



DATABASE SYSTEMS

SQL practice



BUDAPESTI MŰSZAKI
ÉS GAZDASÁGTUDOMÁNYI EGYETEM
Építőmérnöki Kar - építőmérnöki képzés 1782 óta

Fotogrammetria és Térinformatika Tanszék

Bence Molnár

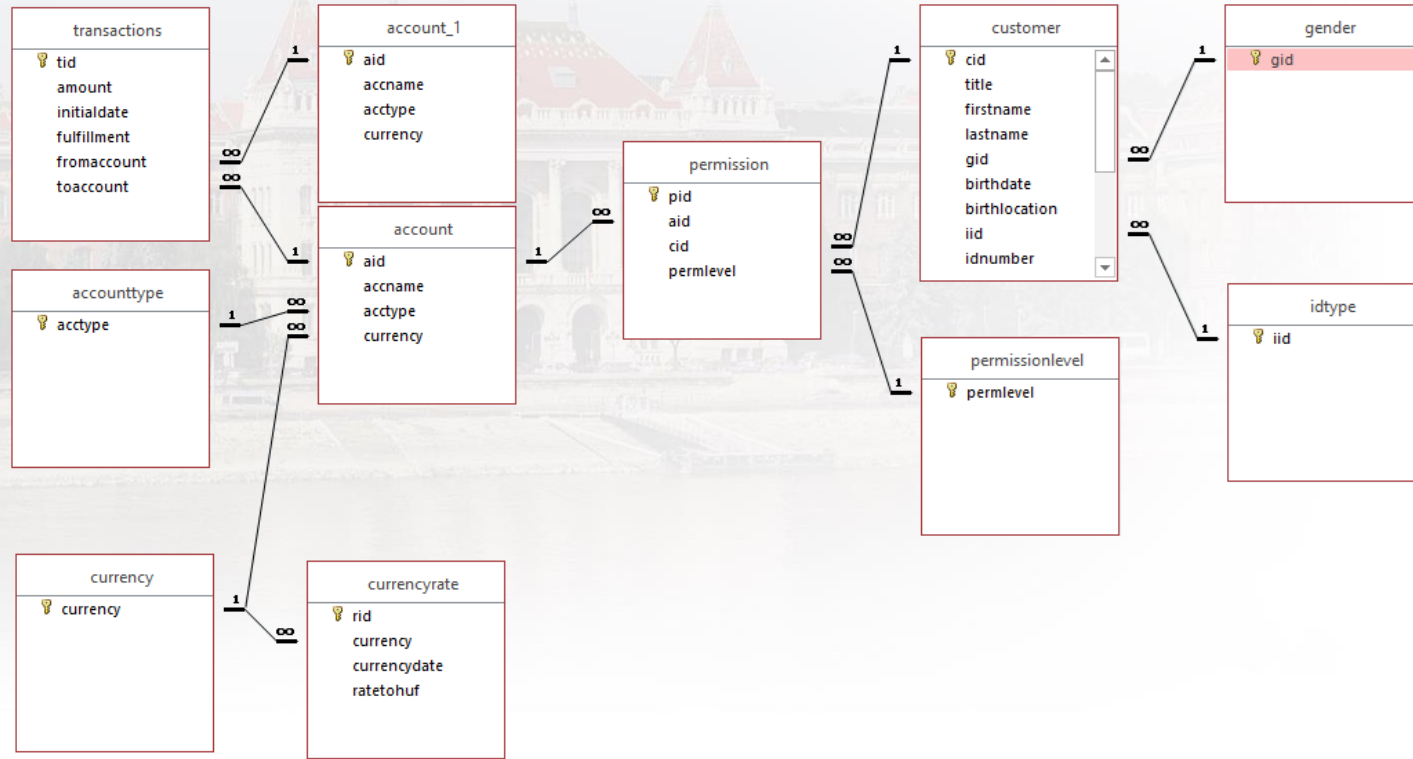
2022.04.20.

AGENDA

Formulating queries in SQL

- Projection
- Join
- Filtering
- Grouping
- Sorting
- Limit rows

RELATIONAL SCHEMA DIAGRAM



LIST ALL CUSTOMERS

Columns: title, firstname, lastname

Tables: customer

Joins: -

```
SELECT  
    title&' '&firstname&' '&lastname  
FROM  
    customer;
```

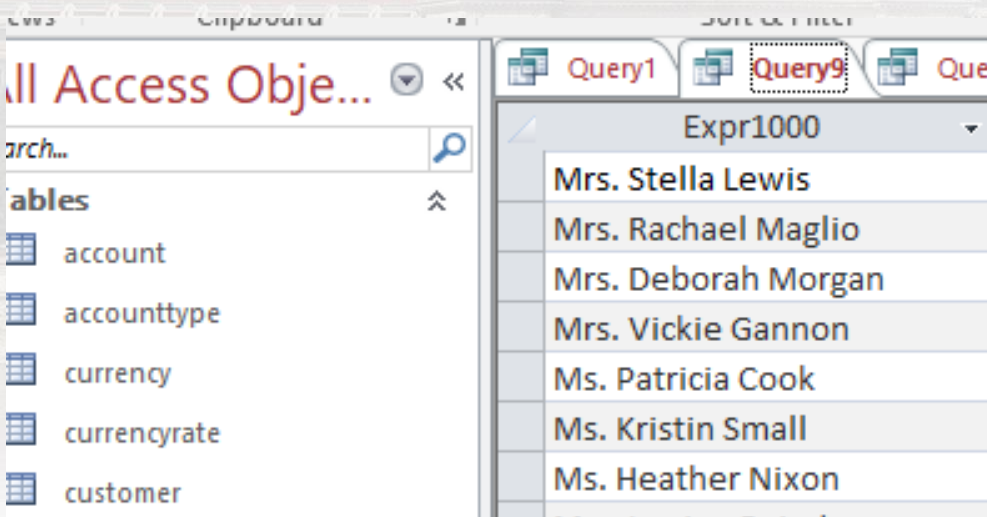
The screenshot shows the Microsoft Access interface. On the left, the 'All Access Objects' pane is visible with a search bar and a list of tables: account, accounttype, currency, currencyrate, and customer. The 'customer' table is selected. In the center, the 'Query1' tab is active, showing a table with a single column 'Expr1000'. The table contains six rows of concatenated text: 'Mr. Brent Moreno', 'Mrs. Stella Lewis', 'Mrs. Rachael Maglio', 'Mr. Jeffrey Bowen', 'Mr. Danny Jones', and 'Mrs. Deborah Morgan'. The last row is 'Mr. Anthony Davis'.

Expr1000
Mr. Brent Moreno
Mrs. Stella Lewis
Mrs. Rachael Maglio
Mr. Jeffrey Bowen
Mr. Danny Jones
Mrs. Deborah Morgan
Mr. Anthony Davis

Concatenate into a single cell!

LIST ALL CUSTOMERS, BUT EXCLUDE EVERYBODY WITH MR. TITLE.

Columns: title, firstname, lastname



title	lastname
Mrs.	Stella Lewis
Mrs.	Rachael Maglio
Mrs.	Deborah Morgan
Mrs.	Vickie Gannon
Ms.	Patricia Cook
Ms.	Kristin Small
Ms.	Heather Nixon

```
SELECT
    title & ' ' & firstname & ' ' & lastname
FROM
    customer
WHERE
    title <> 'Mr.';
```

Concatenate into a single cell!

Sorting: -

Rows num: -

LIST ALL CUSTOMERS AND SORT BY NAME

Columns: title, firstname, lastname

Tables: customer

Joins: -

Filtering: title<>'Mr.'

Grouping: -

Group filtering: -

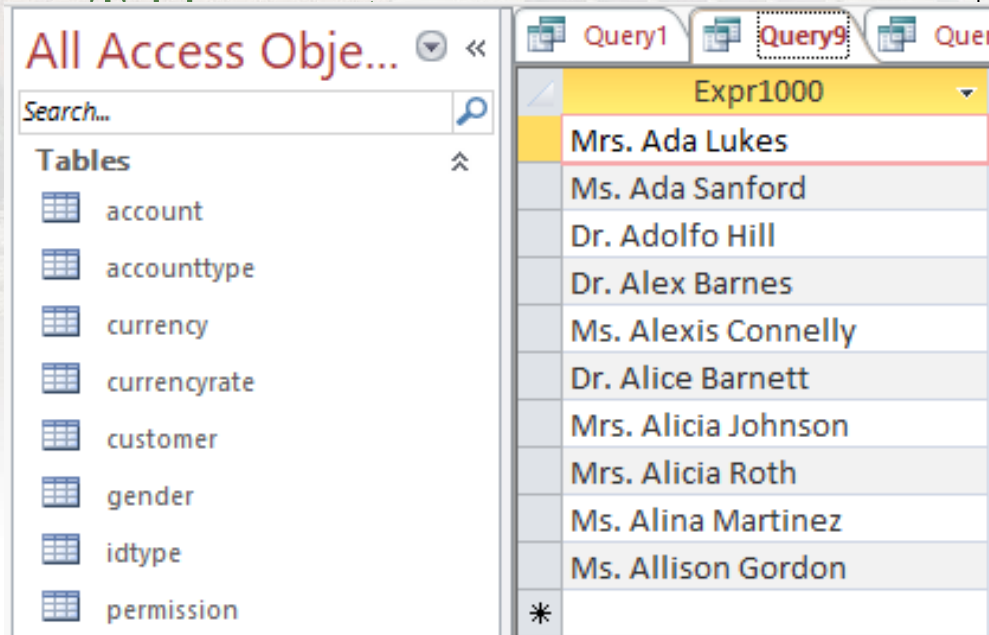
Sorting: firstname, lastname

Num rows: -

```
SELECT
    title&' '&firstname&' '&lastname
FROM
    customer
WHERE
    title<>'Mr.'
ORDER BY
    firstname, lastname;
```

LIST ALL CUSTOMERS, ORDER BY NAME, BUT DISPLAY ONLY THE FIRST 10 ITEMS

Columns: title, firstname, lastname



All Access Objects

Search...

Tables

- account
- accounttype
- currency
- currencyrate
- customer
- gender
- idtype
- permission

Query1 Query9 Query10

Expr1000

Mrs. Ada Lukes
Ms. Ada Sanford
Dr. Adolfo Hill
Dr. Alex Barnes
Ms. Alexis Connolly
Dr. Alice Barnett
Mrs. Alicia Johnson
Mrs. Alicia Roth
Ms. Alina Martinez
Ms. Allison Gordon
*

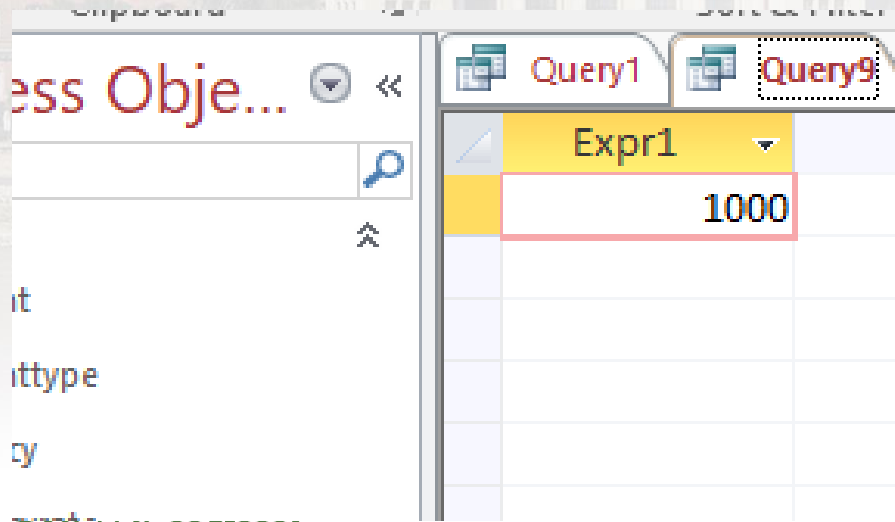
```
SELECT TOP 10
    title & ' ' & firstname & ' ' & lastname
FROM
    customer
WHERE
    title <> 'Mr.'
ORDER BY
    firstname, lastname;
```

NUMBER OF CUSTOMERS

Columns: COUNT(*)

Tables: customer

```
SELECT  
    COUNT(*)  
FROM  
    customer;
```



The screenshot shows a database query interface. On the left, there is a sidebar with a search icon and some text. The main area displays a table with two tabs: 'Query1' and 'Query9'. The table has one column labeled 'Expr1' and one row with the value '1000'.

Expr1
1000

NUMBER OF NON-MALE CUSTOMERS

Columns: COUNT(*)

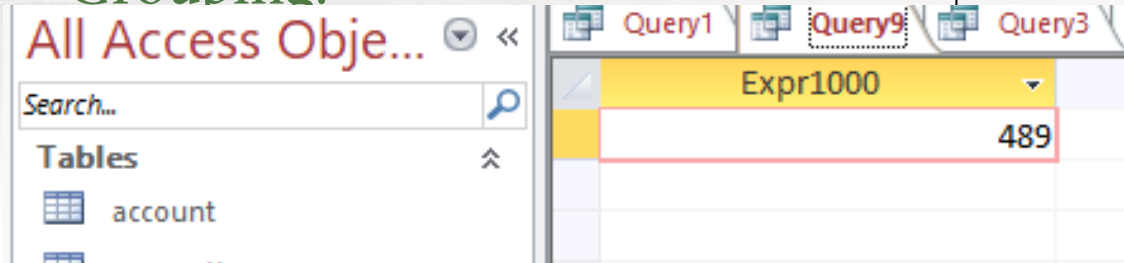
Tables: customer

Joins: -

Filtering: gid<>'male'

Grouping: -

```
SELECT
    COUNT(*)
FROM
    customer
WHERE
    gid<>'male';
```



The screenshot shows the Microsoft Access interface. On the left, the 'All Access Objects' pane is visible with a search bar and a list of tables including 'account'. The main window displays a query result for 'Query9'. The table has a header row with 'Expr1000' and a data row with the value '489'.

Expr1000
489

TOTAL CACHE FLOW

Columns: SUM(amount)

Tables: transaction

Joins: -

Filtering: -

Grouping: -

Group filtering: -

Sorting: -

Rows num: -

```
SELECT
    SUM(amount)
FROM
    transaction;
```

MAXIMUM OF HUF-EUR EXCHANGE RATE

Columns: MAX(ratetohuf)

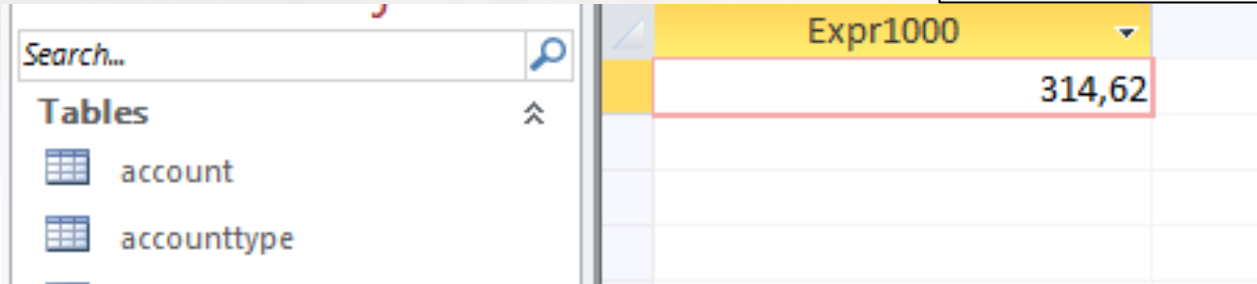
Tables: currencyrate

Joins: -

Filtering: currency='EUR'

Grouping: -

```
SELECT
    MAX(ratetohuf)
FROM
    currencyrate
WHERE
    currency='EUR';
```



The screenshot shows a database query interface. On the left, there is a search bar labeled "Search..." and a list of tables: "account" and "accounttype". On the right, a query result table is displayed. The table has a header row with the column name "Expr1000" and a data row with the value "314,62".

Expr1000
314,62

MAXIMUM OF EACH CURRENCIES

Columns: currency, MAX(ratetohuf)

Tables: currencyrate

```
SELECT
    currency, MAX(ratetohuf)
FROM
    currencyrate
GROUP BY
    currency;
```

currency	Expr1001
CHF	293,16
EUR	314,62
GBP	374,59
HUF	251
USD	297,38

Rows num: -

We will put the date next to it soon!

WHICH ACCOUNT IS ACCESSIBLE BY MOST PERSON?

Please see structure of the permissions table

Columns: aid, COUNT(*)

Tables: permission

Joins: -

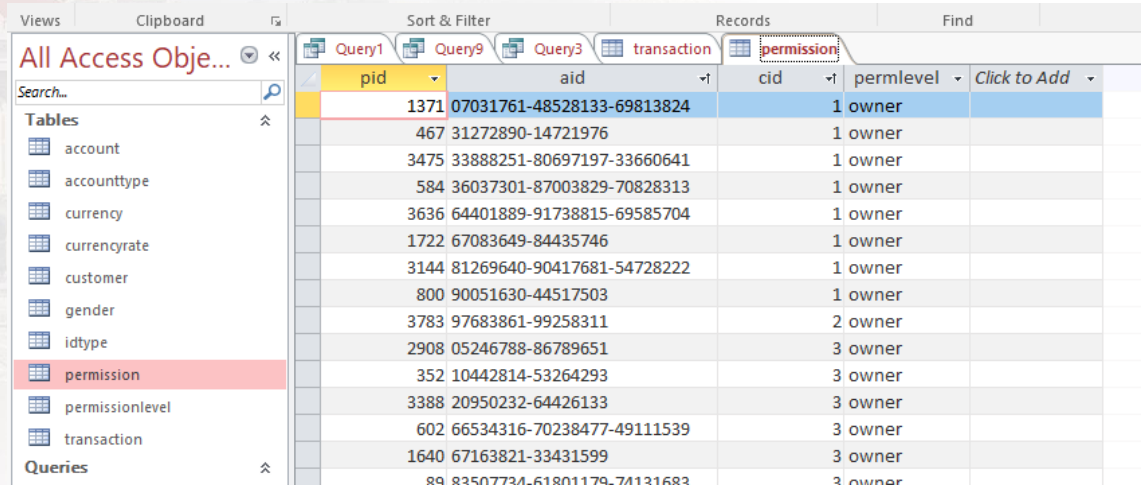
Filtering: -

Grouping: aid

Group filtering: -

Sorting: COUNT(*)

Rows num: -



The screenshot shows a database management interface with a table named 'permission'. The table has columns: pid, aid, cid, permlevel, and Click to Add. The data is sorted by aid in descending order. The first row is highlighted in blue.

pid	aid	cid	permlevel	Click to Add
1371	07031761-48528133-69813824	1	owner	
467	31272890-14721976	1	owner	
3475	33888251-80697197-33660641	1	owner	
584	36037301-87003829-70828313	1	owner	
3636	64401889-91738815-69585704	1	owner	
1722	67083649-84435746	1	owner	
3144	81269640-90417681-54728222	1	owner	
800	90051630-44517503	1	owner	
3783	97683861-99258311	2	owner	
2908	05246788-86789651	3	owner	
352	10442814-53264293	3	owner	
3388	20950232-64426133	3	owner	
602	66534316-70238477-49111539	3	owner	
1640	67163821-33431599	3	owner	
89	83507724-61801179-74121683	3	owner	

WHICH ACCOUNT IS ACCESSIBLE BY MOST PERSON?

Sort & Filter		Rec
Query1	Query9	Query3
transaction		
aid	Expr1001	
78171884-91227545	10	
06684326-26300089	10	
99758242-76257656-19141475	9	
19726924-29805121	9	
29427998-90346996	9	
82351621-04186569	9	
93807336-38250753-10254098	9	
06781933-91757694	9	
74640375-80815842	8	
37291243-18615380	8	
71285733-64801647-47652007	8	
50275028-98726239	8	

```
SELECT
    aid, COUNT(*)
FROM
    permission
GROUP BY
    aid
ORDER BY
    COUNT(*) DESC;
```

WHICH ACCOUNT IS ACCESSIBLE BY MOST PERSON?

Columns: aid, COUNT(*)

Tables: permission

Joins: -

Filtering:

permlevel='owner' OR

Permlevel='partner'

Grouping: aid

Group filtering: -

Sorting: COUNT(*)

Rows num: -

```
SELECT
    aid, COUNT(*)
FROM
    permission
WHERE
    permlevel='owner' OR
    permlevel='partner'
GROUP BY
    aid
ORDER BY
    COUNT(*) DESC;
```

CUSTOMER WITH THE MOST BANK ACCOUNTS

Columns: title, firstname, lastname, COUNT(*)

Table: customer

title	firstname	lastname	Expr1003
Mr.	Timothy	Lopez	11
Ms.	Larissa	Cardwell	11
Mr.	Michael	Dawson	11
Mr.	Clark	Hook	10
Mr.	Jeremy	Garcia	10
Mr.	Jeffrey	Perry	10
Mrs.	Mary	Keck	10
Mr.	Thomas	Fox	10
Mr.	Bryan	Schumacher	10
Mrs.	Ida	Ash	9
Mr.	Daniel	Green	9
Dr.	Adolfo	Hill	9
Mr	Eric	Riv	9

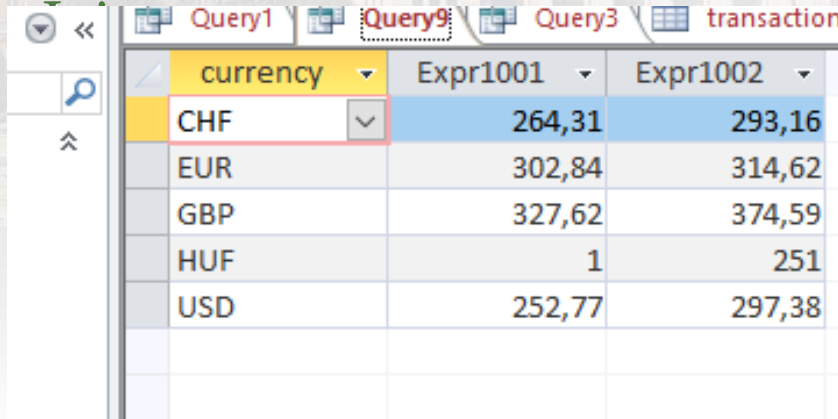
```
SELECT
    title, firstname, lastname, COUNT(*)
FROM
    customer
INNER JOIN
    permission
ON
    customer.cid=permission.cid
GROUP BY
    title, firstname, lastname
ORDER BY
    COUNT(*) DESC;
```

MIN & MAX CURRENCY RATES

Although the maximum value is present, the line has not been identified!

Columns: currency, MAX(ratetohuf), MIN(ratetohuf)

Tables: currencyrate



currency	Expr1001	Expr1002
CHF	264,31	293,16
EUR	302,84	314,62
GBP	327,62	374,59
HUF	1	251
USD	252,77	297,38

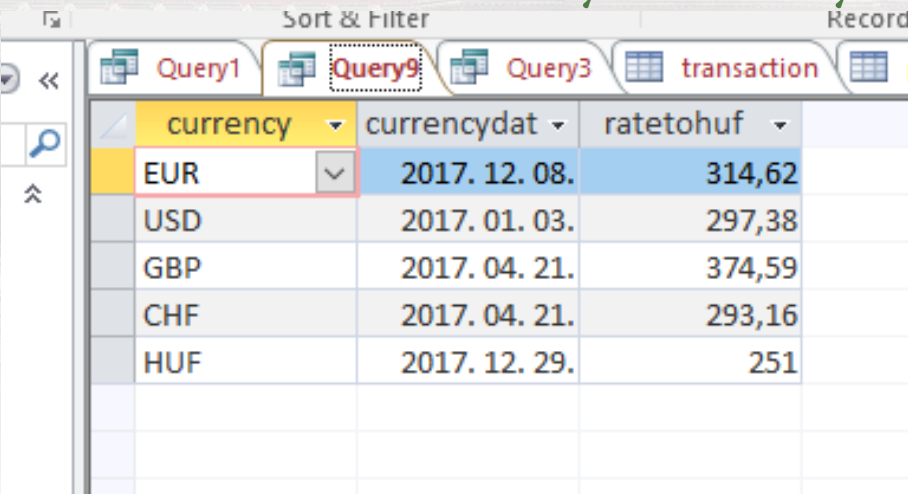
```
SELECT
    currency,
    MIN(ratetohuf),
    MAX(ratetohuf)
FROM
    currencyrate
GROUP BY
    currency;
```

Rows num: -

MAX CURRENCY RATES INCLUDING DATES

Although the maximum value is present, the line has not been identified!

Columns: currency, currencydate, ratetohuf



currency	currencydate	ratetohuf
EUR	2017. 12. 08.	314,62
USD	2017. 01. 03.	297,38
GBP	2017. 04. 21.	374,59
CHF	2017. 04. 21.	293,16
HUF	2017. 12. 29.	251

Sorting: -

Rows num: -

SELECT

currencyrate.currency,
currencyrate.currencydate,
currencyrate.ratetohuf

FROM

currencyrate

INNER JOIN

(SELECT currency, MAX(ratetohuf) AS m

FROM currencyrate GROUP BY currency) AS maxs

ON

currencyrate.currency=maxs.currency

AND currencyrate.ratetohuf=maxs.m;

MAX CURRENCY RATES INCLUDING DATES, IN CASE OF MULTIPLE PEAKS, ONLY SHOW LAST ONE

Columns: currency, LAST(currencydate), ratetohuf

Tables: currencyrate

Joins: (Previous query)

Filtering: -

Grouping: currency, ratetohuf

Group filtering: -

Sorting: -

Rows num: -

```
SELECT
currencyrate.currency,
LAST(currencyrate.currencydate),
currencyrate.ratetohuf
FROM
currencyrate
INNER JOIN
(SELECT currency, MAX(ratetohuf) AS m FROM
currencyrate GROUP BY currency) AS maxs
ON
currencyrate.currency=maxs.currency AND
currencyrate.ratetohuf=maxs.m
GROUP BY
currencyrate.currency, currencyrate.ratetohuf;
```

COMPLEX SQL QUERY IN THE HOMEWORK SYSTEM

```
SELECT users.fullname, users.username, users.mail, gname, users.crid, SUBSTR(u6.fullname, 5, 1), DATE_FORMAT(users.last, '%m/%d %k:%i:%s'),
users.p1, DATE_FORMAT(p1.date, '%m/%d %k:%i:%s'), DATE_FORMAT(p1_d.date, '%m/%d %k:%i:%s'), users.p1_fee, users.p1_spec, users.p1_cor<p1.date, mn1,
users.p2, DATE_FORMAT(p2.date, '%m/%d %k:%i:%s'), DATE_FORMAT(p2_d.date, '%m/%d %k:%i:%s'), users.p2_fee, users.p2_spec, users.p2_cor<p2.date, mn2,
users.p3, DATE_FORMAT(p3.date, '%m/%d %k:%i:%s'), DATE_FORMAT(p3_d.date, '%m/%d %k:%i:%s'), users.p3_fee, users.p3_spec, users.p3_cor<p3.date, mn3,
users.t1, users.t1_group, users.t2, users.t3, grade, u1.p3 u1, u2.p3 u2, u4.p3 u4, u5.p3 u5
FROM users
JOIN groups USING(gid)
JOIN users_grade USING (uid)
LEFT JOIN (SELECT uid, MAX(date) AS date FROM part1 GROUP BY uid) AS p1 USING(uid)
LEFT JOIN (SELECT uid, MAX(date) AS date FROM part1 WHERE date <= '2021-10-01 11:59:59' GROUP BY uid) AS p1_d USING(uid)
LEFT JOIN chain AS c1 ON c1.uid1=uid
LEFT JOIN (SELECT cid AS cid1, COUNT(1) AS mn1 FROM msg AS m1 GROUP BY cid1) AS msg1 ON c1.cid=cid1
LEFT JOIN (SELECT uid, MAX(date) AS date FROM part2 GROUP BY uid) AS p2 USING(uid)
LEFT JOIN (SELECT uid, MAX(date) AS date FROM part2 WHERE date <= '2021-11-05 11:59:59' GROUP BY uid) AS p2_d USING(uid)
LEFT JOIN chain AS c2 ON c2.uid2=uid
LEFT JOIN (SELECT cid AS cid2, COUNT(1) AS mn2 FROM msg AS m2 GROUP BY cid2) AS msg2 ON c2.cid=cid2
LEFT JOIN (SELECT uid, MAX(date) AS date FROM part3 GROUP BY uid) AS p3 USING(uid)
LEFT JOIN (SELECT uid, MAX(date) AS date FROM part3 WHERE date <= '2021-12-10 11:59:59' GROUP BY uid) AS p3_d USING(uid)
LEFT JOIN chain AS c3 ON c3.uid3=uid
LEFT JOIN (SELECT cid AS cid3, COUNT(1) AS mn3 FROM msg AS m3 GROUP BY cid3) AS msg3 ON c3.cid=cid3
LEFT JOIN users u1 ON c3.uid1=u1.uid
LEFT JOIN users u2 ON c3.uid2=u2.uid
LEFT JOIN users u4 ON c1.uid3=u4.uid
LEFT JOIN users u5 ON c2.uid3=u5.uid
LEFT JOIN course ON users.crid=course.crid
LEFT JOIN users u6 ON course.uid=u6.uid
WHERE users.permission=2
ORDER BY users.fullname;
```

COMPLEX SQL QUERY IN NEPTUN

```
SELECT DISTINCT
t100.C_CODE AS "F0",
t100.C_DESCRIPTION AS "F1",
t120.C_CODE AS "F2",
t120.C_NAME AS "F3",
t120.C_NAME_1 AS "F4",
t131.C_NEPTUNCODE AS "F5",
t130.C_TITLE AS "F6",
t130.C_LASTNAME AS "F7",
t130.C_FIRSTNAME AS "F8"
FROM
T_SUBJECT t120,
T_ORGANIZATION t100, T_INTERIORORGANIZATION t101,
T_ORGANIZATION t110, T_INTERIORORGANIZATION t111,
T_COURSE t140
RIGHT JOIN T_CourseTutor t18 ON t140.ID=t18.C_COURSEID
RIGHT JOIN T_Employee t132 ON t18.C_EMPLOYEEID=t132.ID
RIGHT JOIN T_PARTNER t130 ON t132.ID = t130.ID
RIGHT JOIN T_USER t131 ON t132.ID = t131.ID
WHERE t100.ID=t101.ID AND
t100.C_BOSSID = t110.ID AND
t110.ID=t111.ID AND
t120.ID = t140.C_SUBJECTID AND
t120.C_INTERIORORGANIZATIONID = t101.ID AND
(t100.C_CODE LIKE 'EO%')
AND
(t140.C_TERM = 70610) AND
(t140.C_LANGUAGE = 137001) AND
(t140.C_CODE <> 'VK')

ORDER BY
3,
9,
8,
6
```

CONCLUSION

Formulating queries in SQL

- Projection
- Join
- Filtering
- Grouping
- Sorting
- Limit rows



Thank you for your attention!

Questions?



**BUDAPESTI MŰSZAKI
ÉS GAZDASÁGTUDOMÁNYI EGYETEM**

Építőmérnöki Kar - építőmérnöki képzés 1782 óta

Fotogrammetria és Térinformatika Tanszék