MAESTRÍA EN ESTADÍSTICA APLICADA/ MANEJO DE DATOS

Unidad 3: Recopilación, integración y manipulación y almacenamiento de datos

3.3. Selección y recuperación de datos (Queries SQL) de tablas relacionadas, por medio de comandos del software estadístico.

En la sección 3.1 se vio como establecer una conexión con el DBMS de MySQL, para la manipulación y procesamiento de los datos mediante SQL, utilizaremos nuevamente la conexión por usuario DSN.

Sintaxis de conexión:

```
#install.packages("RODBC")                                          #inst
alación de paquete RODBC
library(RODBC)                                                      #refe
rencia a la librería RODBC
con<-odbcConnect("MySQLconexion1", uid = "root", pwd="Matias2710")  #Esta
blecer la conexión con el DBMS
```

En esta sección nuevamente se trabajará con la función sqldf (), mediante la cual es posible utilizar la sintaxis convencional de SQL en el entorno de R. Para el uso de la función sqldf, vamos a requerir previamente que los datos se almacén en un Dataframe, se trabajará con los registros de la tabla country, city y countrylanguage de la base de datos "world" de MySQL.

```
library(sqldf)

## Loading required package: gsubfn

## Loading required package: proto

## Loading required package: RSQLite

df_country<-sqlQuery(con,"Select * from country")                   #
Ejecutar una sentencia SQL
df_city<-sqlQuery(con,"Select * from city")
df_language<-sqlQuery(con,"Select * from countrylanguage")
```

Conozcamos el tipo de objeto y la estructura de df_city y df_language:

```
str(df_city)      #str nos indica la estructura del objeto

## 'data.frame':    4079 obs. of  5 variables:
##  $ ID         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Name       : Factor w/ 4001 levels "[San CristÃ³bal de] la Laguna",
..: 1641 2860 1341 2193 152 2990 1268 3710 995 3567 ...
##  $ CountryCode: Factor w/ 232 levels "ABW","AFG","AGO",..: 2 2 2 2 154
154 154 154 154 154 ...
##  $ District   : Factor w/ 1367 levels "","Â–","Ã–rebros lÃ¤n",..: 580
```

```
995 495 149 875 1365 1365 1277 874 874 ...
##  $ Population : int  1780000 237500 186800 127800 731200 593321 440900
234323 201843 193238 ...

str(df_language)

## 'data.frame':    984 obs. of  4 variables:
##  $ CountryCode: Factor w/ 233 levels "ABW","AFG","AGO",..: 1 1 1 1 2 2
2 2 2 3 ...
##  $ Language   : Factor w/ 457 levels "[South]Mande",..: 106 111 330 39
0 32 97 332 430 437 12 ...
##  $ IsOfficial : logi  TRUE FALSE FALSE FALSE FALSE TRUE ...
##  $ Percentage : num  5.3 9.5 76.7 7.4 0.9 ...
```

Es natural encontrar que los datos se almacenan en diferentes fuentes o tablas de datos, las cuales se relacionan por medio de claves foráneas; en esta ocasión se utilizará la función sqldf() para crear consultas de datos que extraigan registros de varias tablas relacionadas.

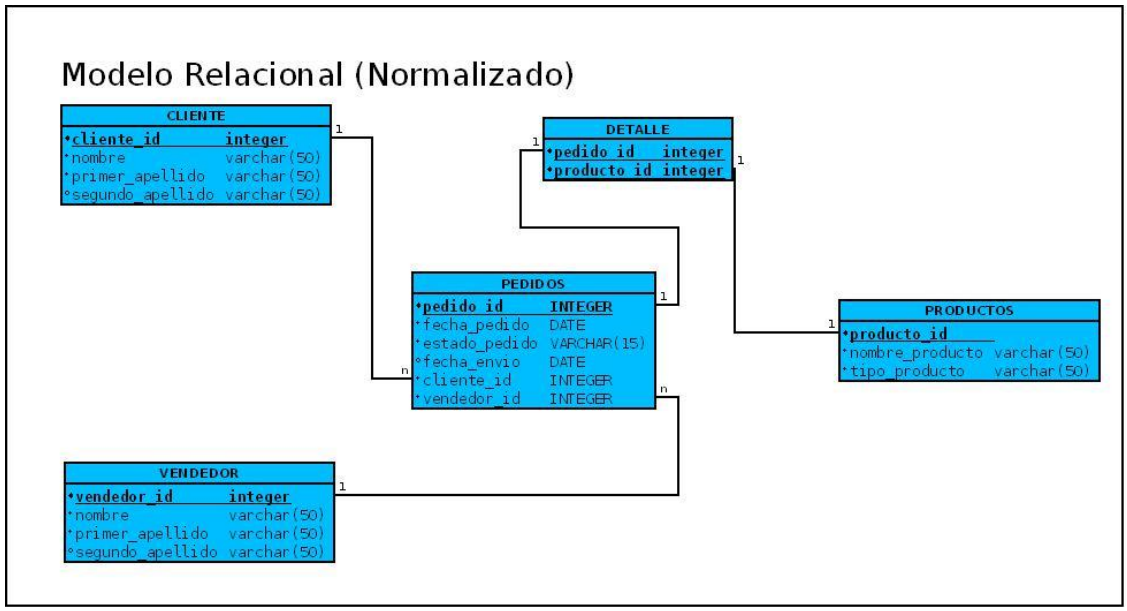

Diagrama Relacional

JOIN

Los JOINs en SQL sirven para combinar filas de dos o más tablas basándose en un campo común entre ellas, devolviendo por tanto datos de diferentes tablas. Un JOIN se produce cuando dos o más tablas se juntan en una sentencia SQL. Los JOINs más utilizados en SQL son:

1) INNER JOIN: Devuelve todas las filas cuando hay al menos una coincidencia en ambas tablas.
2) LEFT JOIN: Devuelve todas las filas de la tabla de la izquierda, y las filas coincidentes de la tabla de la derecha.

3) RIGHT JOIN: Devuelve todas las filas de la tabla de la derecha, y las filas coincidentes de la tabla de la izquierda.
4) OUTER JOIN: Devuelve todas las filas de las dos tablas, la izquierda y la derecha. Tambien se llama FULL OUTER JOIN.

INNER JOIN

INNER JOIN selecciona todas las filas de las dos columnas siempre y cuando haya una coincidencia entre las columnas en ambas tablas. Es el tipo de JOIN más común. Por ejemplo, obtenga un listado con los datos de ciudad, población de ciudad y continente.

```
df_res11<-sqldf("SELECT A.Name, A.Population, B.Continent FROM df_city A
INNER JOIN df_country B on A.CountryCode=B.Code ", connection=NULL)
head(df_res11,15)
```

```
##              Name Population Continent
## 1           Kabul    1780000      Asia
## 2        Qandahar     237500      Asia
## 3           Herat     186800      Asia
## 4  Mazar-e-Sharif     127800      Asia
## 5       Amsterdam     731200    Europe
## 6       Rotterdam     593321    Europe
## 7            Haag     440900    Europe
## 8         Utrecht     234323    Europe
## 9       Eindhoven     201843    Europe
## 10        Tilburg     193238    Europe
## 11      Groningen     172701    Europe
## 12          Breda     160398    Europe
## 13      Apeldoorn     153491    Europe
## 14       Nijmegen     152463    Europe
## 15       Enschede     149544    Europe
```

Actividad 1: *Obtenga un listado con los siguientes datos: Ciudad, País, Continente y Language.*

Actividad 2: *Obtenga un listado con los siguientes datos: Ciudad, País, Languague; considere seleccionar solo aquellos países en donde el porcentaje de habitantes que hablan el respectivo lenguaje sea mayor al 70%.*

LEFT JOIN

LEFT JOIN mantiene todas las filas de la tabla izquierda (la tabla1). Las filas de la tabla derecha se mostrarán si hay una coincidencia con las de la izquierda. Si existen valores en la tabla izquierda pero no en la tabla derecha, ésta mostrará null. Para el siguiente ejemplo se citan tablas que no pertenecen a la Base de Datos “world”, por lo tanto no se usará la conexión a la base de datos. Supongamos que se dispone de las tablas Clientes (df_clientes) y Pedidos (df_pedidos); se nos solicita conocer el listado de clientes con su

respectivo código de pedido, debe incluir en el listado a los clientes que no hayan realizado pedido alguno, ordene el listado por nombre del cliente.

Sintaxis:

df_res<-sqldf("SELECT Clientes.NombreCliente, Pedidos.PedidoID FROM Clientes LEFT JOIN Pedidos ON Clientes.ClienteID=Pedidos.ClienteID ORDER BY Clientes.NombreCliente", connection=NULL)

Resultado:

NombreCliente	PedidoID
Ebbe Therese	236
Lydia Roderic	235
Marco Lambert	(null)
Sofie Mariona	234
Sofie Mariona	237

Ahora vemos que se muestran todas las filas de la tabla Clientes, que es la tabla de la izquierda, tantas veces como haya coincidencias con el lado derecho. Marco Lambert no ha realizado ningún pedido, por lo que se muestra null.

UNION

```
df_res11<-sqldf("SELECT A.Name as 'Ciudad', A.Population as 'PoblacionCiu
dad', B.Name as 'Pais' FROM df_city A
INNER JOIN df_country B on A.CountryCode=B.Code and B.Continent='Europe'
and B.population>50000000", connection=NULL)
df_res12<-sqldf("SELECT A.Name as 'Ciudad', A.Population as 'PoblacionCiu
dad', B.Name as 'Pais' FROM df_city A
INNER JOIN df_country B on A.CountryCode=B.Code and B.Continent='South Am
erica' and B.population>50000000 ", connection=NULL)
df_res13<-sqldf("select Ciudad,PoblacionCiudad,Pais from df_res11
                UNION
                select Ciudad,PoblacionCiudad,Pais from df_res12", connec
tion=NULL)
head(df_res13,20)
```

```
##             Ciudad PoblacionCiudad              Pais
## 1           Aachen          243825           Germany
## 2       Abaetetuba          111258            Brazil
## 3           Abakan          169200 Russian Federation
## 4         Aberdeen          213070    United Kingdom
## 5  Aix-en-Provence          134222            France
## 6       Alagoinhas          126820            Brazil
## 7      Alessandria           90289             Italy
## 8       Almetjevsk          140700 Russian Federation
## 9        AltÂševsk          119000           Ukraine
## 10         Alvorada          175574            Brazil
```

```
## 11       Americana          177409             Brazil
## 12          Amiens          135501             France
## 13      Ananindeua          400940             Brazil
## 14          Ancona           98329              Italy
## 15          Andria           94443              Italy
## 16         Angarsk          264700 Russian Federation
## 17          Angers          151279             France
## 18  Angra dos Reis           96864             Brazil
## 19 Anzero-Sudzensk           96100 Russian Federation
## 20       AnÃ¡polis          282197             Brazil
```

Actividad 3 *Tomando como referencia el lenguaje oficial de cada país, determine para cada ciudad la cantidad de habitantes que habla dicho idioma.*