

NAMA : M. Fahmi Kurniawan
NIM : 225150600111014
KELAS : DBDSQL-A
BAB : FUNGSI SKALAR DAN FUNGSI AGREGASI
ASISTEN : 1. Iqbal Biondy
2. Sri Veranda Lucky Desmala

1. Gambar ERD

2. Langkah-langkah praktikum

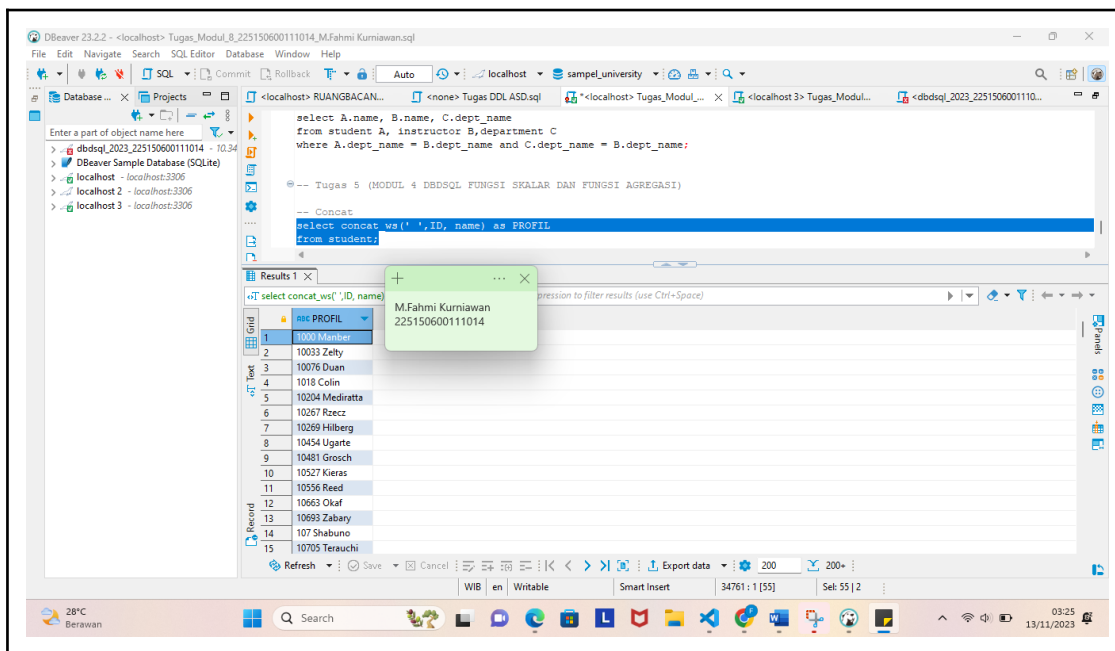
A. Fungsi Skalar

- CONCAT

Jadi saya disini menggunakan sintaks dasar di Modul Fungsi Skalar dan Fungsi agregasi. dimana nanti dikode bawah ini kita membuat concat_ws untuk menggabungkan colom ID dan Name pada Sebuah Colom Profil yang dimana referensi colomnya dari table Student. Dikode dibawah ini juga ada spasi agar ID dan Profil ada spasinya.

```
select concat_ws(' ',ID, name) as PROFIL  
from student;
```

Output



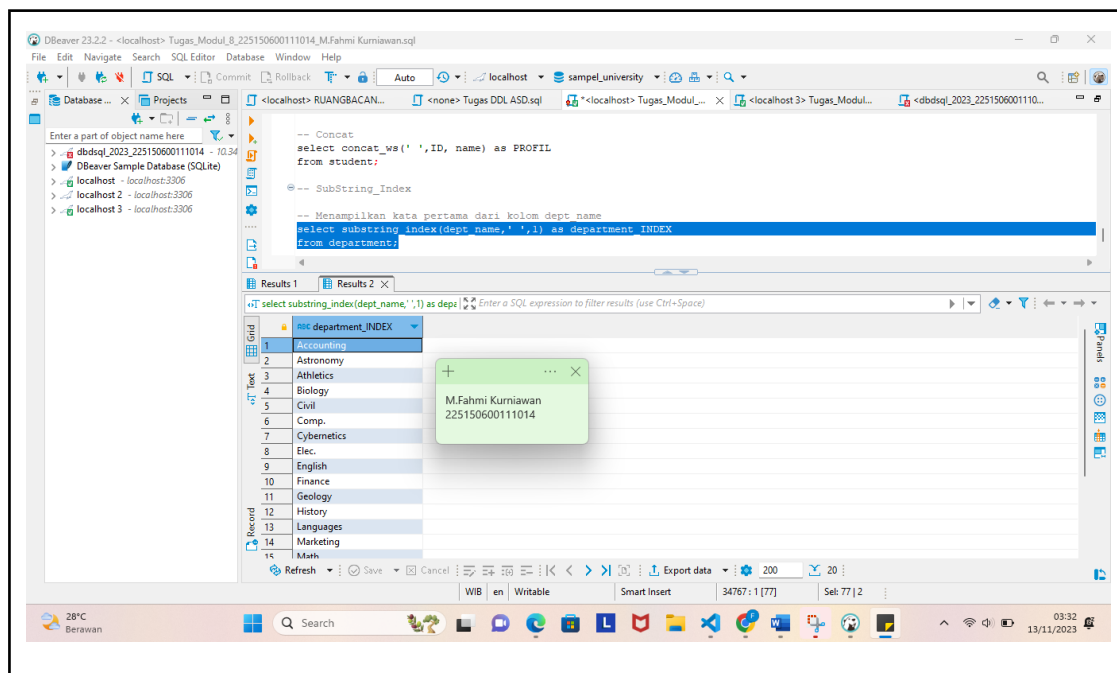
PROFIL	ID
M.Fahmi Kurniawan	225150600111014
1000 Member	1000
10033 Zely	10033
10076 Duan	10076
1018 Colin	1018
10204 Mediratta	10204
10267 Rzcz	10267
10269 Hilberg	10269
10454 Ugarte	10454
10481 Grosch	10481
10527 Kieras	10527
10556 Reed	10556
10663 Okaf	10663
10693 Zahary	10693
107 Shabuno	107
10705 Terauchi	10705

- **SUBSTR_INDEX**

Untuk SUBSTR_INDEX saya masih menggunakan acuan sintaks yang ada dimodul, yang dimana saya hanya merubah/menyesuaikan nama tabel dan kolom yang bisa digunakan pada perintah substring. Disini Perintah substring saya berguna untuk Menampilkan kata pertama dari kolom dept_name. didalam kurung buka dan tutup saya menuliskan nama kolom dept_name, untuk delimiternya saya menggunakan Spasi(" "), dan untuk index to return nya satu yang dimana nantin agar kita bisa menampilkan satu kata pertaman pada kolom dept_name.

```
-- SubString_Index
-- Menampilkan kata pertama dari kolom dept_name
select substring_index(dept_name, ' ', 1) as department_INDEX
from department;
```

Output



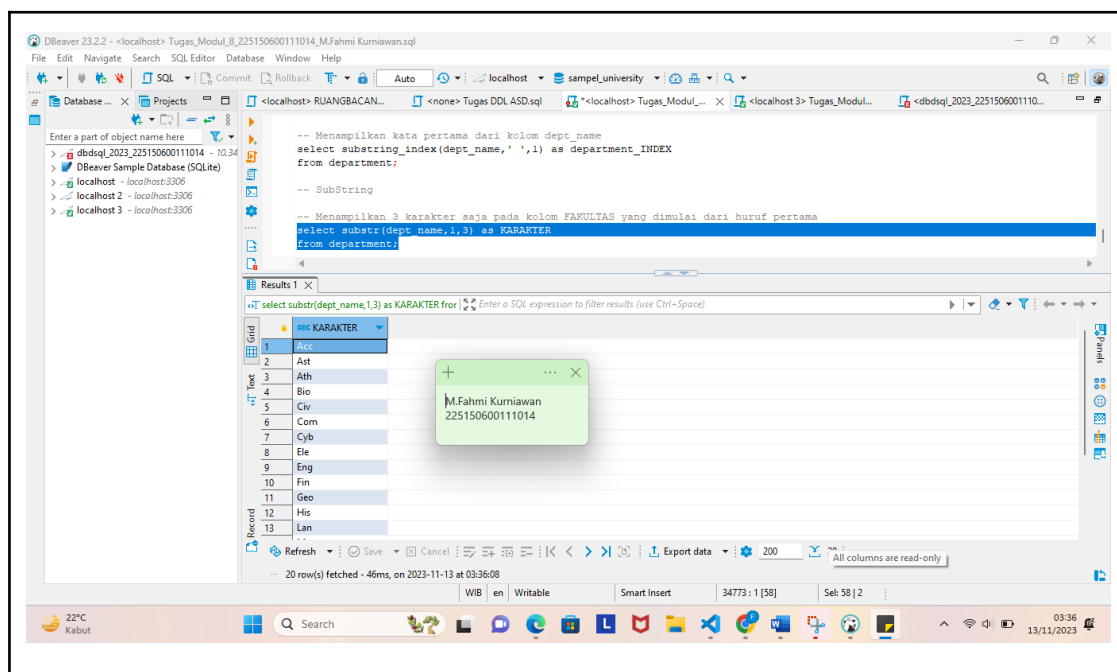
- **SUBSTR**

Untuk SUBSTR saya masih menggunakan acuan sintaks yang ada dimodul, yang dimana saya hanya merubah/menyesuaikan nama tabel dan kolom yang bisa digunakan pada perintah substring. Disini Perintah substring saya berguna untuk Menampilkan 3 karakter saja pada kolom dept_name yang dimulai dari huruf pertama. didalam kurung buka dan tutup saya menuliskan nama kolom dept_name, untuk Start

Index saya tuliskan 1 yang dimana nantinya index dari data per kolom akan ditampilkan dari index ke 1, dan untuk Number of string to be extract saya menggunakan 3 yang nantinya kita hanya mengambil 3 karakter yang dimulai dari index ke 1.

```
-- SubString
-- Menampilkan 3 karakter saja pada kolom dept_name yang dimulai dari
huruf pertama
select substr(dept_name,1,3) as KARAKTER
from department;
```

Output

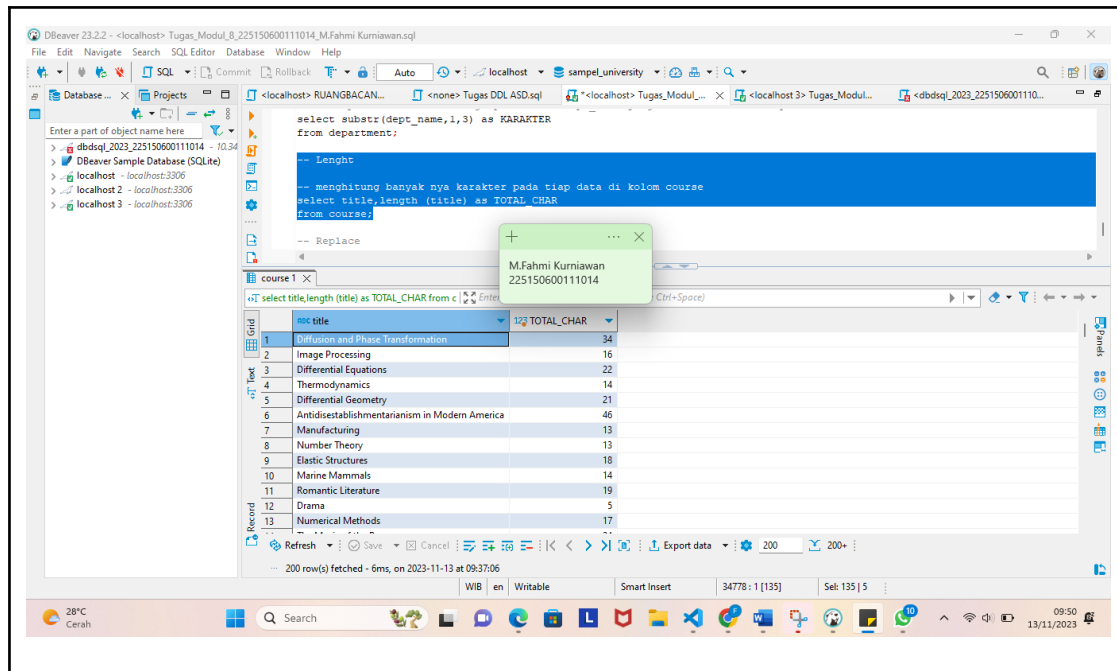


- **LENGTH**

Untuk LENGTH saya masih menggunakan acuan sintaks yang ada dimodul, yang dimana saya hanya merubah/menyesuaikan nama tabel dan kolom yang bisa digunakan pada perintah substring. Disini Perintah LENGTH saya berguna untuk menghitung banyak nya karakter pada tiap data di kolom course. didalam kurung buka dan tutup saya menuliskan nama kolom Title, dan From nya saya tuliskan table course yang nantinya akan merujuk kekolom title yang ada di course.

```
-- Lenght
-- menghitung banyak nya karakter pada tiap data di kolom course
select title,length (title) as TOTAL_CHAR
from course;
```

Output



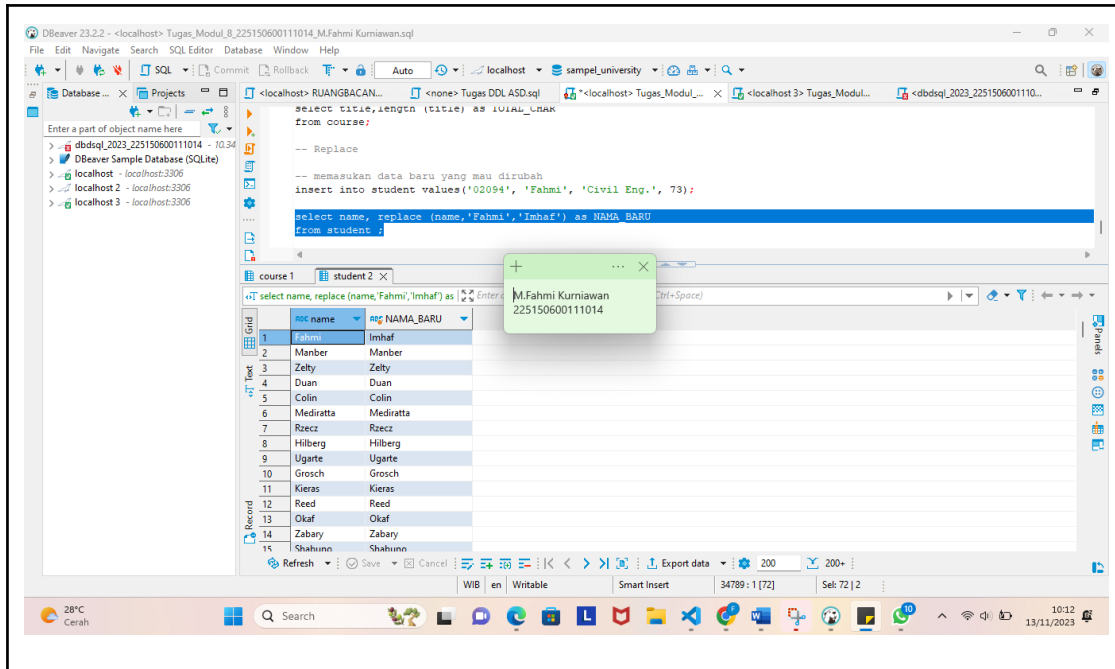
• REPLACE

Untuk REPLACE saya masih menggunakan acuan sintaks yang ada dimodul, yang dimana saya hanya merubah/menyesuaikan nama tabel dan kolom yang bisa digunakan pada perintah substring. Disini Perintah REPLACE saya berguna untuk merubah nama student. pertama saya menambahkan data mahasiswa baru yang bernama Fahmi. didalam kurung buka dan tutup saya menuliskan nama kolom name, lalu menuliskan data name Fahmi, lalu menuliskan data name yang mau diganti yaitu Imhaf dan From nya saya tuliskan table student yang nantinya akan merujuk kekolom name yang ada di student.

```
-- Replace
-- memasukan data baru yang mau dirubah

insert into student values('02094', 'Fahmi', 'Civil Eng.', 73);
select name, replace (name,'Fahmi','Imhaf') as NAMA_BARU
from student ;
```

Output



The screenshot shows the DBeaver 23.2.2 interface with the following components:

- SQL Editor:** Contains the following SQL code:


```
select title, length(title) as LUAL_UKAP
from course;

-- Replace
-- memasukan data baru yang mau dirubah
insert into student values('02094', 'Fahmi', 'Civil Eng.', 73);

select name, replace(name, 'Fahmi', 'Imhaf') as NAMA_BARU
from student;
```
- Database Navigator:** Shows the project structure with 'dbdsq_2023_225150600111014' selected.
- Table Grid:** Displays the results of the SQL query. The table has columns 'name' and 'NAMA_BARU'. The first row shows 'Fahmi' being replaced with 'Imhaf'.
- Table Data:**

name	NAMA_BARU
Fahmi	Imhaf
Manber	Manber
Zelty	Zelty
Duan	Duan
Colin	Colin
Mediratta	Mediratta
Rzezz	Rzezz
Hilberg	Hilberg
Ugarte	Ugarte
Grosch	Grosch
Kieras	Kieras
Reed	Reed
Okaf	Okaf
Zabary	Zabary
Shahunn	Shahunn
- Task List:** Shows a task 'M.Fahmi Kurniawan' with ID '225150600111014'.
- Bottom Bar:** Displays system information including temperature (28°C), search bar, and system clock (10:12, 13/11/2023).

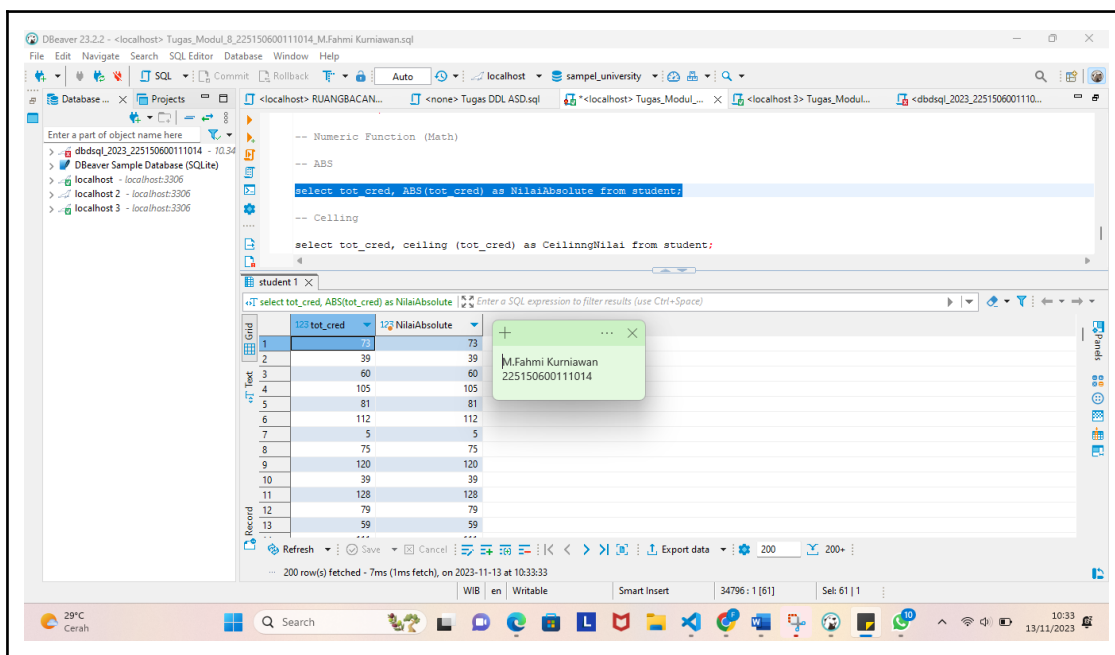
B. Numeric Function (Math)

- **ABS/Absolute**

Untuk sintaks absolute saya menggunakan sintaks yang mirip seperti modul, tapi saya membedakan jadi saya mengselect table tot_cred, setelah itu baru masukan sintaks seperti di modul untuk menampilkan colom nilai absolute dari tot_cred.

```
-- ABS
select tot_cred, ABS(tot_cred) as NilaiAbsolute from student;
```

Output



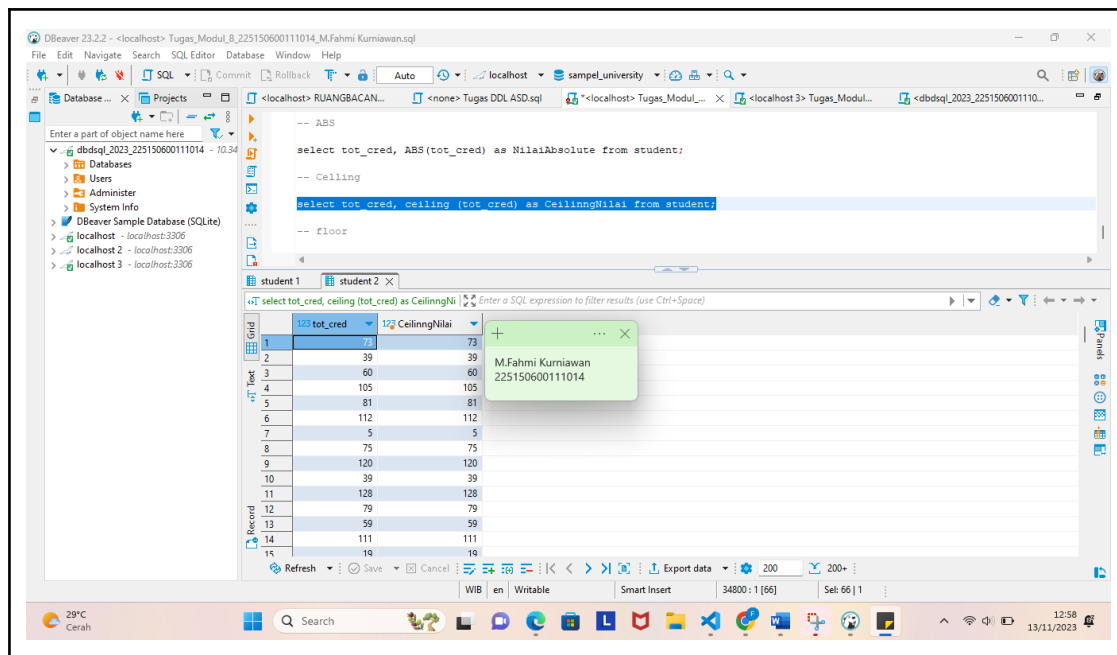
tot_cred	NilaiAbsolute
73	73
39	39
60	60
105	105
81	81
112	112
5	5
75	75
120	120
39	39
128	128
79	79
59	59

- **CEILING**

Untuk CEILING saya menggunakan kolom yang sama, yaitu kolom tot_cred, untuk perbedaan CEILING dan ABS, di ABS sebelum membuat kurung buka dan tutup menggunakan perintah ABS. Sedangkan di CEILING menggunakan perintah CEILING.

```
-- Ceiling
select tot_cred, ceiling (tot_cred) as CeilinggNilai from student;
```

Output

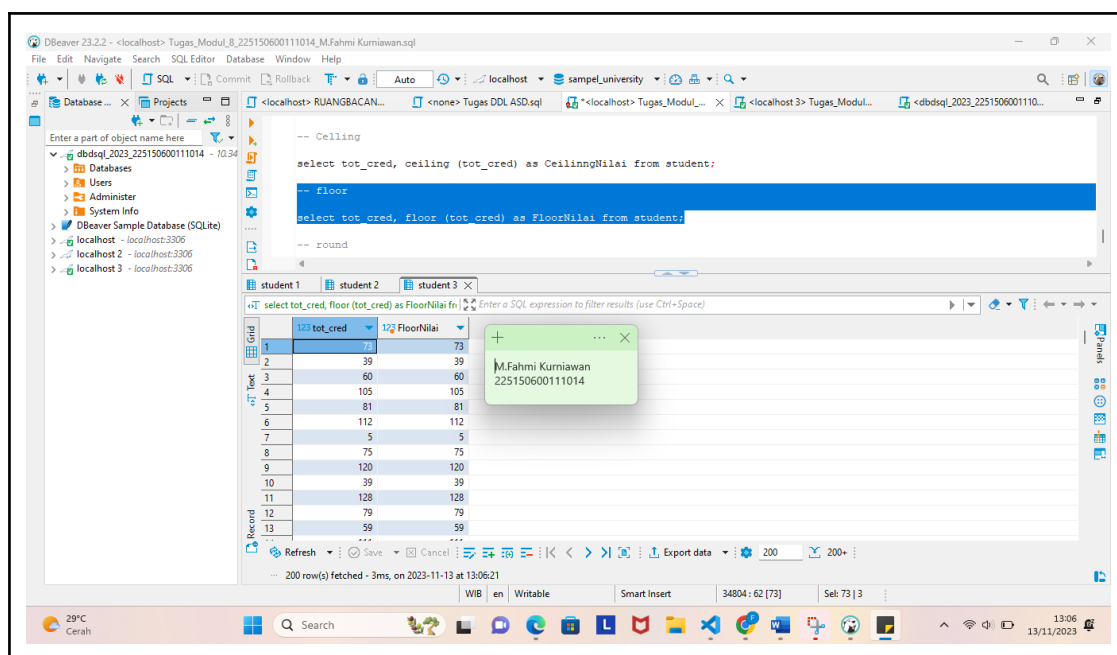


- Floor

Untuk perbedaan Floor dan ABS atau dengan yang lain sama yaitu sintaks perintah sebelum kurung buka kurung tutup, dan fungsinya.

```
-- floor
select tot_cred, floor (tot_cred) as FloorNilai from student;
```

Output



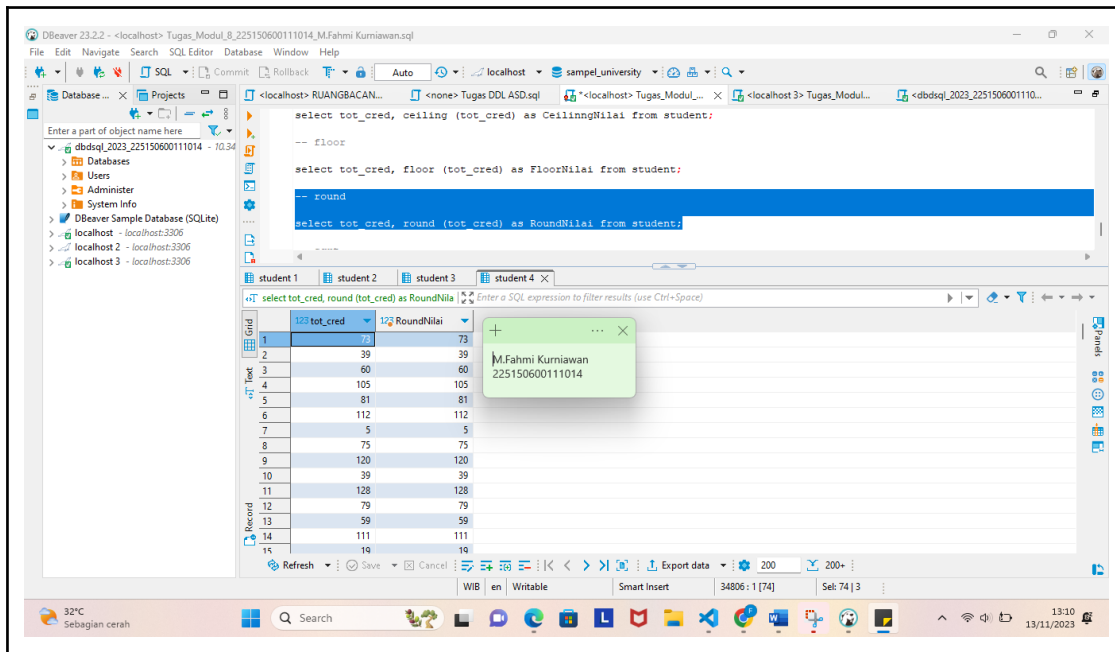
- Round

Untuk perbedaan Round dan ABS atau dengan yang lain sama yaitu sintaks perintah sebelum kurung buka kurung tutup, dan fungsinya.

```
-- round

select tot_cred, round (tot_cred) as RoundNilai from student;
```

Output



tot_cred	RoundNilai
73	73
39	39
60	60
105	105
81	81
112	112
5	5
75	75
120	120
39	39
128	128
79	79
59	59
111	111
10	10

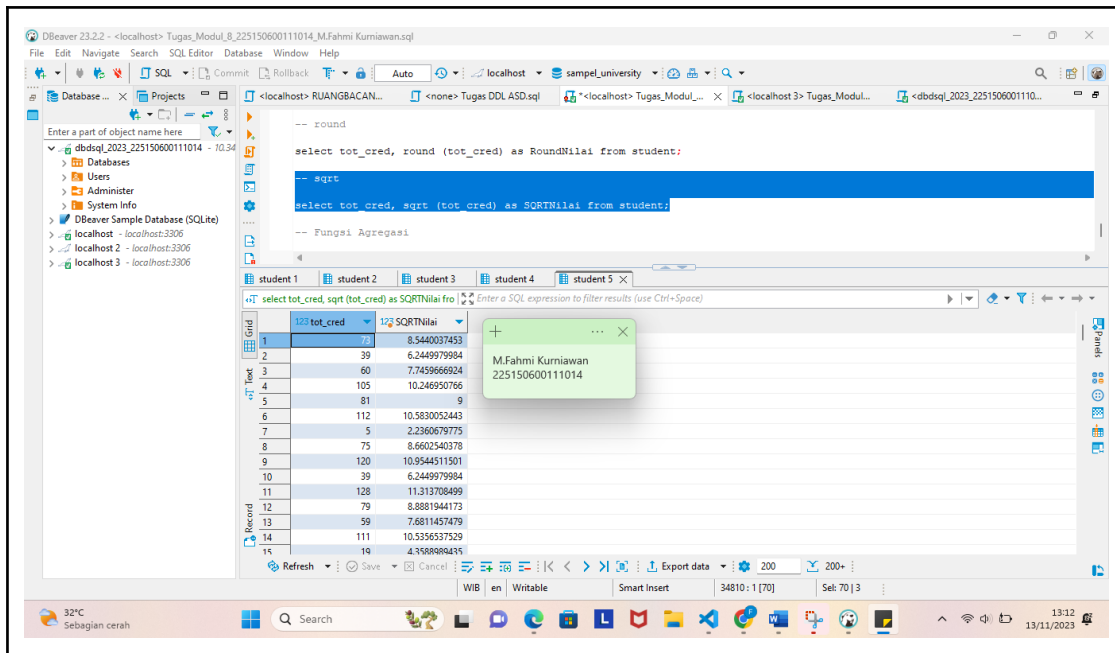
- SQRT

Untuk perbedaan SQRT dan ABS atau dengan yang lain sama yaitu sintaks perintah sebelum kurung buka kurung tutup, dan fungsinya.

```
-- sqrt

select tot_cred, sqrt (tot_cred) as SQRTNilai from student;
```

Output



DBeaver 23.2.2 - <localhost> Tugas_Modul_8_225150600111014_M.Fahmi Kurniawan.sql

File Edit Navigate Search SQL Editor Database Window Help

Database ... Projects

Enter a part of object name here

Database: dbdsql_2023_225150600111014 - 10.34

Users: Users, Administrator, System Info

DBEaver Sample Database (SQLite)

localhost - localhost:3306

localhost 2 - localhost:3306

localhost 3 - localhost:3306

SQL Editor:

```
-- round
select tot_cred, round (tot_cred) as RoundNilai from student;

-- sqrt
select tot_cred, sqrt (tot_cred) as SQRTNilai from student;

-- Fungsi Agregasi
```

student 1 student 2 student 3 student 4 student 5

select tot_cred, sqrt (tot_cred) as SQRTNilai fro

Enter a SQL expression to filter results (use Ctrl+Space)

Grid	1	2	3	4	5
1	tot_cred	73	8.5440037453		
2		39	6.2449979984		
3		60	7.7459666924		
4		105	10.246950766		
5		81	9		
6		112	10.5830052443		
7		5	2.2360679775		
8		75	8.6602540378		
9		120	10.9544511501		
10		39	6.2449979984		
11		128	11.313708499		
12		79	8.8881944173		
13		59	7.6811457479		
14		111	10.535657329		
15		19	4.3588986495		

Record

Refresh Save Cancel

WIB en Writable Smart Insert 34810 : 1 [70] Sel: 70 | 3

32°C Sebagian cerah

Search

13:12 13/11/2023

C.

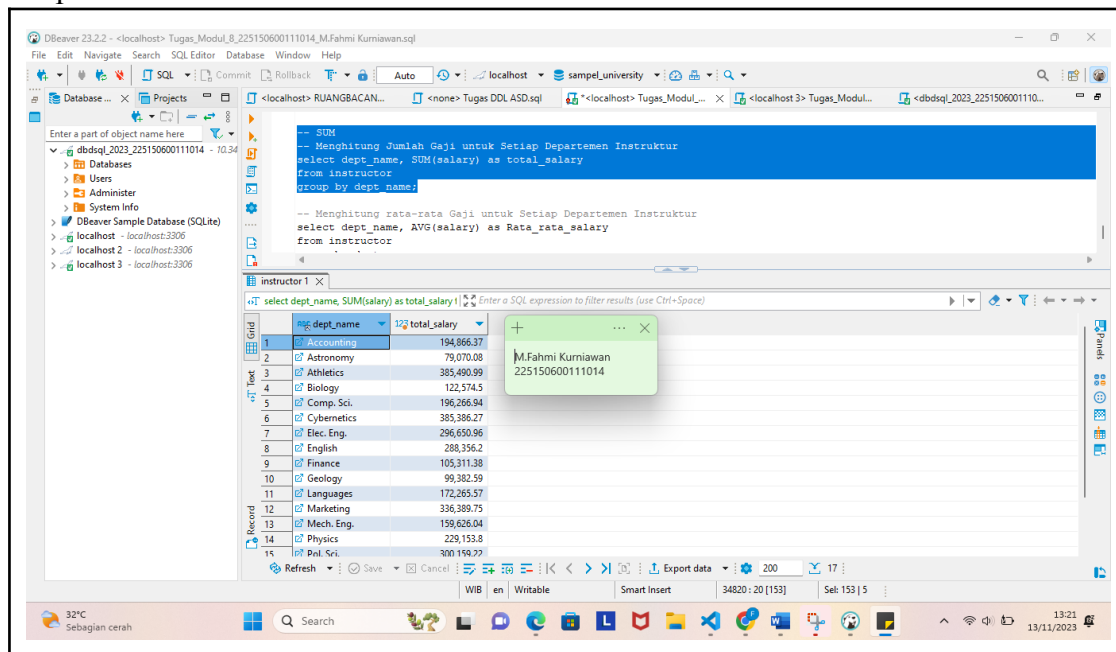
D. Fungsi Agregasi

- SUM

Untuk sintaks SUM mirip-mirip dengan contoh di modul namun saya ada perbedaan dimana saya memunculkan kolom lain sebelum melakukan perintah sum. Disini saya menggunakan perintah SUM untuk Menghitung Jumlah Gaji untuk Setiap Departemen Instruktur. pertama saya memunculkan kolom dept_name/nama departemen lalu saya melakukan sintaks SUM seperti dibawah ini dengan From Table Instruktur, saya menggunakan group by untuk memunculkan semua departement.

```
-- SUM
-- Menghitung Jumlah Gaji untuk Setiap Departemen Instruktur
select dept_name, SUM(salary) as total_salary
from instructor
group by dept_name;
```

Output



The screenshot shows the DBeaver SQL Editor interface. The SQL Editor window displays the following query:

```
-- SUM
-- Menghitung Jumlah Gaji untuk Setiap Departemen Instruktur
select dept_name, SUM(salary) as total_salary
from instructor
group by dept_name;
```

Below the query, the output is displayed in a table with the following data:

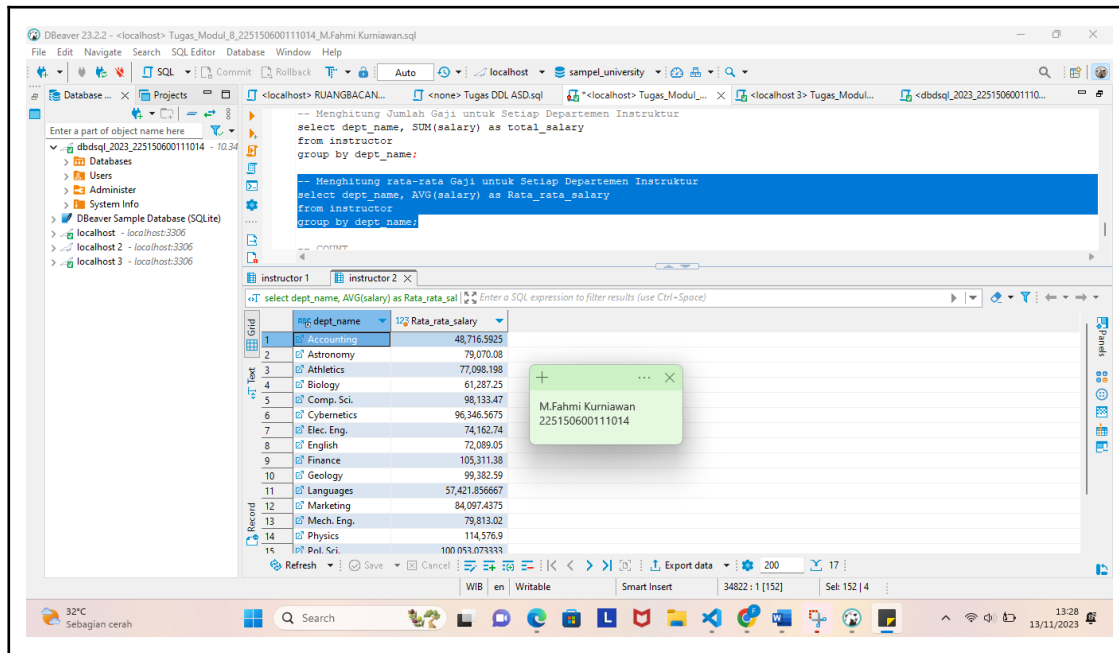
dept_name	total_salary
1	194,066.37
2	79,070.08
3	385,490.99
4	122,574.5
5	196,266.94
6	385,386.27
7	296,650.96
8	288,356.2
9	105,311.38
10	99,382.59
11	172,265.57
12	336,389.75
13	159,626.04
14	229,153.8
15	300,159.29

- AVG

Untuk AVG dan SUM sintaksnya sebenarnya hampir mirip kalau SUM sebelumnya mengetikkan perintah SUM sebelum kurung buka, sedangkan untuk AVG mengetikkan perintah AVG sebelum kurung buka. AVG memiliki fungsi untuk menghitung rata-rata nilai data dari suatu kolom yang bertipe data numeric. Perintah AVG saya, saya gunakan untuk Mengitung Rata-rata Gaji Instruktur di Setiap Departement.

```
-- Menghitung rata-rata Gaji untuk Setiap Departemen Instruktur
select dept_name, AVG(salary) as Rata_rata_salary
from instructor
group by dept_name;
```

Output



dept_name	Rata_rata_salary
Accounting	48,716.5925
Astronomy	79,070.08
Athletics	77,098.198
Biology	61,287.25
Comp. Sci.	96,133.47
Cybernetics	96,346.5675
Elec. Eng.	74,162.74
English	72,089.05
Finance	105,311.38
Geology	99,382.59
Languages	57,421.856667
Marketing	84,097.4375
Mech. Eng.	79,813.02
Physics	114,576.9
Pol. Sci.	100,053.073333

● COUNT

Untuk Count memiliki format sintaks yang hampir mirip dengan SUM dan AVG. perbedaan hanya diperintahnya saja sebelum kurung buka. count Digunakan untuk menghitung banyak nya baris di suatu kolom. Sedang di sintaks COUNT saya berguna untuk Menghitung Jumlah Instruktur dalam Setiap Departemen.

```
-- COUNT
-- Menghitung Jumlah Instruktur dalam Setiap Departemen
select dept_name, COUNT(ID) as jumlah_instruktur
from instructor
group by dept_name;
```

Output

dept_name	jumlah_instruktur
Accounting	1
Astronomy	1
Athletics	1
Biology	1
Comp. Sci.	1
Cybernetics	1
Elec. Eng.	1
English	1
Finance	1
Geology	1
Languages	1
Marketing	1
Mech. Eng.	1
Physics	1
Pol. Sci.	1