assignment2a

Dan Gueco, 300449479

07/09/2020

```
library(DBI)
library(RSQLite)
```

Question 1

a.Write R code to create and connect to a new SQLite database with storage file cricket.sqlite

ANS:

```
test_conn <- dbConnect(RSQLite::SQLite(), "cricket.sqlite")
test_conn
```

```
## <SQLiteConnection>
##   Path: C:\Users\GGPC\Desktop\DATA202\cricket.sqlite
##   Extensions: TRUE
```

Question 2

Write R code to read the seven data files into R. Combine the three outcomes data files (T20, ODI and Test) into a single R object, and report the numbers of rows it has.

ANS: The number of rows of the three combined files is 32,097

```
players<-read.csv("players.csv", stringsAsFactors=FALSE)
teams<-read.csv("teams.csv", stringsAsFactors=FALSE)
grounds<-read.csv("grounds.csv", stringsAsFactors=FALSE)
innings<-read.csv("innings.csv", stringsAsFactors=FALSE)
ODI<-read.csv("wcricket-ODI.csv", stringsAsFactors=FALSE)
T20<-read.csv("wcricket-T20.csv", stringsAsFactors=FALSE)
Test<-read.csv("wcricket-Test.csv", stringsAsFactors=FALSE)

#Combine the three outcomes data files (T20, ODI and Test) into a single R object
m = rbind(ODI,T20)
wcricket = rbind(m,Test)
```

Question 3a

The dates in the innings.csv file are stored as characters, but are inconsistent. The ODI entries have a different format to the others.

a.Create a new column innings.date which has character type, and which has a single consistent format. **ANS:**

```
#Create a new column innings.date which has character type, and which has a single consistent format.
innings.date <- as.character(c(innings$date))
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
# and which has a single consistent format.
# The ODI entries have a different format to the others.

innings.date <- parse_date_time(x = innings.date, c("ymd","mdy","dmy"))

# now the innings.date column is altered

#cbind to dataset innings.
innings <- cbind(innings, innings.date)

#Change back to character
innings.date <- as.character(innings.date)
typeof(innings.date)
```

```
## [1] "character"
```

Question 3b

b.Create another column innings. Rdate which has these dates stored in the R date type. **ANS:**

```
#Rdate which has these dates stored in the R date type.
innings.Rdate <- as.Date(innings.date)

#Create another column innings, cbind to innings
innings <- cbind(innings, innings.Rdate)
#Store as double
typeof(innings.Rdate)
```

```
## [1] "double"
```

```
head(innings)
```

```
##   innings.id team.id team.id.opp ground.id type       date innings.date
## 1      79650     654         231      9264  ODI 24/02/2019   2019-02-24
## 2      37696     230         231      9264  ODI 28/02/2006   2006-02-28
```

```
## 3      21153     230         231      9264 Test 2006-02-18   2006-02-18
## 4      87746     231         654      9264  ODI 03/02/1996   1996-02-03
## 5      44726     175         231      9264 Test 1984-12-21   1984-12-21
## 6      13304     231         654      9264  ODI 10/02/2010   2010-02-10
##   innings.Rdate
## 1    2019-02-24
## 2    2006-02-28
## 3    2006-02-18
## 4    1996-02-03
## 5    1984-12-21
## 6    2010-02-10
```

Question 3c

c.Find the range of dates of the innings in the database **ANS:**

```
range(innings$innings.Rdate, na.rm=TRUE)
```

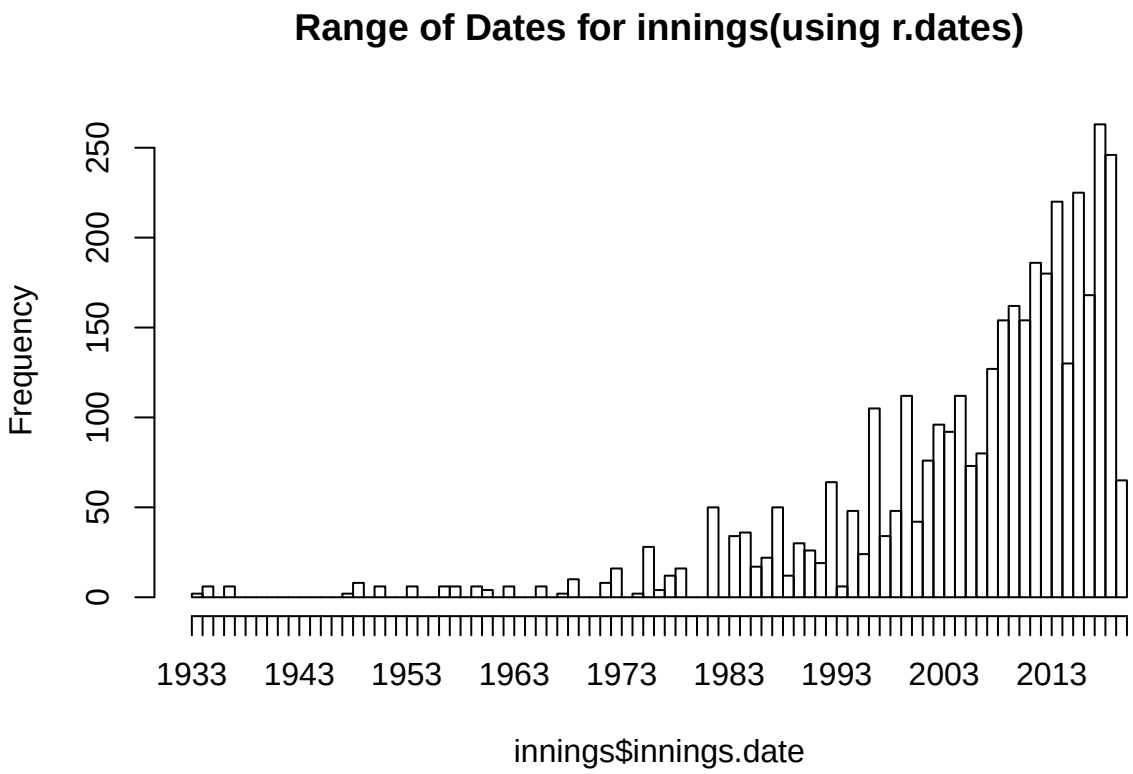
```
## [1] "1934-12-28" "2020-03-08"
```

Question 3d

d.Draw a histogram of the dates of the innings recorded in the database.

ANS:

```
hist(innings$innings.date, breaks="years", freq=TRUE, format="%Y",
     main="Range of Dates for innings(using r.dates)")
```



##Question 4

Q4. (2 Marks) Write R code to copy the tables grounds, teams, players, innings and the combined outcomes table into your SQL database. List the tables in the database to verify that the tables have been created.

ANS:

```
#grounds
dbWriteTable(test_conn, "players", players, overwrite=TRUE)
#Teams
dbWriteTable(test_conn, "teams", teams, overwrite=TRUE)
#players
dbWriteTable(test_conn, "grounds", grounds, overwrite=TRUE)
#innings
dbWriteTable(test_conn, "innings", innings, overwrite=TRUE)
#combines outcomes
dbWriteTable(test_conn, "wcricket", wcricket, overwrite=TRUE)
dbListTables(test_conn)
```

```
## [1] "grounds"  "innings"  "players"  "teams"    "types"    "types2"   "wcricket"
```

Question 5

Q5. (6 Marks) Using a CREATE TABLE command followed by an INSERT command, create a table called types in the database which can be used to decode the type column from its values ODI, Test, T20 into fuller descriptions One Day International, Test Match and Twenty20. Make the type column the primary

key of the table. Use a SELECT command to show the contents of the table. The output of the SELECT statement should be as follows:

ANS: Gets rid of compilation error

```
DROP TABLE IF EXISTS types
```

```
CREATE TABLE types (
  type TEXT,
  type_description TEXT,
  PRIMARY KEY(type)
)
```

```
INSERT INTO types (type, type_description)
VALUES
("ODI", "One Day International"),
("Test", "Test Match"),
("T20", "Twenty20")
```

```
SELECT * FROM types
```

Table 1: 3 records

type	type_description
ODI	One Day International
Test	Test Match
T20	Twenty20

Question 6

Q6. (2 Marks)

a.What is the function of having a primary key in a table in a database?

ANS: The function of having a primary key is a field in a table which uniquely identifies each row/record in a database table

Question 6b

b.What does it mean for a column to be a foreign key in a table in a database?

ANS: A foreign key is pretty much a field in one table that refers to the primary key in another table.

Question 7

Q7. (4 Marks) Write SQL code to count the number of innings by type of match - show your output. Write the query two ways: one way just using the innings table,

ANS:

```
SELECT innings.type, COUNT(*)
FROM  innings
group by innings.type
```

Table 2: 3 records

type	COUNT(*)
ODI	2306
T20	1171
Test	279

Gets rid of compilation error

```
DROP TABLE IF EXISTS types2
```

A second way using the join to types to show the full description of the type column.

```
CREATE TABLE types2 (
  type TEXT,
  Counts INTEGER,
  PRIMARY KEY(type)
)
```

```
INSERT INTO types2 (type, Counts)
VALUES
("ODI", 2306    ),
("T20", 1171    ),
("Test", 279    )
```

```
select *
from types2
```

Table 3: 3 records

type	Counts
ODI	2306
T20	1171
Test	279

```
SELECT *
FROM  types LEFT JOIN types2
ON types.type=types2.type
Order BY type
```

Table 4: 3 records

type	type_description	type	Counts
ODI	One Day International	ODI	2306
T20	Twenty20	T20	1171
Test	Test Match	Test	279

Question 8

Q8. (1 Mark) Write SQL code to show the maximum number of runs scored by any player in an innings.

ANS:

```
select "player.id",MAX(innings_runs_scored_num ) as Maxinnings
from wcricket
```

Table 5: 1 records

player.id	Maxinnings
201658	242

Question 9

Q9. (2 Marks) Write SQL code to find players and innings where 200 or more runs were scored. Show the player id, innings id, and number of runs in your output. Order by DESCENDING number of runs scored.

ANS:

```
select "player.id", "innings.id", innings_runs_scored_num
from wcricket
where innings_runs_scored_num >= 200
order by innings_runs_scored_num DESC
```

Table 6: 9 records

player.id	innings.id	innings_runs_scored_num
201658	27408	242
341870	76485	232
780038	56378	229
142683	37634	214
711527	63131	213
103743	60793	209
440400	37532	204
727715	89141	204
587556	86235	200

Question 10

Q10. (3 Marks) Modify the output of the previous question to include, in addition to the number of runs scored, the name of the player, the name of the team, the date and the name of the ground where the match took place.

ANS:

```
select innings."innings.date", wcricket."innings_runs_scored_num", grounds."ground.name", teams."team.na
from players,teams, innings, grounds, wcricket
where wcricket."innings_runs_scored_num" >= 200
and innings."ground.id" =  grounds."ground.id"
and innings."innings.id" = wcricket."innings.id"
and wcricket."player.id" = players."player.id"
and wcricket."team.id" = teams."team.id"
order by innings_runs_scored_num DESC
```

Table 7: 9 records

innings.date	innings_runs_scored_num	ground.name	team.name	player.name
1079308800	242	Karachi	Pakistan	Kiran Baluch
1528848000	232	Dublin	New Zealand	AC Kerr
882230400	229	Mumbai	Australia	BJ Clark
1029283200	214	Taunton	India	M Raj
1510185600	213	Sydney	Australia	EA Perry
994377600	209	Leeds	Australia	KL Rolton
993340800	204	Shenley	Australia	MAJ Goszko
835574400	204	Scarborough	New Zealand	KE Flavell
902361600	200	Guildford	Australia	J Broadbent

Question 11

Q11. (2 Marks) Using SQL, give the names of the players who have scored more than 6000 runs in total.

ANS: An inning is equivalent to 6ish runs on average. So the max innings is 243. 243*6 = 1458 So in total there is no player that has scored more than 6000 runs in total

```
select players."player.name", wcricket."innings_runs_scored_num" as inningrun
from wcricket, players
where (inningrun*6)>6000
and wcricket."player.id" = players."player.id"
```

Table 8: 0 records

player.name	inningrun

Question 12

Q12. (2 Marks) Using SQL, calculate the average number of runs per player per innings for the three types of game. Include the type_description in your output as well as the short type code. (Hint: the function

AVG() computes the mean of a set of numeric values in a column.)

ANS:

```
select "player.id",AVG(innings_runs_scored_num ) as avgMaxinnings
from wcricket
```

Table 9: 1 records

player.id	avgMaxinnings
719834	17.11023

Question 13

Q13. (2 Marks) Write R code that computes the number of seconds that New Zealand was in Level 4 lockdown earlier this year. Show the R code that creates the date-time objects needed, and the calculation of the time difference.

ANS:

```
#27th april 2020 - 25th march 2020
start <- as.Date("2020-03-25")
end <- as.Date("2020-04-27")

difference <- end-start
difference <- difference*24*60*60
cat("Seconds: ", difference)
```

```
## Seconds:  2851200
```

Disconnect connection

```
dbDisconnect(test_conn)
```