STEVENS INSTITUTE OF TECHNOLOGY
FE 513 - Database Design
Instructor: Cheng Lu

Homework 3

Student: Aidana Bekboeva

Contents

1 Part I. Querying Multiple Tables

1.1 Task 1

Import data from banks_sec_2002 and banks_al_2002. Delete duplicate rows from banks_sec_2002

Here we are creating the table and filling it with data from the csv file.

```
CREATE TABLE IF NOT EXISTS public . banks_al
(
id integer ,
date date ,
asset integer ,
liability integer
)

COPY banks_al (id , date , asset , liability)
FROM 'C:\Users\Public\banks_al_2002.csv'
DELIMITER ','
CSV HEADER

SELECT * FROM banks_al
```

	id integer	date date	asset integer	liability integer
1	23373	2002-09-30	95914	87304
2	23376	2002-12-31	95937	87453
3	23376	2002-03-31	83335	75939
4	23376	2002-06-30	84988	77125
5	23376	2002-09-30	90501	82248
6	234	2002-12-31	56866	49406
7	234	2002-03-31	55204	47914
8	234	2002-06-30	55180	47695
9	234	2002-09-30	56940	49249
10	23404	2002-12-31	78625	72580
11	23404	2002-03-31	72425	66709
12	23404	2002-06-30	73619	67798
13	23404	2002-09-30	73962	68002
14	23406	2002-12-31	2210000	1890000
15	23406	2002-03-31	1940000	1630000

Here we are doing the same for the second csv file. Commented is the part of the code that shows us which rows are duplicates. We then delete these rows.

```
CREATE TABLE IF NOT EXISTS public . banks_sec
(
id integer ,
date date ,
security integer
)

COPY banks_sec (id , date , security)
FROM 'C:\Users\Public\banks_sec_2002-1.csv'
DELIMITER ','
```

```
CSV HEADER

/*SELECT * FROM banks_sec
WHERE id IN (
    SELECT id FROM banks_sec
    EXCEPT SELECT MIN(id) FROM banks_sec
    GROUP BY date, security
    )
ORDER BY id, date, security;*/

DELETE FROM banks_sec
WHERE id IN (
    SELECT id FROM banks_sec
    EXCEPT SELECT MIN(id) FROM banks_sec
    GROUP BY id, date, security
    );

SELECT * FROM banks_sec ORDER BY id
```

	id integer	date date	security integer
1	9	2002-12-31	34957
2	9	2002-09-30	37081
3	9	2002-06-30	39440
4	9	2002-03-31	35605
5	14	2002-06-30	3680000
6	14	2002-12-31	3660000
7	14	2002-09-30	3790000
8	14	2002-03-31	3700000
9	28	2002-06-30	0
10	28	2002-09-30	0
11	28	2002-03-31	0
12	35	2002-09-30	139065
13	35	2002-06-30	125282
14	35	2002-12-31	138784
15	35	2002-03-31	106002

1.2 Task 2

Select the proper join manner to join banks sec 2002 and banks_al_2002. Make sure that all data from banks sec 2002 are kept in the joint table. Report the first 10 observations.

1.3 Task 3

Create a new table banks total. Insert the values from previous joint table into this new one. And set a primary key for the table.

I combined tasks 2 and 3 into one with the following query:

```
CREATE TABLE banks_total AS (
```

```
SELECT banks_al.id , banks_al.asset , banks_sec.security
FROM banks_al
INNER JOIN banks_sec  on banks_al.id = banks_sec.id
AND banks_al.date = banks_sec.date
LIMIT 10);

SELECT * FROM banks_total;
ALTER TABLE banks_total ADD PRIMARY KEY (id);
```

	id [PK] integer	asset integer	security integer
1	32307	53714	0
2	22598	57360	0
3	15879	76960	5357
4	35373	46551	0
5	5226	53873	7960
6	22092	147828	0
7	13749	214733	17476
8	823	271961	44972
9	29831	58300	6505
10	10203	320853	3889

1.4 Task 4

For each quarter of the year, 2002 count how many banks have security over 20% of their' asset?

For this task, we need to make an inner join between two tables and select such rows that would display those banks that have security over 20% of their asset. Unfortunately, I just remembered that I only counted banks for one quarter, but I would calculate the other three similarly, changing the dates in lines 5' and '13'. For the first quarter, the answer is 984 banks.

```
SELECT banks_al.id , banks_al.asset , banks_sec.security
FROM banks_al
INNER JOIN banks_sec ON banks_al.id = banks_sec.id AND
                  banks_al.date = banks_sec.date
WHERE (banks_al.date BETWEEN '2002-01-01' AND '2002-03-31')
AND (banks_sec.security > banks_al.asset * 0.2)
ORDER BY banks_al.id;

SELECT COUNT(banks_al.id)
FROM banks_al
INNER JOIN banks_sec ON banks_al.id = banks_sec.id AND
                  banks_al.date = banks_sec.date
WHERE (banks_al.date BETWEEN '2002-01-01' AND '2002-03-31')
AND (banks_sec.security > banks_al.asset * 0.2)
```

	id integer	asset integer	security integer
1	35	471056	106002
2	131	1250000	349389
3	183	96051	19685
4	402	90791	29730
5	404	48957	14809
6	406	31823	8150
7	441	919796	190868
8	445	226980	58660
9	609	888969	182312
10	622	3020000	1750000
11	659	182116	40764
12	679	2400000	581941
13	753	55594	11574
14	851	84932	18539
15	862	78435	28314

	count bigint
1	984

1.5 Task 5

How many banks have liability over 90% of assets in the first quarter of 2002 but go below 90% in the second quarter of 2002?

```
drop table tempor
CREATE TABLE tempor AS (
SELECT banks_al.id , banks_al.date, banks_al.asset , banks_al.liability
FROM banks_al
WHERE ((banks_al.date BETWEEN '2002-01-01' AND '2002-03-31')
AND (banks_al.liability > banks_al.asset * 0.9))
ORDER BY banks_al.id)

SELECT * FROM tempor

SELECT COUNT(tempor.id)
FROM tempor
WHERE ((tempor.date BETWEEN '2002-04-01' AND '2002-06-30')
AND (tempor.liability < tempor.asset * 0.9))
```

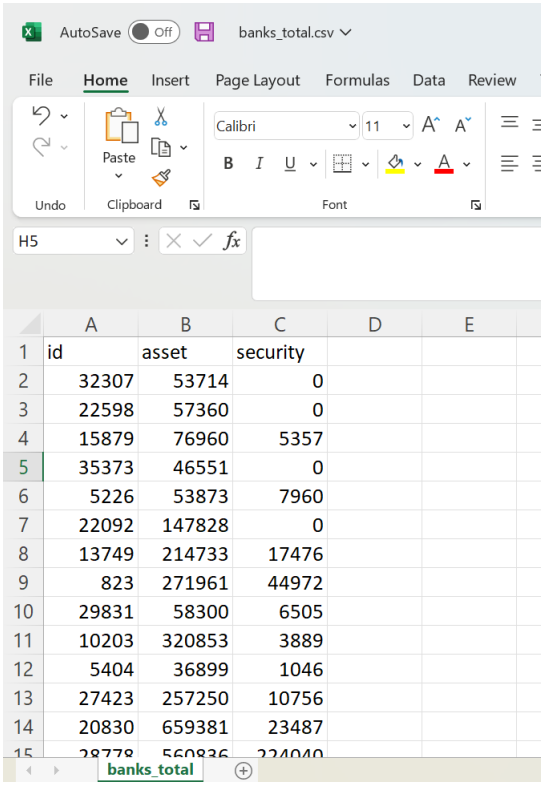
Answer: 0

2 Part II. PostgreSQL API in R

Making a connection to the local PostgreSQL database using API. Exporting the joint table (banks total) to a csv file.

```
library(DBI)

con <- dbConnect(RPostgres::Postgres(),dbname = 'postgres',
                 host = "localhost",
                 port = 5432,
                 user = 'postgres',
```

```
                  password ='mypassword')

bankstotal <- dbSendQuery(con,
                          "SELECT id, asset, security
FROM banks_total;")

bankstotal <- as.data.frame(dbFetch(bankstotal))
write.csv(bankstotal, "C:\\Users\\Public\\banks_total.csv", row.names=FALSE)

length(bankstotal$id)
```



	A	B	C	D	E
1	id	asset	security		
2	32307	53714	0		
3	22598	57360	0		
4	15879	76960	5357		
5	35373	46551	0		
6	5226	53873	7960		
7	22092	147828	0		
8	13749	214733	17476		
9	823	271961	44972		
10	29831	58300	6505		
11	10203	320853	3889		
12	5404	36899	1046		
13	27423	257250	10756		
14	20830	659381	23487		