PubmedDB

#PubMedDB ####Jordan Harrop ####Robert Passas The following is a database created to store pubmed publication data, provided as XML data. It concludes with queries exploring publication patterns.

##Connect to the Database

```
# 1. Library
library(RMySQL)
```

```
## Loading required package: DBI
```

```
library(XML)
library(DBI)
library(knitr)
```

```
# 2. Settings (Jordan's db)
db_user <- 'cs5200practicum2'
db_password <- 'tctvuje8'
db_name <- 'dbpracticum2'
db_host <- 'practicum2.cb9tzbdsyfxk.us-east-2.rds.amazonaws.com'
db_port <- 3306

# 3. Read data from db
mydb <-  dbConnect(MySQL(), user = db_user, password = db_password,
                 dbname = db_name, host = db_host, port = db_port)
```

#These are our file paths, they should be false to run

```
if(FALSE){
path <- "C:/Users/jorda/Documents/CS_Masters/CS5200_Databases/Homework/Practicum2/"
fn <- "pubmed_sample.xml"
fpn = paste0(path, fn)
}
```

#These are our file paths, they should be false to run

```
path <-"/Users/robert/Documents/CS5200/Practicum2/"
fn <- "pubmed_sample.xml"
fpn = paste0(path, fn)
```

```
# Reading the XML file and parse into DOM
xmlDOM <- xmlParse(file = fpn)

# get the root node of the DOM tree
r <- xmlRoot(xmlDOM)
```

###PART 1; Question 2 ##Define the Tables and Data Frames that will hold XML

```
CREATE TABLE IF NOT EXISTS Author (
  author_id INT NOT NULL PRIMARY KEY,
  first_name TEXT NOT NULL,
  last_name TEXT NOT NULL
);
```

```r
Author.df <- data.frame (author_id = integer(),
                          first_name = character(),
                          last_name = character(),
                          stringsAsFactors = F)
```

```sql
CREATE TABLE IF NOT EXISTS HistoryStatus (
  status_id INT NOT NULL PRIMARY KEY,
  pub_status TEXT NOT NULL,
  status_date DATE NOT NULL
);
```

```r
HistoryStatus.df <- data.frame (status_id = integer(),
                          pub_status = character(),
                          status_date = character(),
                          stringsAsFactors = F)
```

```sql
CREATE TABLE IF NOT EXISTS JournalIssue (
  issue_id INT PRIMARY KEY,
  journal_title VARCHAR(200) NOT NULL,
  cited_medium TEXT NOT NULL,
  volume INT NOT NULL,
  issue INT NOT NULL,
  pub_date DATE NOT NULL,
  FOREIGN KEY (journal_id) REFERENCES Journal(journal_id)
  ON DELETE CASCADE
);
```

```r
Issue.df <- data.frame (pub_id = integer(),
                        journal_title = character(),
                          cited_medium = character(),
                          volume = integer(),
                          issue = integer(),
                          pub_date_year = integer(),
                          pub_date_month = character(),
                          stringsAsFactors = F)
```

```sql
CREATE TABLE IF NOT EXISTS Article (
  article_id INT NOT NULL PRIMARY KEY,
  issue_id INT NOT NULL,
  article_title TEXT NOT NULL,
  FOREIGN KEY (issue_id) REFERENCES JournalIssue(issue_id)
  ON DELETE CASCADE
);
```

```r
numArticles <- xmlSize(r)

Article.df <- data.frame (article_id = integer(),
                          issue_id = integer(),
                          article_title = character(),
                          stringsAsFactors = F)
```

```sql
CREATE TABLE IF NOT EXISTS AuthorJoin (
  author_id INT NOT NULL,
  article_id INT NOT NULL,
  FOREIGN KEY (author_id) REFERENCES Author(author_id)
  ON DELETE CASCADE,
  FOREIGN KEY (article_id) REFERENCES Article(article_id)
  ON DELETE CASCADE,
  PRIMARY KEY (author_id, article_id)
);
```

```
AuthorJoin.df <- data.frame (author_id = integer(),
                          article_id = integer(),
                          stringsAsFactors = F)
```

```
CREATE TABLE IF NOT EXISTS History (
  status_id INT NOT NULL,
  article_id INT NOT NULL,
  FOREIGN KEY (status_id) REFERENCES HistoryStatus(status_id)
  ON DELETE CASCADE,
  FOREIGN KEY (article_id) REFERENCES Article(article_id)
  ON DELETE CASCADE,
  PRIMARY KEY (status_id, article_id)
);
```

```
History.df <- data.frame (status_id = integer(),
                          article_id = integer(),
                          stringsAsFactors = F)
```

###PART 1; Question 3 ###Parse functions to get XML data into Data Frames

```
parseAuthors <- function (anAuthorListNode)
{
  newAuthor.df <- data.frame (author_id = integer(),
                          first_name = character(),
                          last_name = character(),
                          stringsAsFactors = F)
  n <- xmlSize(anAuthorListNode)

  for (m in 1:n)
  {
    anAuthor <- anAuthorListNode[[m]]
    first_name <- xmlValue(anAuthor[[2]])
    last_name <- xmlValue(anAuthor[[1]])

    newAuthor.df[m,2] <- first_name
    newAuthor.df[m,3] <- last_name

  }

  return(newAuthor.df)
}
```

```
parseIssues <- function (anArticle)
{
  newIssue.df <- Issue.df <- data.frame (issue_id = integer(),
                        journal_title = character(),
                          cited_medium = character(),
                          volume = integer(),
                          issue = integer(),
                          pub_date_year = integer(),
                          pub_date_month = character(),
                          stringsAsFactors = F)

  #Getting Cited Medium
  CMexp <-"string(./MedlineCitation/Article/Journal/JournalIssue/@CitedMedium)"
  tempCM <- xpathSApply(anArticle,CMexp)
  cited_medium <- tempCM

  #Getting Volume
  volexp <-"./MedlineCitation/Article/Journal/JournalIssue/Volume"
  tempVolume <- xpathSApply(anArticle,volexp)
  volume <- xmlValue(tempVolume)
  #volume <- strtoi(volume)

  # #Getting Issue
  issueexp <-"./MedlineCitation/Article/Journal/JournalIssue/Issue"
  tempIssue <- xpathSApply(anArticle,issueexp)
  issue <- xmlValue(tempIssue)
  #issue <- strtoi(issue)

  #Getting Title
  titleeexp <-"./MedlineCitation/Article/Journal/Title"
  tempTitle <- xpathSApply(anArticle,titleeexp)
  title <- xmlValue(tempTitle)

  #Getting PubDate information
  pubdateexp <- "./MedlineCitation/Article/Journal/JournalIssue/PubDate"
  tempPubDate <- xpathSApply(anArticle,pubdateexp)
  singlenode <- tempPubDate[[1]]
  childnodes <- xmlChildren(singlenode)

  year <- xmlValue(childnodes[1])
  month <- xmlValue(childnodes[2])

  newIssue.df[1,2] <- journal_title
  newIssue.df[1,3] <- cited_medium
  newIssue.df[1,4] <- volume
  newIssue.df[1,5] <- issue
  newIssue.df[1,6] <- year
  newIssue.df[1,7] <- month

  return(newIssue.df)
}
```

```
parseHistoryStatus <- function (aHistoryNode, i)
{
  #newHistory.df[m,2] <- i

  n <- xmlSize(aHistoryNode)

  newHistoryStatus.df <- data.frame (status_id = integer(),
                      pub_status = character(),
                      status_date = character(),
                      stringsAsFactors = F)

  for (m in 1:n)
  {
    aDateNode <- aHistoryNode[[m]]

    dateNodeAttributes <- xmlAttrs(aDateNode)
    pub_status <- as.character(dateNodeAttributes[1])

    hisYear <- as.character(xmlValue(aDateNode[[1]]))
    hisMonth <- as.character(xmlValue(aDateNode[[2]]))
    hisDay <- as.character(xmlValue(aDateNode[[3]]))

    hisDate <- paste(hisDay, hisMonth, hisYear, sep = "-")
    historystatusrow <- nrow(HistoryStatus.df) + 1

    newHistoryStatus.df[m,2] <- pub_status
    newHistoryStatus.df[m,3] <- hisDate

  }
  return(newHistoryStatus.df)
}
```

#Parsing the articles

```
r <- xmlRoot(xmlDOM)
#Iterate through the number of articles
for (i in 1:numArticles) #should be numArticles not 3
{
  #Get the next article node
  anArticle <- r[[i]]

  #Parse Author information, returns a data frame of the authors of an individual article
  authorNode <-"./MedlineCitation/Article/AuthorList/Author"
  xauth <- xpathSApply(anArticle,authorNode)
  newAuthor.df <- parseAuthors(xauth)

  #Adding the Authors to the Author.df
  tempAuthors <- Author.df
  Author.df <- rbind(tempAuthors, newAuthor.df)

  #get the title for later
  journalTitleNode <-"./MedlineCitation/Article/Journal/Title"
  tempTitle <- xpathSApply(anArticle,journalTitleNode)
  journal_title <- xmlValue(tempTitle)

  #Parse the issue node
  newIssue.df <- parseIssues(anArticle)

  #Adding the issues to the issue.df
  tempIssue <-Issue.df
  Issue.df <- rbind(tempIssue,newIssue.df)
```

```r
  #Parse History Status
  historyStatusNode <-"./PubmedData/History/PubMedPubDate"
  xhistorystatus <- xpathSApply(anArticle,historyStatusNode)
  newHistoryStatus.df <- parseHistoryStatus(xhistorystatus, i)

  #Adding the Hisotry Status to the historystatus.df
  tempHistoryStatus <-HistoryStatus.df
  HistoryStatus.df <- rbind(tempHistoryStatus,newHistoryStatus.df)

  #Getting article title Node
  titleNode <-"./MedlineCitation/Article/ArticleTitle"
  xtitle <- xpathSApply(anArticle,titleNode)
  artTitle <- as.character(xmlValue(xtitle[1]))

  #Adding article title and article_id to data frame
  Article.df[i,3] <- artTitle
  Article.df[i,1] <- i
}

#Delete Issue Duplicates
duplicateIssues <- Issue.df
Issue.df <- duplicateIssues[!duplicated(duplicateIssues),]

#Delete Author Duplicates
duplicateAuthors <- Author.df
Author.df <- duplicateAuthors[!duplicated(duplicateAuthors),]

#Delete History Duplicates
duplicateHistory <- History.df
History.df <- duplicateHistory[!duplicated(duplicateHistory),]

#Delete HistoryStatus Duplicates
duplicateHistoryStatus <- HistoryStatus.df
HistoryStatus.df <- duplicateHistoryStatus[!duplicated(duplicateHistoryStatus),]
```

###Clean the Data Frames #Here we add ID's, change date formats, and make sure data matches across data frames.

```r
num.authors <- nrow(Author.df)

for (r in 1:num.authors){
  Author.df$author_id[r] <- r
}
```

```r
num.historystatus <- nrow(HistoryStatus.df)

for (r in 1:num.historystatus){
  HistoryStatus.df$status_id[r] <- r
}
```

```r
num.issue <- nrow(Issue.df)

for (r in 1:num.issue){
  Issue.df$issue_id[r] <- r
}
# make columns int
cols.num <- c("volume","issue")
Issue.df[cols.num] <- sapply(Issue.df[cols.num],as.integer)
sapply(Issue.df, class)
```

```
##       issue_id  journal_title   cited_medium         volume          issue
##      "integer"    "character"    "character"      "integer"      "integer"
##  pub_date_year pub_date_month
##    "character"    "character"
```

```r
# years vector
year <- c()
for (r in 1:num.issue){
  year <- c(year, substr(Issue.df$pub_date_year[r],1,4))
}

#mont vector
month <- c()
for (r in 1:num.issue){
  if(is.na(Issue.df$pub_date_month[r])){
    m <- substr(Issue.df$pub_date_year[r],6,8)
  }else{
    m <- Issue.df$pub_date_month[r]
  }
  # month to numbers
  if(m == 'Jan'){
    m <- "1"
  }else if( m == 'Feb'){
    m <- "2"
  }else if( m == 'Mar'){
    m <- "3"
  }else if( m == 'Apr'){
    m <- "4"
  }else if( m == 'May'){
    m <- "5"
  }else if( m == 'Jun'){
    m <- "6"
  }else if( m == 'Jul'){
    m <- "7"
  }else if( m == 'Aug'){
    m <- "8"
  }else if( m == 'Sep'){
    m <- "9"
  }else if( m == 'Oct'){
    m <- "10"
  }else if( m == 'Nov'){
    m <- "11"
  }else if( m == 'Dec'){
    m <- "12"
  }else{
    m <- "1"
  }
  month <- c(month, m)
}



Issue.df$pub_date <- NA

for(i in 1:num.issue){
  d <- paste(year[i], month[i], sep="-")
  d <- paste(d, "-01", sep="")
  Issue.df$pub_date[i] <- d
}

Issue.df$pub_date <- as.Date(Issue.df$pub_date,
                               format = "%Y-%m-%d")
```

```r
Issue.df$pub_date_month <- NULL
Issue.df$pub_date_year <- NULL
print(Issue.df)
```

```
##    issue_id
## 1         1
## 2         2
## 3         3
## 4         4
## 5         5
## 6         6
## 7         7
## 8         8
## 9         9
## 10       10
## 11       11
## 12       12
## 13       13
## 14       14
## 15       15
## 16       16
## 17       17
## 18       18
## 19       19
##                                                                 journal_title
## 1  HSS journal : the musculoskeletal journal of Hospital for Special Surgery
## 2                                                              Psychosomatics
## 3                                   Stroke; a journal of cerebral circulation
## 4                                       Regional anesthesia and pain medicine
## 5                   Seizure : the journal of the British Epilepsy Association
## 6                                  Clinical orthopaedics and related research
## 7                                                 The Journal of arthroplasty
## 8                                                              Anesthesiology
## 9                                                         Pediatric radiology
## 10                                          Diseases of the colon and rectum
## 11                                            Journal of clinical anesthesia
## 12                                                                  PloS one
## 13                                     Regional anesthesia and pain medicine
## 14                                               The Journal of arthroplasty
## 15                                                                     Spine
## 16                                                                    Cancer
## 17                                                         BJU international
## 18                                        Journal of intensive care medicine
## 19                                                                     Spine
##    cited_medium volume issue   pub_date
## 1         Print      8     2 2012-07-01
## 2      Internet     54     2 2013-03-01
## 3      Internet     43    11 2012-11-01
## 4      Internet     37     6 2012-11-01
## 5      Internet     22     1 2013-01-01
## 6      Internet    471     1 2013-01-01
## 7      Internet     27    10 2012-12-01
## 8      Internet    117     1 2012-07-01
## 9      Internet     42     8 2012-08-01
## 10     Internet     55     4 2012-04-01
## 11     Internet     24     2 2012-03-01
## 12     Internet      7     1 2012-01-01
## 13     Internet     37     1 2012-01-01
## 14     Internet     27     6 2012-06-01
## 15     Internet     37    11 2012-05-01
## 16     Internet    118    12 2012-06-01
## 17     Internet    109     5 2012-03-01
## 18     Internet     27     5 2012-09-01
## 19     Internet     37     3 2012-02-01
```

```r
#New root
root <- xmlRoot(xmlDOM)

#go through each article
for ( i in 1:numArticles){

  #Gets an article
  anArticle <- root[[i]]

  #Getting Volume
  volexp <-"./MedlineCitation/Article/Journal/JournalIssue/Volume"
  tempVolume <- xpathSApply(anArticle,volexp)
  volume <- strtoi(xmlValue(tempVolume))

  #Getting Issue
  issueexp <-"./MedlineCitation/Article/Journal/JournalIssue/Issue"
  tempIssue <- xpathSApply(anArticle,issueexp)
  issue <- strtoi(xmlValue(tempIssue))

  #Getting Title
  titleeexp <-"./MedlineCitation/Article/Journal/Title"
  tempTitle <- xpathSApply(anArticle,titleeexp)
  title <- xmlValue(tempTitle)

  #Getting article title Node
  titleNode <-"./MedlineCitation/Article/ArticleTitle"
  xtitle <- xpathSApply(anArticle,titleNode)
  artTitle <- as.character(xmlValue(xtitle[1]))
  #Where article title corresponds to issue, volume, and journal add an id
  for(i in 1:nrow(Article.df)){
    if(Article.df$article_title[i] == artTitle){
      for(j in 1:nrow(Issue.df)){
        if(Issue.df$journal_title[j] == title && Issue.df$issue[j] == issue
           && Issue.df$volume[j] == volume ){
          Article.df$issue_id[i] = Issue.df$issue_id[j]
        }
      }
    }
  }

}
```

```
#New root
root <- xmlRoot(xmlDOM)

#go through each article
for ( i in 1:numArticles){

  #Gets an article
  anArticle <- root[[i]]

  #Parse Author list node
  authorNode <-"./MedlineCitation/Article/AuthorList/Author"
  xauth <- xpathApply(anArticle,authorNode)

  #size of authorlist
  n <- xmlSize(xauth)

  #finds the first name/last name of each author
  for (m in 1:n)
  {
    anAuthor <- xauth[[m]]
    first_name <- xmlValue(anAuthor[[2]])
    last_name <- xmlValue(anAuthor[[1]])

    #if the first name and last name match whats in the Author.df it adds the author_id and article_id
    for( j in 1:num.authors) {
      if (Author.df$first_name[j] == first_name && Author.df$last_name[j] == last_name) {
        val <- Author.df$author_id[j]

        authorjoinrow <- nrow(AuthorJoin.df) + 1
        AuthorJoin.df[authorjoinrow,2] <- i
        AuthorJoin.df[authorjoinrow,1] <- val
      }
    }
  }
}
```

```
#New root
root <- xmlRoot(xmlDOM)
```

```r
#go through each article
for ( i in 1:numArticles){

  #Gets an article
  anArticle <- root[[i]]

  #Parse History information, returns a data frame of the history dates of an individual article
  historyNode <-"./PubmedData/History/PubMedPubDate"
  xhistory <- xpathSApply(anArticle,historyNode)

  #size of historyStatus
  n <- xmlSize(xhistory)

  #finds the first name/last name of each author
  for (m in 1:n)
  {
    aDateNode <- xhistory[[m]]

    dateNodeAttributes <- xmlAttrs(aDateNode)
    pub_status <- as.character(dateNodeAttributes[1])

    hisYear <- as.character(xmlValue(aDateNode[[1]]))
    hisMonth <- as.character(xmlValue(aDateNode[[2]]))
    hisDay <- as.character(xmlValue(aDateNode[[3]]))

    hisDate <- paste(hisDay, hisMonth, hisYear, sep = "-")

    #if the first name and last name match whats in the Author.df it adds the author_id and article_id
    for( j in 1:num.historystatus) {
      if (HistoryStatus.df$pub_status[j] == pub_status && HistoryStatus.df$status_date[j] == hisDate) {
        val <- HistoryStatus.df$status_id[j]

        historystatusjoinrow <- nrow(History.df) + 1
        History.df[historystatusjoinrow,1] <- val
        History.df[historystatusjoinrow,2] <- i
      }
    }
  }
}
```

```r
head(Article.df, 5)
```

```
##   article_id issue_id
## 1          1        1
## 2          2        2
## 3          3        3
## 4          4        4
## 5          5        5
##
## 1                                  Regional anesthesia for children undergoing orthopedic ambulatory
## 2 Demographics and perioperative outcome in patients with depression and anxiety undergoing total jo
## 3                Cerebrovascular reserve and stroke risk in patients with carotid stenosis or occlus
## 4    Comparative perioperative outcomes associated with neuraxial versus general anesthesia for simul
## 5                                  Vagus nerve stimulation vs. corpus callosotomy in the treatment o
```

```r
head(AuthorJoin.df, 5)
```

```
##   author_id article_id
## 1         1          1
## 2         2          1
## 3         3          1
## 4         4          1
## 5         5          2
```

```r
head(Author.df, 5)
```

```
##   author_id first_name  last_name
## 1         1     Cassie        Kuo
## 2         2     Alison    Edwards
## 3         3      Madhu   Mazumdar
## 4         4 Stavros G Memtsoudis
## 5         5    Ottokar   Stundner
```

```r
head(History.df, 5)
```

```
##   status_id article_id
## 1         1          1
## 2         2          1
## 3         3          1
## 4         4          1
## 5         5          1
```

```r
head(HistoryStatus.df, 5)
```

```
##   status_id pub_status status_date
## 1         1   received   15-1-2012
## 2         2   accepted   16-4-2012
## 3         3   epublish   20-6-2012
## 4         4     entrez   23-7-2013
## 5         5     pubmed   23-7-2013
```

```r
head(Issue.df, 5)
```

```
##   issue_id
## 1        1
## 2        2
## 3        3
## 4        4
## 5        5
##                                                                journal_title
## 1 HSS journal : the musculoskeletal journal of Hospital for Special Surgery
## 2                                                             Psychosomatics
## 3                                  Stroke; a journal of cerebral circulation
## 4                                      Regional anesthesia and pain medicine
## 5                    Seizure : the journal of the British Epilepsy Association
##   cited_medium volume issue   pub_date
## 1        Print      8     2 2012-07-01
## 2     Internet     54     2 2013-03-01
## 3     Internet     43    11 2012-11-01
## 4     Internet     37     6 2012-11-01
## 5     Internet     22     1 2013-01-01
```

###Write data to SQL tables

```
dbWriteTable(mydb, "Article", Article.df, overwrite = T, row.names = F)
```

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

```
dbWriteTable(mydb, "AuthorJoin", AuthorJoin.df, overwrite = T, row.names = F)
```

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

```
dbWriteTable(mydb, "Author", Author.df, overwrite = T, row.names = F)
```

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

```
dbWriteTable(mydb, "History", History.df, overwrite = T, row.names = F)
```

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

```
dbWriteTable(mydb, "HistoryStatus", HistoryStatus.df, overwrite = T, row.names = F)
```

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

```
dbWriteTable(mydb, "JournalIssue", Issue.df, overwrite = T, row.names = F)
```

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

#Part 2, Question 1

```
SELECT * FROM Article LIMIT 5;
```

Table 1: 5 records

article_id	issue_id	article_title
1	1	Regional anesthesia for children undergoing orthopedic ambulatory surgeries in the United States, 1996-2006.
2	2	Demographics and perioperative outcome in patients with depression and anxiety undergoing total joint arthroplasty: a population-based study.
3	3	Cerebrovascular reserve and stroke risk in patients with carotid stenosis or occlusion: a systematic review and meta-analysis.
4	4	Comparative perioperative outcomes associated with neuraxial versus general anesthesia for simultaneous bilateral total knee arthroplasty.
5	5	Vagus nerve stimulation vs. corpus callosotomy in the treatment of Lennox-Gastaut syndrome: a meta-analysis.

```
SELECT * FROM AuthorJoin LIMIT 5;
```

Table 2: 5 records

author_id	article_id
1	1
2	1
3	1
4	1
5	2

```
SELECT * FROM Author LIMIT 5;
```

Table 3: 5 records

author_id	first_name	last_name
1	Cassie	Kuo
2	Alison	Edwards
3	Madhu	Mazumdar
4	Stavros G	Memtsoudis
5	Ottokar	Stundner

```sql
SELECT * FROM History LIMIT 5;
```

Table 4: 5 records

status_id	article_id
1	1
2	1
3	1
4	1
5	1

```sql
SELECT * FROM HistoryStatus LIMIT 5;
```

Table 5: 5 records

status_id	pub_status	status_date
1	received	15-1-2012
2	accepted	16-4-2012
3	epublish	20-6-2012
4	entrez	23-7-2013
5	pubmed	23-7-2013

```sql
SELECT * FROM JournalIssue LIMIT 5;
```

Table 6: 5 records

issue_id	journal_title	cited_medium	volume	issue	pub_date
1	HSS journal : the musculoskeletal journal of Hospital for Special Surgery	Print	8	2	2012-07-01
2	Psychosomatics	Internet	54	2	2013-03-01
3	Stroke; a journal of cerebral circulation	Internet	43	11	2012-11-01
4	Regional anesthesia and pain medicine	Internet	37	6	2012-11-01
5	Seizure : the journal of the British Epilepsy Association	Internet	22	1	2013-01-01

#Drop tables for running full file

```
DROP SCHEMA IF EXISTS starschema
```

```
DROP TABLE IF EXISTS starschema.AuthorDimension
```

```
DROP TABLE IF EXISTS starschema.JournalDimension
```

```
DROP TABLE IF EXISTS starschema.TimeDimension
```

```
DROP TABLE IF EXISTS starschema.HistoryDimension
```

```
DROP TABLE IF EXISTS starschema.ArticleFactTable
```

```
DROP TABLE IF EXISTS starschema.ArticleSummary
```

```
DROP TABLE IF EXISTS starschema.ByTime
```

```
DROP TABLE IF EXISTS starschema.ByJournal
```

```
DROP TABLE IF EXISTS starschema.ByAuthor
```

```
CREATE SCHEMA IF NOT EXISTS starschema
```

###Creates the Author dimension table

```
CREATE TABLE IF NOT EXISTS starschema.AuthorDimension
  AS SELECT Author.author_id as AuthorDim_id,
            Author.first_name,
            Author.last_name,
            Article.article_id
  FROM dbpracticum2.Author
  JOIN dbpracticum2.AuthorJoin USING(author_id)
  JOIN dbpracticum2.Article USING(article_id);
```

###Creates the Journal dimension table

```
CREATE TABLE IF NOT EXISTS starschema.JournalDimension (
    issue_id INT PRIMARY KEY,
    journal_title TEXT NOT NULL,
    issue INT NOT NULL,
    volume INT NOT NULL,
    pub_date TEXT NOT NULL,
    cited_medium TEXT NOT NULL,
    article_id INT NOT NULL
);
```

```
INSERT INTO starschema.JournalDimension (issue_id, journal_title, issue, volume, pub_date, cited_medium
SELECT JournalIssue.issue_id,
            JournalIssue.journal_title,
            JournalIssue.issue,
            JournalIssue.volume,
            JournalIssue.pub_date,
            JournalIssue.cited_medium,
            Article.article_id
  FROM dbpracticum2.JournalIssue
  JOIN dbpracticum2.Article USING(issue_id);
```

###Creates the History dimension table

```
CREATE TABLE IF NOT EXISTS starschema.HistoryDimension
  AS SELECT HistoryStatus.status_id as history_id,
            HistoryStatus.status_date,
            HistoryStatus.pub_status,
            Article.article_id
  FROM dbpracticum2.HistoryStatus
  JOIN dbpracticum2.History USING(status_id)
  JOIN dbpracticum2.Article USING(article_id);
```

###creates the fact table

```
CREATE TABLE IF NOT EXISTS starschema.ArticleFactTable (
  article_id INT NOT NULL PRIMARY KEY,
  article_title TEXT NOT NULL,
  issue_id INT NOT NULL
);
```

###inserts into the fact table

```
INSERT INTO starschema.ArticleFactTable(article_id, article_title, issue_id)
SELECT Article.article_id, Article.article_title, JournalIssue.issue_id
  FROM dbpracticum2.JournalIssue
  JOIN dbpracticum2.Article USING(issue_id);
```

###Creates the time dimension table

```
CREATE TABLE IF NOT EXISTS starschema.TimeDimension (
  time_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  charDate TEXT NOT NULL,
  year INT NOT NULL,
  month INT NOT NULL,
  quarter INT NOT NULL
);
```

###Inserts into time dimension table

```
INSERT INTO starschema.TimeDimension (charDate, year, month, quarter)
  SELECT DISTINCT pub_date as charDate,
  CAST(SUBSTRING(pub_date, 1,4) AS UNSIGNED) as year,
  CAST(SUBSTRING(pub_date, 6,2) AS UNSIGNED) as month,
  0 as quarter
  from starschema.JournalDimension;
```

###Updates quarter values for time dimension table

```
UPDATE starschema.TimeDimension SET quarter =
  CASE
    WHEN month <= 3 THEN 1
    WHEN month <= 6 THEN 2
    WHEN month <= 9 THEN 3
    ELSE 4
  END
WHERE quarter = 0;
```

###Adds time_id to journal dimension table

```
ALTER TABLE starschema.JournalDimension
  ADD time_id INT;
```

###sets the time_id from journal dimension to the time dimension equivalent

```
UPDATE starschema.JournalDimension
  SET time_id = (Select time_id from starschema.TimeDimension WHERE JournalDimension.pub_date = TimeDim
```

###Drops old pub_date column

```
ALTER TABLE starschema.JournalDimension
  DROP COLUMN pub_date;
```

```
SELECT * FROM starschema.JournalDimension
```

Table 7: Displaying records 1 - 10

issue_id	journal_title	issue	volume	cited_medium	article_id	time_id
1	HSS journal : the musculoskeletal journal of Hospital for Special Surgery	2	8	Print	1	1
2	Psychosomatics	2	54	Internet	2	2
3	Stroke; a journal of cerebral circulation	11	43	Internet	3	3
4	Regional anesthesia and pain medicine	6	37	Internet	4	3
5	Seizure : the journal of the British Epilepsy Association	1	22	Internet	5	4
6	Clinical orthopaedics and related research	1	471	Internet	6	4
7	The Journal of arthroplasty	10	27	Internet	7	5
8	Anesthesiology	1	117	Internet	8	1
9	Pediatric radiology	8	42	Internet	9	6
10	Diseases of the colon and rectum	4	55	Internet	10	7

```
SELECT * FROM starschema.ArticleFactTable
```

Table 8: Displaying records 1 - 10

article_id	article_title	issue_id
1	Regional anesthesia for children undergoing orthopedic ambulatory surgeries in the United States, 1996-2006.	1
2	Demographics and perioperative outcome in patients with depression and anxiety undergoing total joint arthroplasty: a population-based study.	2
3	Cerebrovascular reserve and stroke risk in patients with carotid stenosis or occlusion: a systematic review and meta-analysis.	3
4	Comparative perioperative outcomes associated with neuraxial versus general anesthesia for simultaneous bilateral total knee arthroplasty.	4
5	Vagus nerve stimulation vs. corpus callosotomy in the treatment of Lennox-Gastaut syndrome: a meta-analysis.	5
6	Have bilateral total knee arthroplasties become safer? A population-based trend analysis.	6
7	The metabolic syndrome in patients undergoing knee and hip arthroplasty: trends and in-hospital outcomes in the United States.	7
8	Utilization of critical care services among patients undergoing total hip and knee arthroplasty: epidemiology and risk factors.	8
9	Visualization of the normal appendix with MR enterography in children.	9
10	FDG-PET assessment of rectal cancer response to neoadjuvant chemoradiotherapy is not associated with long-term prognosis: a prospective evaluation.	10

##Part 2, Question 2 ##Summary Fact Table

```
CREATE TABLE IF NOT EXISTS starschema.ArticlesSummary(
  article_id INT NOT NULL,
  time_id INT NOT NULL,
  journal_title TEXT NOT NULL,
  author_id INT NOT NULL
);
```

###Creates the byAuthor Table

```
CREATE TABLE IF NOT EXISTS starschema.byAuthor(
  author_id INT PRIMARY KEY,
  first_name TEXT NOT NULL,
  last_name TEXT NOT NULL
);
```

###Creates the byTime table

```
CREATE TABLE IF NOT EXISTS starschema.byTime(
  time_id INT PRIMARY KEY,
  year INT NOT NULL,
  month INT NOT NULL,
  quarter INT NOT NULL
);
```

###Creates the byJournal table

```
CREATE TABLE IF NOT EXISTS starschema.byJournal(
  journal_id INT AUTO_INCREMENT PRIMARY KEY,
  journal_title TEXT NOT NULL
);
```

###Inserts into the byJournal table

```
INSERT INTO starschema.byJournal(journal_title)
SELECT DISTINCT JournalDimension.journal_title
  FROM starschema.JournalDimension;
```

###Inserts into the byAuthor table

```
INSERT INTO starschema.byAuthor(author_id, first_name, last_name)
SELECT DISTINCT(AuthorDimension.authorDim_id) AS author_id,
AuthorDimension.first_name AS author_first,
AuthorDimension.last_name AS author_last
FROM starschema.AuthorDimension;
```

###Inserts into the byTime table

```
INSERT INTO starschema.byTime(time_id, year, month, quarter)
SELECT TimeDimension.time_id AS time_id,
TimeDimension.year AS year,
TimeDimension.month AS month,
TimeDimension.quarter AS quarter
  FROM starschema.TimeDimension;
```

```
INSERT INTO starschema.ArticlesSummary(article_id, time_id, journal_title, author_id)
SELECT starschema.ArticleFactTable.article_id,
  starschema.TimeDimension.time_id,
  starschema.JournalDimension.journal_title,
  starschema.AuthorDimension.AuthorDim_id
  FROM starschema.TimeDimension
  JOIN starschema.JournalDimension USING(time_id)
  JOIN starschema.ArticleFactTable USING(issue_id)
  JOIN starschema.AuthorDimension ON starschema.AuthorDimension.article_id = starschema.ArticleFactTabl
  GROUP BY time_id, journal_title,AuthorDim_id;
```

```
select * from starschema.ArticlesSummary LIMIT 10
```

Table 9: Displaying records 1 - 10

article_id	time_id	journal_title	author_id
1	1	HSS journal : the musculoskeletal journal of Hospital for Special Surgery	1
1	1	HSS journal : the musculoskeletal journal of Hospital for Special Surgery	2
19	13	Spine	3
18	12	Journal of intensive care medicine	3
17	8	BJU international	3
16	10	Cancer	3
15	11	Spine	3
14	10	The Journal of arthroplasty	3
13	9	Regional anesthesia and pain medicine	3
12	9	PloS one	3

###Adds time_id to journal dimension table

```
ALTER TABLE starschema.ArticlesSummary
  ADD journal_id INT;
```

###sets the time_id from journal dimension to the time dimension equivalent

```
UPDATE starschema.ArticlesSummary
  SET journal_id = (Select journal_id from starschema.byJournal WHERE starschema.ArticlesSummary.journal
```

###Drops old pub_date column

```
ALTER TABLE starschema.ArticlesSummary
  DROP COLUMN journal_title;
```

###Part 3 Question 1 ##Exploring Publication patterns ###Grouping by quarter It seems that quarter 1 (jan, feb, march) is the most productive quarter.

```
select quarter, count(distinct article_id) from starschema.ArticlesSummary
  JOIN starschema.byTime USING(time_id)
  GROUP BY quarter
  order by quarter
```

Table 10: 4 records

quarter	count(distinct article_id)
1	8
2	4
3	4
4	3

###Article publication by month Let's break it down further.

```sql
select month, count(distinct article_id) as articles
  from starschema.ArticlesSummary
  JOIN starschema.byTime USING(time_id)
  GROUP BY month
  order by month
```

Table 11: Displaying records 1 - 10

month	articles
1	4
2	1
3	3
4	1
5	1
6	2
7	2
8	1
9	1
11	2

###Most productive year When did people publish?

```sql
select year, count(distinct article_id) as 'Articles_Published'
  from starschema.ArticlesSummary
  JOIN starschema.byTime USING(time_id)
  GROUP BY year
  order by Articles_Published DESC
```

Table 12: 2 records

year	Articles_Published
2012	16
2013	3

###Collaboration by quarter How many authors per article in each quarter?

```sql
select quarter, count(author_id)/count(distinct article_id) as collab
  from starschema.ArticlesSummary
  JOIN starschema.byTime USING(time_id)
  JOIN starschema.byAuthor USING(author_id)
  GROUP BY quarter
  order by quarter
```

Table 13: 4 records

quarter	collab
1	7.1250
2	8.0000
3	5.5000
4	8.3333

###Top 5 most published authors Who is publishing the most?

```sql
select first_name, last_name, count(distinct article_id) as 'Articles_Published' from starschema.Articl
  JOIN starschema.byAuthor USING(author_id)
  GROUP BY author_id
  order by Articles_Published DESC
  LIMIT 5
```

Table 14: 5 records

first_name	last_name	Articles_Published
Madhu	Mazumdar	19
Stavros G	Memtsoudis	12
Yan	Ma	7
Ya Lin	Chiu	5
Ottokar	Stundner	4

#Top 5 most published authors by quarter Let's explore top publishers.

```sql
select first_name, last_name, quarter, count(distinct article_id) as 'Articles_Published' from starschem
  JOIN starschema.byAuthor USING(author_id)
  JOIN starschema.byTime USING(time_id)
  GROUP BY quarter
  order by Articles_Published DESC
  LIMIT 5
```

Table 15: 4 records

first_name	last_name	quarter	Articles_Published
Madhu	Mazumdar	1	8
Madhu	Mazumdar	2	4
Cassie	Kuo	3	4
Madhu	Mazumdar	4	3

#Authors published by unique journals Who is publishing broadly?

```sql
select first_name, last_name, count(distinct journal_id) as 'Authors_by_Unique_Journals' from starschema
  JOIN starschema.byJournal USING(journal_id)
  JOIN starschema.byAuthor USING(author_id)
  GROUP BY author_id
  order by Authors_by_Unique_Journals DESC
  LIMIT 5
```

Table 16: 5 records

first_name	last_name	Authors_by_Unique_Journals
Madhu	Mazumdar	16
Stavros G	Memtsoudis	9
Yan	Ma	5
Ottokar	Stundner	4
Ya Lin	Chiu	4

#Journals with the most published articles Which journals are hot?

```
select journal_title, count(distinct article_id) as 'Journal_Articles_Published' from starschema.Article
  JOIN starschema.byJournal USING(journal_id)
  GROUP BY journal_id
  order by Journal_Articles_Published DESC
  LIMIT 5
```

Table 17: 5 records

journal_title	Journal_Articles_Published
Regional anesthesia and pain medicine	2
Spine	2
The Journal of arthroplasty	2
Cancer	1
PloS one	1

```
dbDisconnect(mydb)
```

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