San Francisco
## 53                     The Spice Jar     2   4.5     200  San Francisco
## 54                          Stuffed      2   4.5     183  San Francisco
## 55                    Rooster & Rice     2   4.5     179  San Francisco
## 56                          20 Spot      2   4.5     176  San Francisco
## 57                    All Good Pizza     2   4.5     164  San Francisco
## 58                           Takoba      2   4.5     159  San Francisco
## 59                      Lord George      2   4.5     151  San Francisco
## 60                  KitTea Cat Cafe      2   4.5     140  San Francisco
## 61                  Burma Superstar      2   4.0    5647  San Francisco
## 62                          San Tung     2   4.0    5362  San Francisco
## 63                            Zazie      2   4.0    3543  San Francisco
## 64           Dotties True Blue Cafe      2   4.0    3420  San Francisco
## 65                          Beretta      2   4.0    2934  San Francisco
## 66                       SuppenkÃ¼che    2   4.0    2829  San Francisco
## 67                         Zero Zero     2   4.0    2680  San Francisco
## 68                  LimÃ³n Rotisserie    2   4.0    2669  San Francisco
## 69                 Mission Beach Cafe    2   4.0    2312  San Francisco
## 70                       Sweet Maple     2   4.0    2167  San Francisco
## 71                          Marlowe      2   4.0    2007  San Francisco
## 72                   The Monks Kettle    2   4.0    1968  San Francisco
## 73                      Chow - Church    2   4.0    1816  San Francisco
## 74                    Kitchen Story      2   4.0    1813  San Francisco
## 75           Woodhouse Fish Company      2   4.0    1785  San Francisco
## 76                              Plow     2   4.0    1755  San Francisco
## 77                       The Alembic     2   4.0    1579  San Francisco
## 78                            LolÃ³    2   4.0    1532  San Francisco
## 79                              SO       2   4.0    1468  San Francisco
## 80                    Pacific Catch     2   4.0    1375  San Francisco
## 81                            Straw     2   4.0    1316  San Francisco
## 82             The Little Chihuahua     2   4.0    1306  San Francisco
## 83                            Skool     2   4.0    1270  San Francisco
## 84                   La Mediterranee    2   4.0    1264  San Francisco
## 85       Basil Thai Restaurant & Bar    2   4.0    1253  San Francisco
## 86                   Universal Cafe     2   4.0    1191  San Francisco
## 87                           Lavash     2   4.0    1182  San Francisco
## 88                    Red Door Cafe     2   4.0    1092  San Francisco
```

```
## 89                          Tataki      2   4.0    1040  San Francisco
## 90                        El Techo      2   4.0    1003  San Francisco
## 91            Fino Ristorante & Bar     2   4.0     852  San Francisco
## 92                        Nopalito      2   4.0     831  San Francisco
## 93                       Parada 22      2   4.0     827  San Francisco
## 94                   Mikkeller Bar      2   4.0     824  San Francisco
## 95                          Gamine      2   4.0     816  San Francisco
## 96         Old Jerusalem Restaurant     2   4.0     816  San Francisco
## 97                            Domo      2   4.0     769  San Francisco
## 98                       Fat Angel      2   4.0     737  San Francisco
## 99                 Chez Maman West      2   4.0     729  San Francisco
## 100                         Souvla      2   4.0     675  San Francisco
## 101                  Griddle Fresh      2   4.0     646  San Francisco
## 102                   The Sycamore      2   4.0     611  San Francisco
## 103                       Schmidts      2   4.0     551  San Francisco
## 104                         Maven       2   4.0     540  San Francisco
## 105            Papito Potrero Hill      2   4.0     539  San Francisco
## 106                 Lupa Trattoria     2   4.0     531  San Francisco
## 107                    Burma Love      2   4.0     512  San Francisco
## 108                  Azucar Lounge     2   4.0     506  San Francisco
## 109           Brendas Meat & Three     2   4.0     506  San Francisco
## 110        Canela Bistro & Wine Bar     2   4.0     489  San Francisco
## 111  Farmhouse Kitchen Thai Cuisine     2   4.0     455  San Francisco
## 112                       Causwells     2   4.0     450  San Francisco
## 113                         Upcider     2   4.0     432  San Francisco
## 114                         Noeteca     2   4.0     375  San Francisco
## 115                         Redford     2   4.0     353  San Francisco
## 116                           Fable     2   4.0     337  San Francisco
## 117                       Iza Ramen     2   4.0     296  San Francisco
## 118     Big Chef Toms Belly Burgers     2   4.0     258  San Francisco
## 119                   Deccan Spice      2   4.0     245  San Francisco
## 120                          Kazan      2   4.0     212  San Francisco
## 121             Elmira Rosticceria     2   4.0     208  San Francisco
## 122 Thoughts Style Cuisine Showroom     2   4.0     183  San Francisco
## 123                       The House     3   4.5    3892  San Francisco
## 124             Kokkari Estiatorio     3   4.5    3717  San Francisco
## 125                       Chapeau!      3   4.5    2399  San Francisco
## 126                        Frances      3   4.5    1281  San Francisco
## 127            Anchor Oyster Bar      3   4.5    1172  San Francisco
## 128           Liholiho Yacht Club     3   4.5    1066  San Francisco
## 129                      La Ciccia     3   4.5     894  San Francisco
## 130                  Stones Throw      3   4.5     674  San Francisco
## 131                  The Richmond      3   4.5     582  San Francisco
## 132                      Prubechu      3   4.5     203  San Francisco
## 133                         Nopa       3   4.0    4341  San Francisco
## 134                Foreign Cinema      3   4.0    4088  San Francisco
## 135          State Bird Provisions     3   4.0    1968  San Francisco
## 136             Firefly Restaurant     3   4.0    1332  San Francisco
## 137                       Lolinda      3   4.0    1292  San Francisco
## 138                    Blue Plate      3   4.0    1226  San Francisco
## 139                    Rich Table      3   4.0     774  San Francisco
## 140                    Cockscomb       3   4.0     632  San Francisco
## 141                       Bouche      3   4.0     589  San Francisco
## 142                         1760      3   4.0     564  San Francisco
```

```
## 143                Monsieur Benjamin      3   4.0     558  San Francisco
## 144                         Octavia      3   4.0     259  San Francisco
## 145               1601 Bar & Kitchen      3   4.0     182  San Francisco
## 146                         Bellota      3   4.0     148  San Francisco
## 147                      Gary Danko      4   4.5    4607  San Francisco
## 148                      Californios      4   4.5     163  San Francisco
## 149                     The Progress      4   4.0     490  San Francisco
```

5) List the best meeting places in Boston? (restaurants, bars and coffee shops)

```
# 100 results (10 pages of 10 results) for Bars in Boston
for (i in 0:1) { # 10 pages

  # Use the function to fetch data, one page at a time (10 results per page)
  business.data <- YelpParse(i, "Bars", "Boston")

  # Collect the data, and set the ID attribute based on whether it's been in our database or not
  for(j in 1:10) # 10 results per page
    dbSendQuery(conn = db,
            paste("INSERT OR REPLACE INTO SmallB
                  (ID, Name, Type, Price, Score, Reviews, Phone, Address, Business, Location)
                  VALUES (COALESCE((select ID from SmallB where Name = '",
              as.character(business.data[j,]$NAME), "' and TYPE = '",
              as.character(business.data[j,]$TYPE), "'),", as.character(i*10+j) ,"), '",

              as.character(business.data[j,]$NAME), "', '",
              as.character(business.data[j,]$TYPE), "', ",
              as.character(business.data[j,]$PRICE), ", ",
              as.character(business.data[j,]$SCORE), ", ",
              as.character(business.data[j,]$REVIEW_COUNT), ", '",
              as.character(business.data[j,]$TELL), "', '",
              as.character(business.data[j,]$ADD), "', '",
              as.character(business.data[j,]$BUS), "', '",
              as.character(business.data[j,]$LOC), "')"))

}

# 100 results (10 pages of 10 results) for Coffee shops in Boston
for (i in 0:1) { # 10 pages

  # Use the function to fetch data, one page at a time (10 results per page)
  business.data <- YelpParse(i, "Coffeeshops", "Boston")

  # Collect the data, and set the ID attribute based on whether it's been in our database or not
  for(j in 1:10) # 10 results per page
    dbSendQuery(conn = db,
            paste("INSERT OR REPLACE INTO SmallB
                  (ID, Name, Type, Price, Score, Reviews, Phone, Address, Business, Location)
                  VALUES (COALESCE((select ID from SmallB where Name = '",
              as.character(business.data[j,]$NAME), "' and TYPE = '",
              as.character(business.data[j,]$TYPE), "'),", as.character(i*10+j) ,"), '",
```

```
              as.character(business.data[j,]$NAME), "', '",
              as.character(business.data[j,]$TYPE), "', ",
              as.character(business.data[j,]$PRICE), ", ",
              as.character(business.data[j,]$SCORE), ", ",
              as.character(business.data[j,]$REVIEW_COUNT), ", '",
              as.character(business.data[j,]$TELL), "', '",
              as.character(business.data[j,]$ADD), "', '",
              as.character(business.data[j,]$BUS), "', '",
              as.character(business.data[j,]$LOC), "')"))

}

# Delete duplicate results
dbSendQuery(conn = db,
            "DELETE FROM SmallB WHERE DataID NOT IN (
               SELECT min(DataID)
               FROM SmallB
               GROUP BY ID, Business, Location, Address
             )")
```

```
## <SQLiteResult>
```

```
dbGetQuery(conn = db, paste("SELECT Name,Price,Score,Reviews,Business,Location FROM SmallB
                        WHERE Location = ", "\"", "Boston", "\"",
                        " AND Price >= '2'",
                        " AND (Business = ", "\"", "Restaurants", "\"",
                        " OR Business = ", "\"", "Bars", "\"",
                        " OR Business = ", "\"", "Coffeeshops", "\")",
                        " ORDER BY Price ASC,",
                        " Score DESC, Reviews DESC LIMIT 20"))
```

```
##                             Name Price Score Reviews      Business
## 1           Flour Bakery + CafÃ©      2   4.5     817  Restaurants
## 2           Flour Bakery + CafÃ©      2   4.5     817  Coffeeshops
## 3                  Sam LaGrassas      2   4.5     739  Restaurants
## 4                         Paulis      2   4.5     548  Restaurants
## 5                  Lukes Lobster      2   4.5     483  Restaurants
## 6   Barcelona Wine Bar South End      2   4.5     469  Restaurants
## 7   Barcelona Wine Bar South End      2   4.5     469         Bars
## 8                   Thinking Cup      2   4.5     461  Restaurants
## 9                   Thinking Cup      2   4.5     461  Coffeeshops
## 10                 Cafe Polonia      2   4.5     347  Restaurants
## 11      Italian Express Pizzeria      2   4.5     264  Restaurants
## 12               Monicas Mercato      2   4.5     250  Restaurants
## 13           The Glenville Stops      2   4.5     190  Restaurants
## 14         North End Fish Market      2   4.5     184  Restaurants
## 15       KO Pies At the Shipyard      2   4.5     181  Restaurants
## 16              The Brewers Fork      2   4.5     157  Restaurants
## 17                      Pomodoro      2   4.5     142  Restaurants
## 18        Mi Pueblito Restaurant      2   4.5     121  Restaurants
## 19                     DavidsTea      2   4.5      85  Coffeeshops
## 20                     Gracenote      2   4.5      63  Coffeeshops
##    Location
```

```
## 1   Boston
## 2   Boston
## 3   Boston
## 4   Boston
## 5   Boston
## 6   Boston
## 7   Boston
## 8   Boston
## 9   Boston
## 10  Boston
## 11  Boston
## 12  Boston
## 13  Boston
## 14  Boston
## 15  Boston
## 16  Boston
## 17  Boston
## 18  Boston
## 19  Boston
## 20  Boston
```

For validation I checked every result with the Yelp website. So they are collected, saved and retrieved correctly, because if either of those steps had a problem I wouldn't be getting the results matching with Yelp.

The ER model would be a single table because all the data I collected are attributes of small businesses (Name, Type, Price, Score, Reviews, Address, Phone) and not another entity. I could have used another table for addresses, but having that I had to add lots of other Keys for the new table and also foreign keys for the main table, all for a few businesses that have multiple branches. It didn't seem efficient so I didn't use it.

These were the questions that neither Yelp nor any other similar website would be able to answer. I have collected the data from these sources, stored them in my database and retrieved whatever information I wanted from them. This system can be used in many other ways and can benefit users of these Yelp-like websites.

```
dbDisconnect(db)
```

```
## [1] TRUE
```