Database Project

Team Members : Kazi Huq,Mushfiqul Islam

Submitted To : Prof. Dr. Praveen Kumar

What is our project about?

-Our project is all based on **Sales Database Management System**.

What are the key concepts of this project?

-Sales Database Management System is highly effective on tracking records of big sales data, inserting, updating or deleting employees', supervisors' as well as merchandise inventory/refill data. Mammoth and renowned sales industries are mostly depending on Sales Database Management System to expanding their sales, managing a vast number of employees' personal info/paychecks as well as the data security of their industry.

What are the entities of our database?

-In order to build Sales Management Database, we are going to use seven entities. Those are given as following :

1. Employee

2.Management

3.Head Office

4.Dependent

5.Merchandise

6.Stores

7.Paycheck

8.Transaction

The list of all the entity data types as following :

Strong entity types :

-**Employee** (empl_id, Fname, Lname, address, empl_ssn, position)

-**Management** (super_id, firstName, lastName, address, super_ssn, super_position, super_Dno)

-**Head Office** (HR_id, HR_name, HR_ssn, PR_id, PR_name, PR_ssn, Logstc_id, Logstc_name, Logstc_ssn)

-**Merchandise** (Merch_no, Mens, Ladies, Kids, Price, Stocked, Sold)

-**Stores** (Store_No, Location, Super_id, Super_ssn, Revenue)

-**Paycheck** (E_id, E_ssn, Salary, Hour, Position, S_id, S_ssn, Salary, Hour, Position)

-**Transaction** (payment_id, empl_id, customer_id, Total Price, Total Due, payment_date, payment_time, store_no)

Weak entity type :

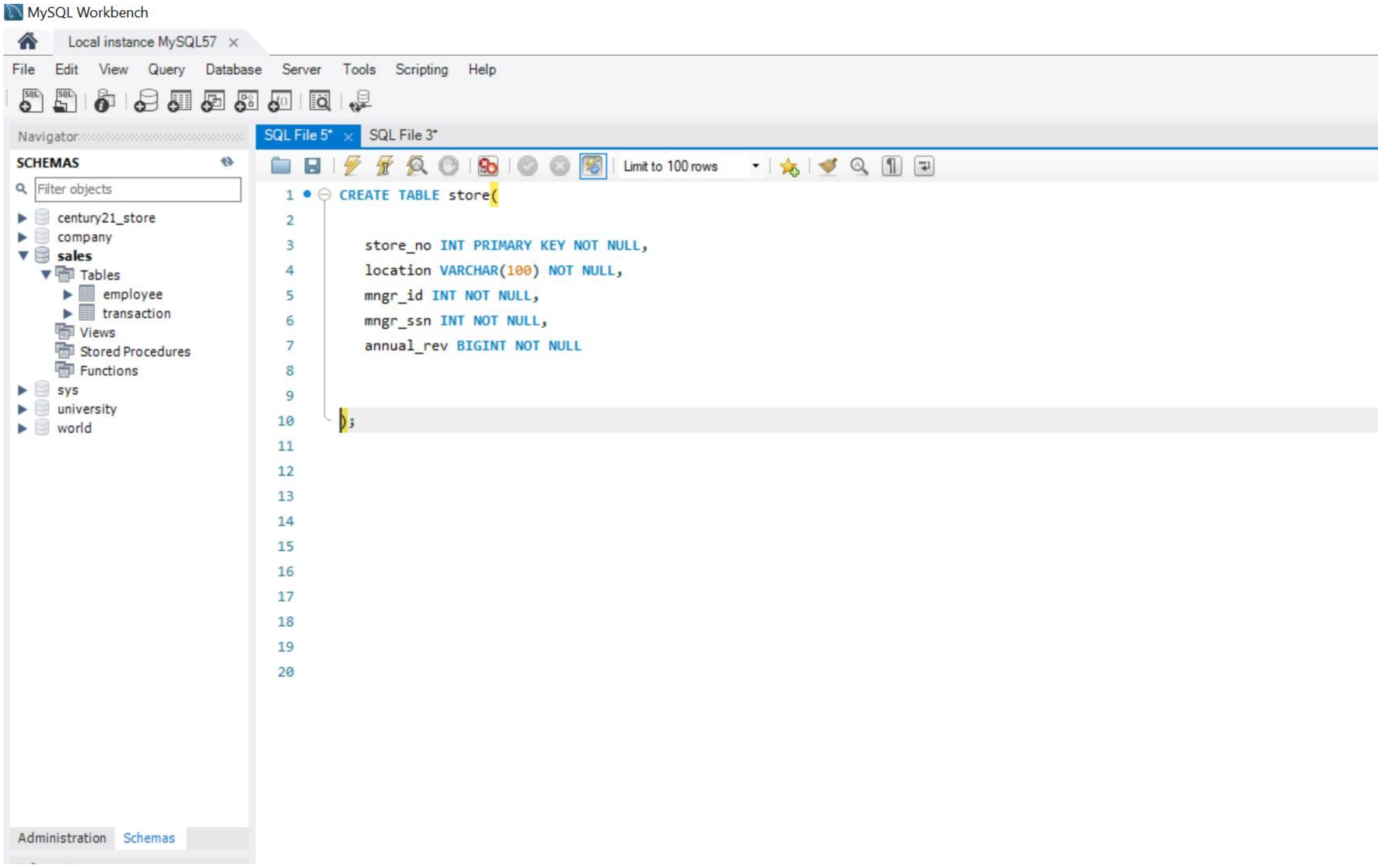
-Dependent(empl_id, Dpndnt_name, Relation, super_id, Dpndnt_name, Relation)

The steps and procedures of how we **CREATED TABLES, ADDED BULK DATA** from local storage are given as following (with screenshot) :

1) Creating the table : CREATE TABLE table_name(

Column_name data_type primary_key(only if it is unique/for referential integrity) Null/Not Null

);



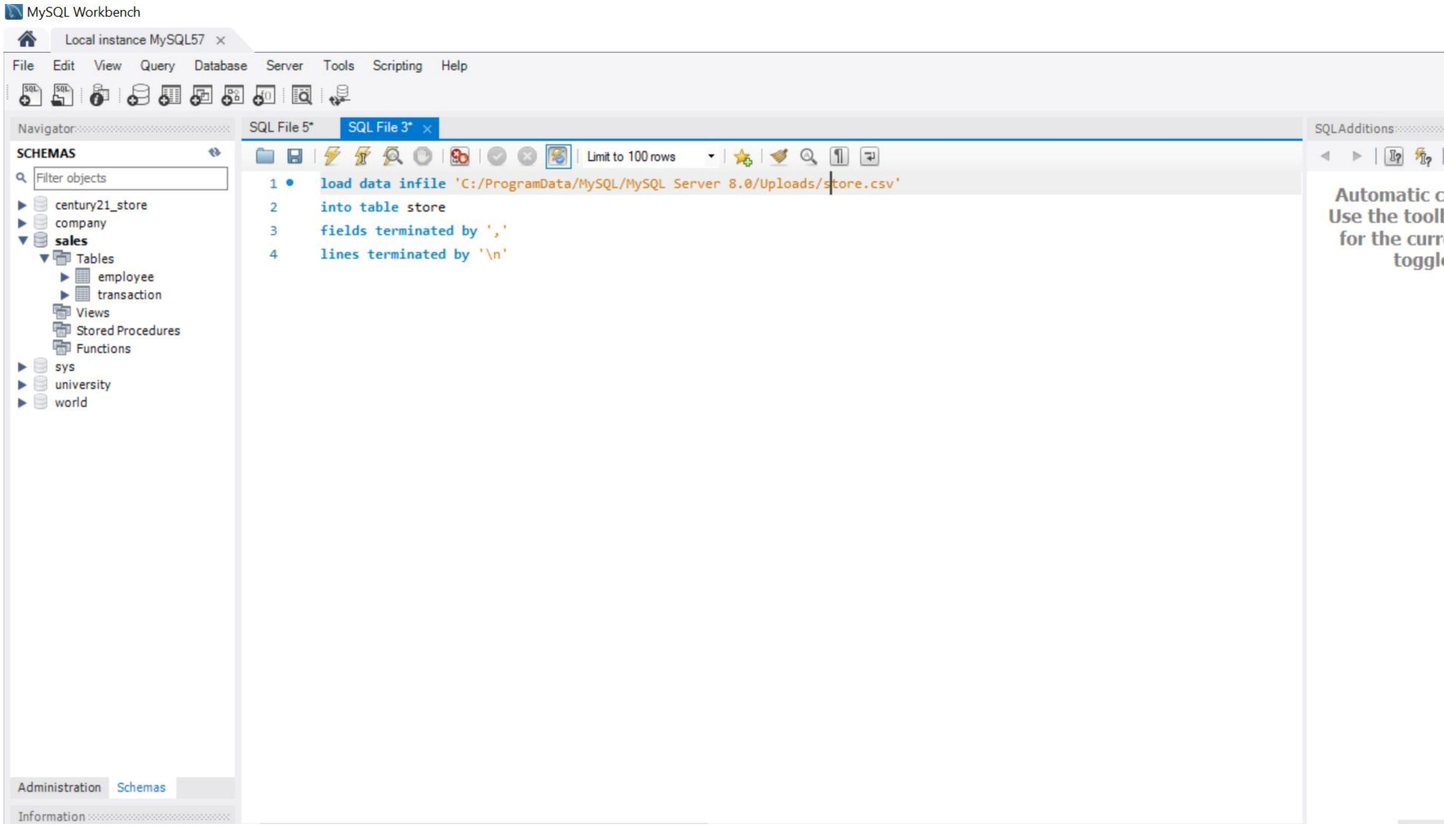
```
CREATE TABLE store(

    store_no INT PRIMARY KEY NOT NULL,
    location VARCHAR(100) NOT NULL,
    mngr_id INT NOT NULL,
    mngr_ssn INT NOT NULL,
    annual_rev BIGINT NOT NULL


);
```

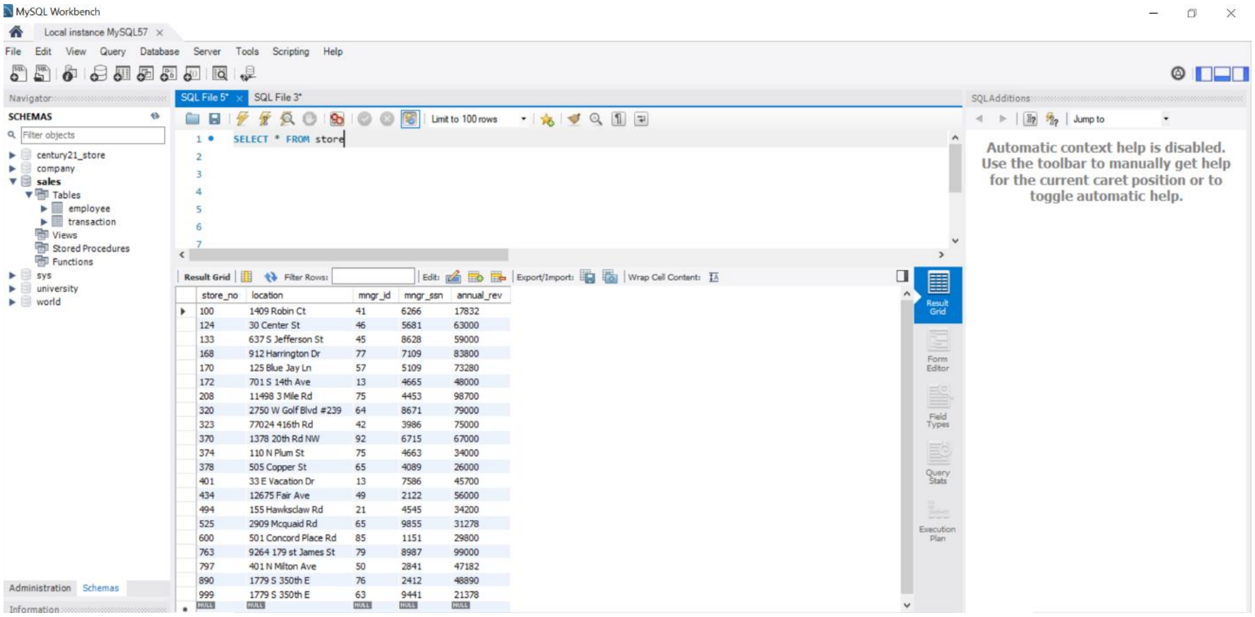
2) Adding Bulk Data : load data infile ‘/local disc full path/’

Into table table_name



```
load data infile 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/store.csv'
into table store
fields terminated by ','
lines terminated by '\n'
```

3)SELECT the table : select * from table_name



```
SELECT * FROM store
```

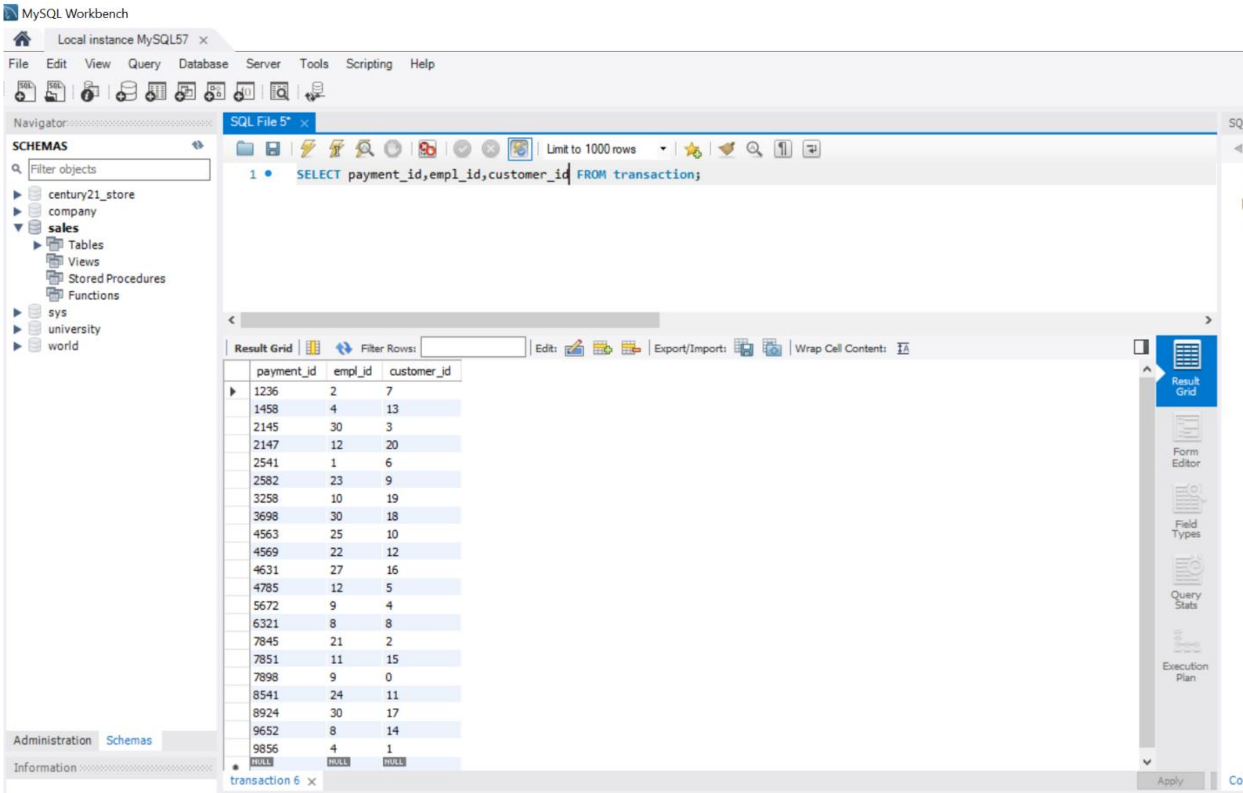
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

store_no	location	mngr_id	mngr_ssn	annual_rev
100	1409 Robin Ct	41	6266	17832
124	30 Center St	46	5681	63000
133	637 S Jefferson St	45	8628	59000
168	912 Harrington Dr	77	7109	83800
170	125 Blue Jay Ln	57	5109	73280
172	701 S 14th Ave	13	4665	48000
208	11498 3 Mile Rd	75	4453	98700
320	2750 W Golf Blvd #239	64	8671	79000
323	77024 416th Rd	42	3986	75000
370	1378 20th Rd NW	92	6715	67000
374	110 N Plum St	75	4663	34000
378	505 Copper St	65	4089	26000
401	33 E Vacation Dr	13	7586	45700
434	12675 Fair Ave	49	2122	56000
494	155 Hawksclaw Rd	21	4545	34200
525	2909 Mcquaid Rd	65	9855	31278
600	501 Concord Place Rd	85	1151	29800
763	9264 179 st James St	79	8987	99000
797	401 N Milton Ave	50	2841	47182
890	1779 S 350th E	76	2412	48890
999	1779 S 350th E	63	9441	21378
NULL	NULL	NULL	NULL	NULL

Listing all of the **SQL Statement Fundamentals**, **Advanced SQL Commands** and all other **Useful Queries** to retrieve data with different sort of conditions (Example is given with each statement and screenshot)

- SQL Select Statements and useful Conditions :

- SELECT payment_id,empl_id,customer_id FROM Transaction



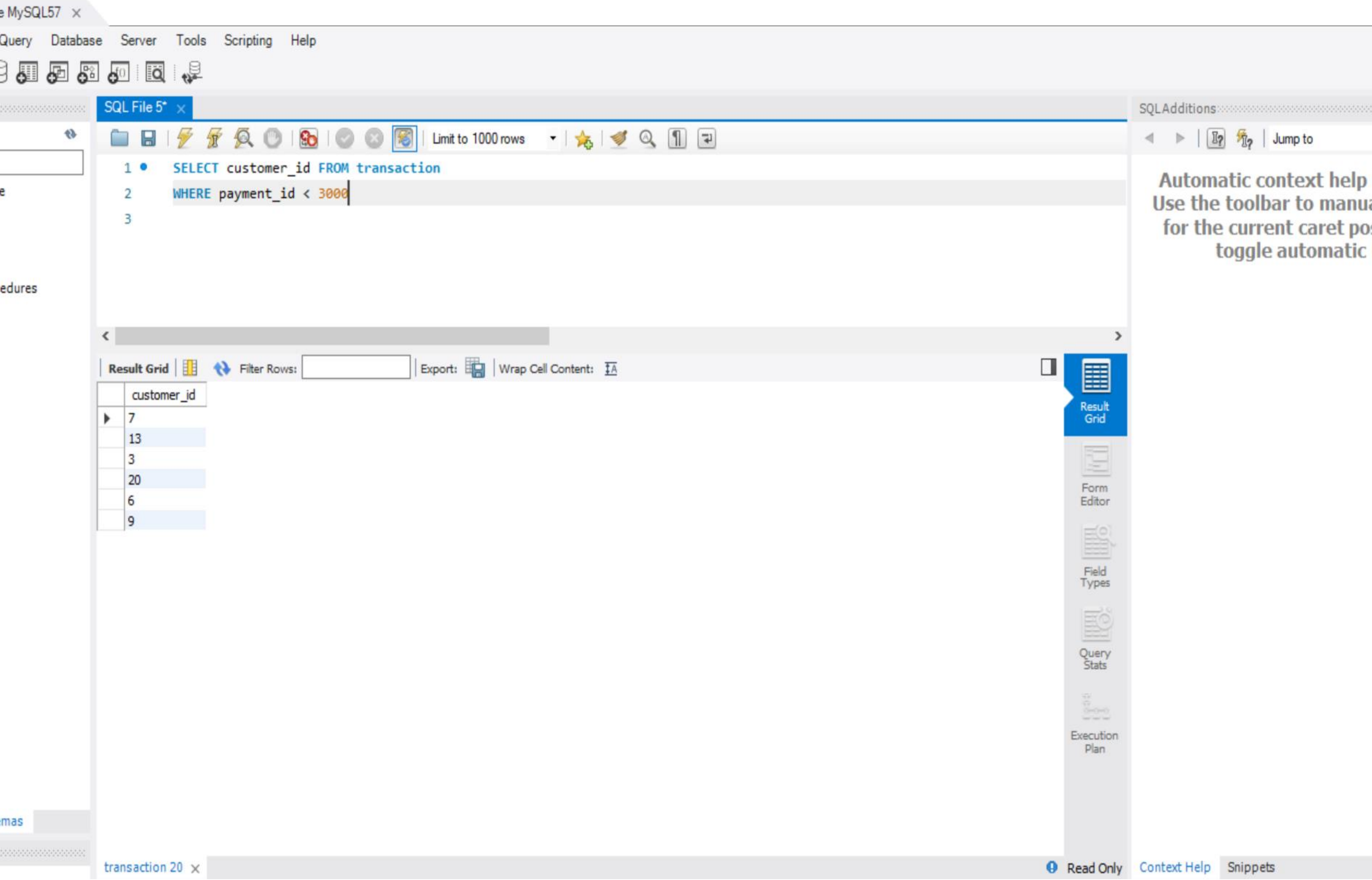
```
SELECT payment_id,empl_id,customer_id FROM transaction;
```

payment_id	empl_id	customer_id
1236	2	7
1458	4	13
2145	30	3
2147	12	20
2541	1	6
2582	23	9
3258	10	19
3698	30	18
4563	25	10
4569	22	12
4631	27	16
4785	12	5
5672	9	4
6321	8	8
7845	21	2
7851	11	15
7898	9	0
8541	24	11
8924	30	17
9652	8	14
9856	4	1
NULL	NULL	NULL

❖ WHERE :

SELECT customer_id FROM transaction

WHERE payment_id < 3000



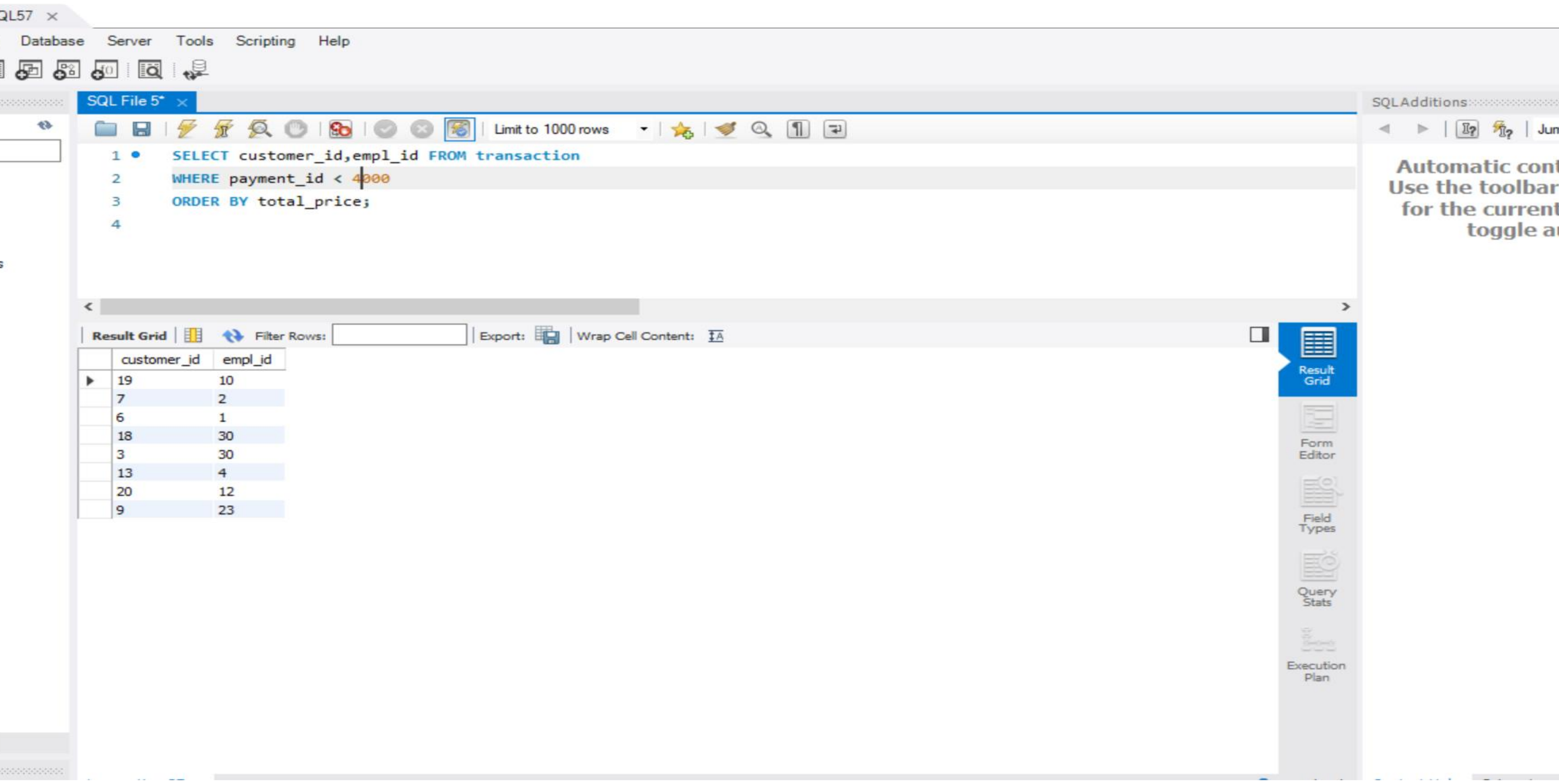
```
SELECT customer_id FROM transaction
WHERE payment_id < 3000
```

```
customer_id
7
13
3
20
6
9
```

❖ ORDER BY :

```
SELECT customer_id,empl_id FROM transaction
WHERE payment_id < 4000
```

ORDER BY total_price



```
SELECT customer_id,empl_id FROM transaction
WHERE payment_id < 4000
ORDER BY total_price;
```

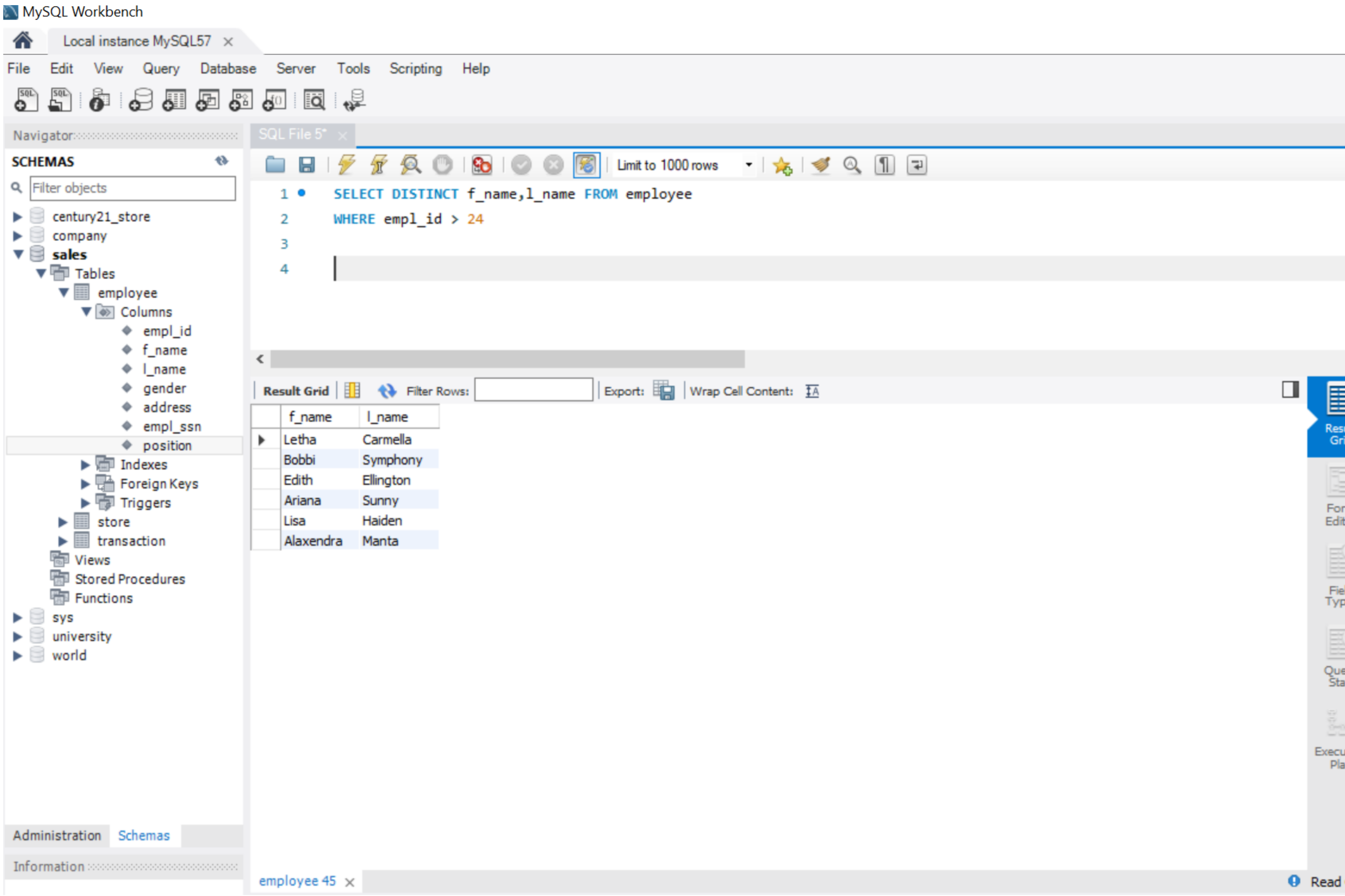
customer_id	empl_id
19	10
7	2
6	1
18	30
3	30
13	4
20	12
9	23

Automatic cont
Use the toolbar
for the current
toggle au

❖ <u>DISTINCT :</u>

SELECT DISTINCT f_name,l_name FROM employee

WHERE empl_id > 24

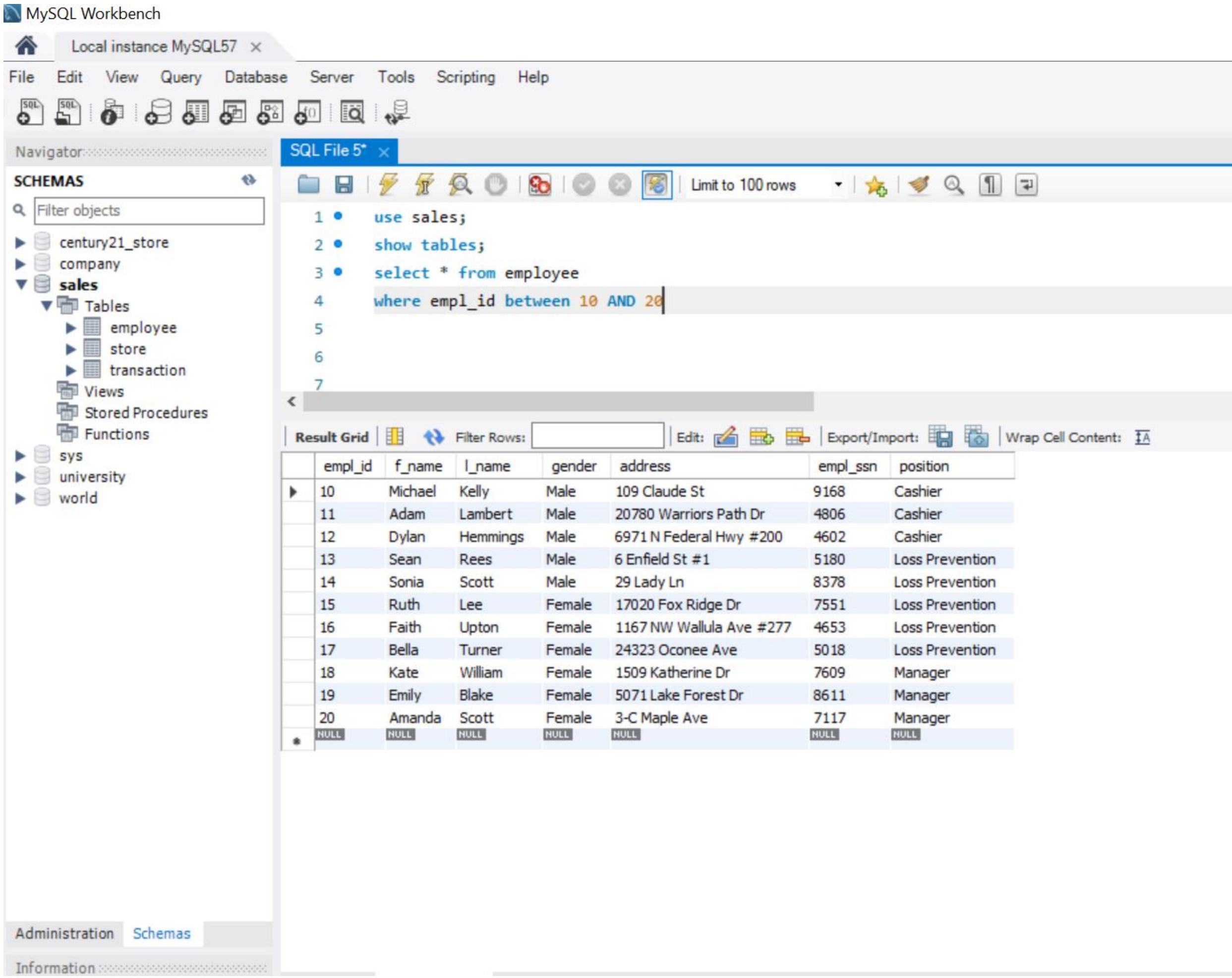


```
SELECT DISTINCT f_name,l_name FROM employee
WHERE empl_id > 24
```

f_name	l_name
Letha	Carmella
Bobbi	Symphony
Edith	Ellington
Ariana	Sunny
Lisa	Haiden
Alaxendra	Manta

❖ BETWEEN :

```
select * from employee
where empl_id between 10 AND 20
```

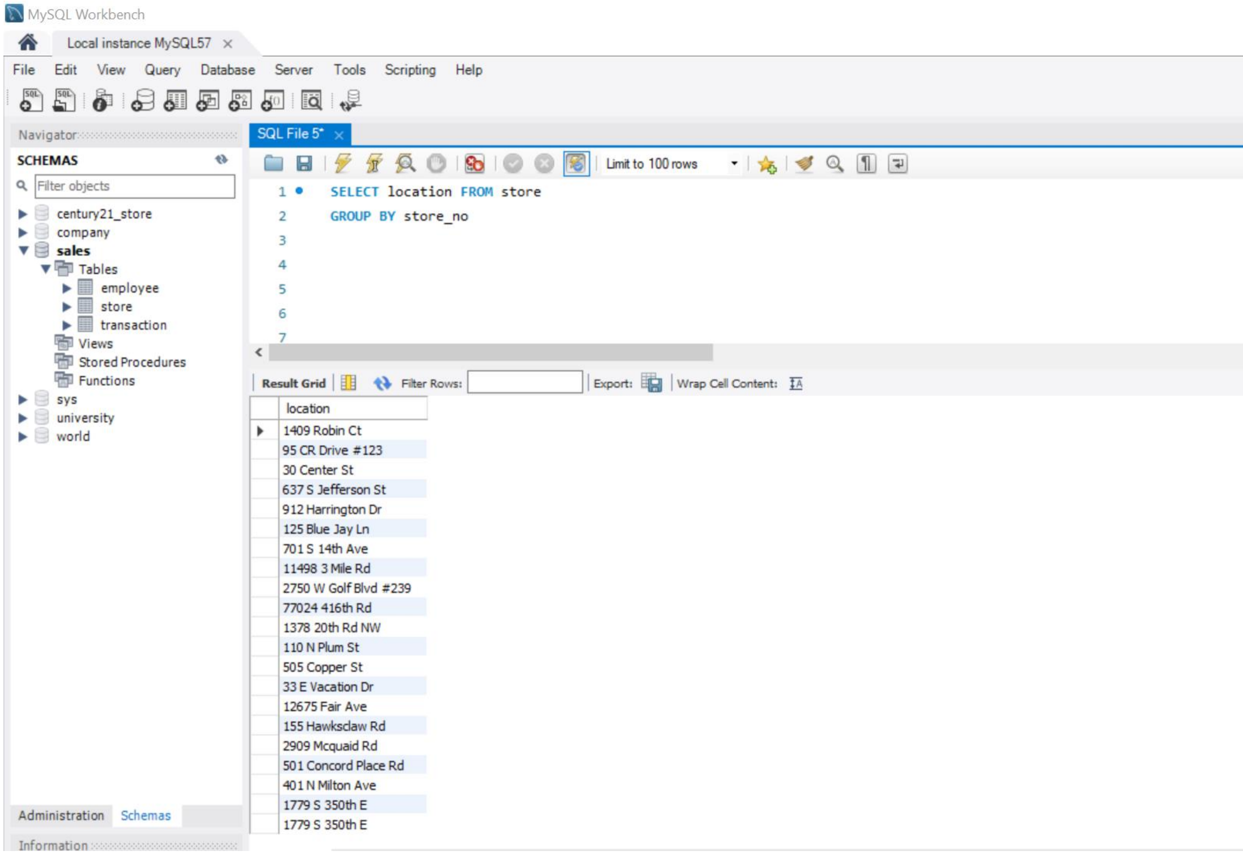


```
use sales;
show tables;
select * from employee
where empl_id between 10 AND 20
```

empl_id	f_name	l_name	gender	address	empl_ssn	position
10	Michael	Kelly	Male	109 Claude St	9168	Cashier
11	Adam	Lambert	Male	20780 Warriors Path Dr	4806	Cashier
12	Dylan	Hemmings	Male	6971 N Federal Hwy #200	4602	Cashier
13	Sean	Rees	Male	6 Enfield St #1	5180	Loss Prevention
14	Sonia	Scott	Male	29 Lady Ln	8378	Loss Prevention
15	Ruth	Lee	Female	17020 Fox Ridge Dr	7551	Loss Prevention
16	Faith	Upton	Female	1167 NW Wallula Ave #277	4653	Loss Prevention
17	Bella	Turner	Female	24323 Oconee Ave	5018	Loss Prevention
18	Kate	William	Female	1509 Katherine Dr	7609	Manager
19	Emily	Blake	Female	5071 Lake Forest Dr	8611	Manager
20	Amanda	Scott	Female	3-C Maple Ave	7117	Manager
NULL	NULL	NULL	NULL	NULL	NULL	NULL

❖ GROUP BY :

SELECT location FROM store
GROUP BY store_no

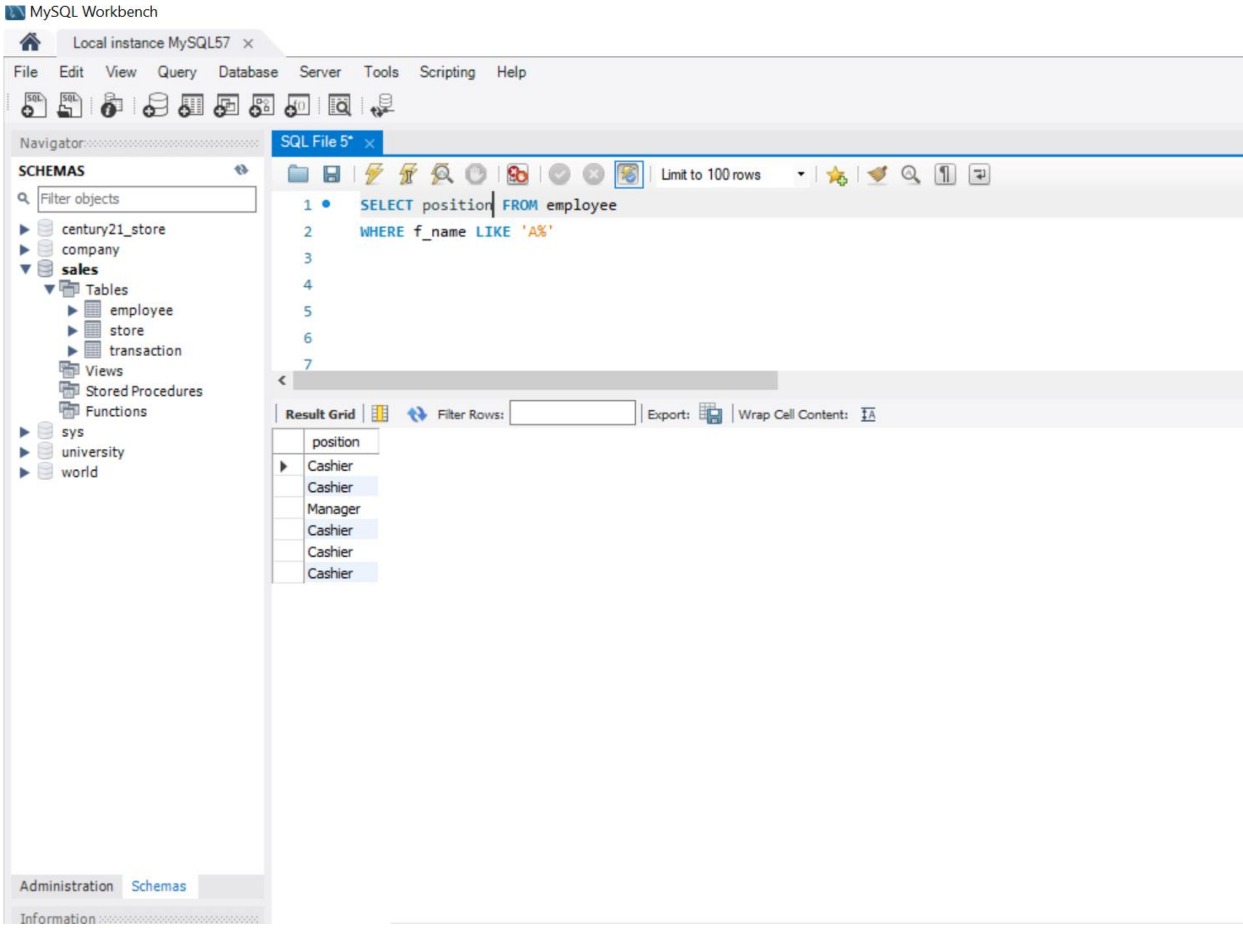


```
SELECT location FROM store
GROUP BY store_no
```

```
location
1409 Robin Ct
95 CR Drive #123
30 Center St
637 S Jefferson St
912 Harrington Dr
125 Blue Jay Ln
701 S 14th Ave
11498 3 Mile Rd
2750 W Golf Blvd #239
77024 416th Rd
1378 20th Rd NW
110 N Plum St
505 Copper St
33 E Vacation Dr
12675 Fair Ave
155 Hawksclaw Rd
2909 Mcquaid Rd
501 Concord Place Rd
401 N Milton Ave
1779 S 350th E
1779 S 350th E
```

❖ LIKE :

```
SELECT position FROM employee
WHERE f_name LIKE 'A%'
```

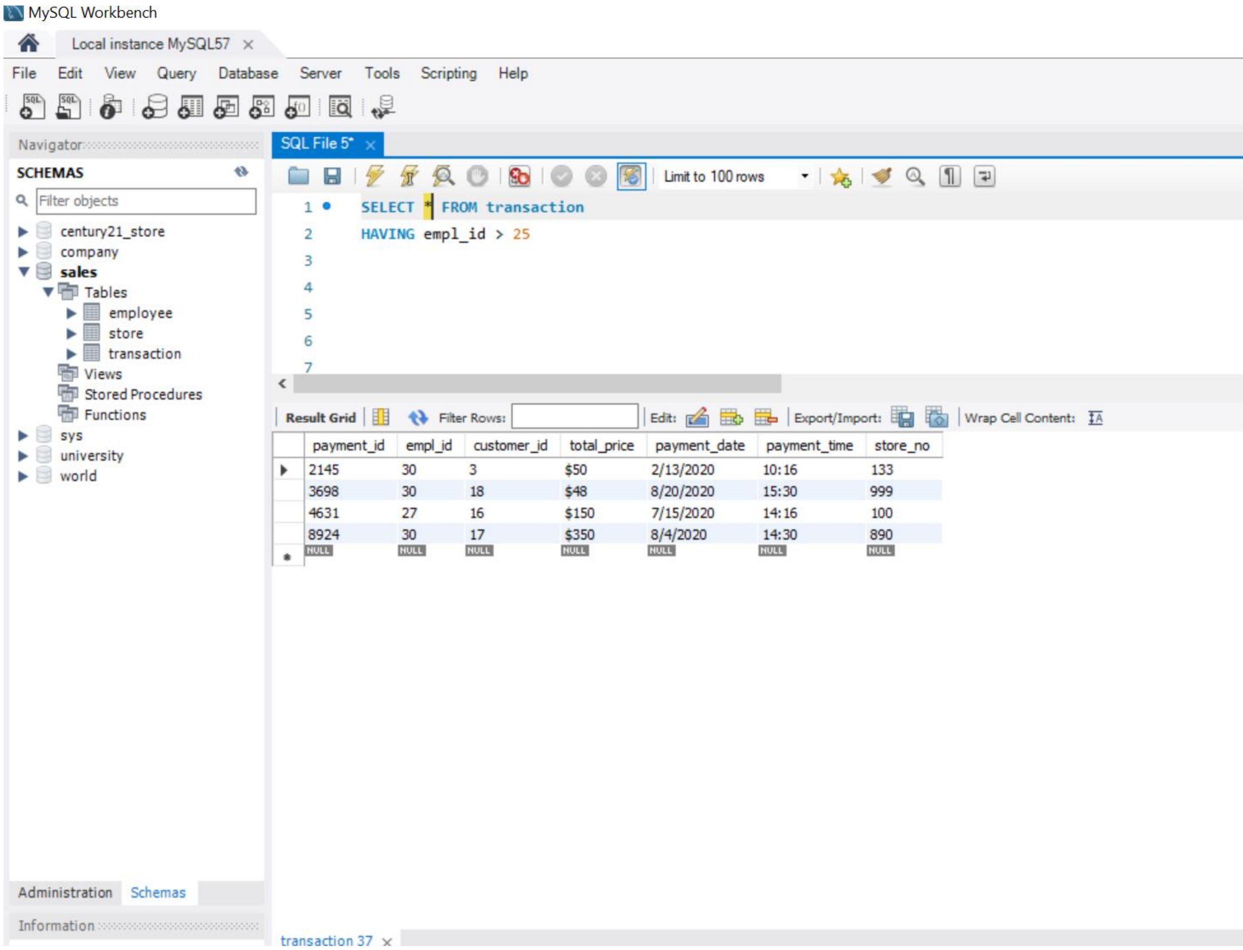


```sql
SELECT position FROM employee
WHERE f_name LIKE 'A%'
```

position
Cashier
Cashier
Manager
Cashier
Cashier
Cashier

- ❖ HAVING :

SELECT * FROM transaction
HAVING empl_id > 25



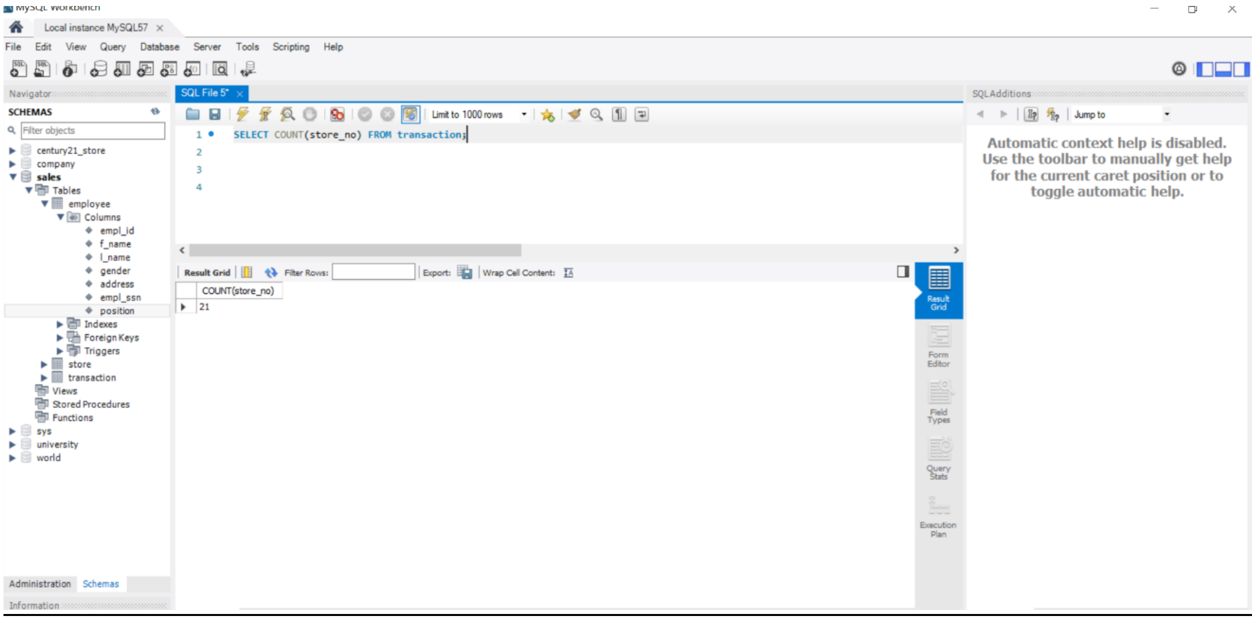
```
SELECT * FROM transaction
HAVING empl_id > 25
```

payment_id	empl_id	customer_id	total_price	payment_date	payment_time	store_no
2145	30	3	$50	2/13/2020	10:16	133
3698	30	18	$48	8/20/2020	15:30	999
4631	27	16	$150	7/15/2020	14:16	100
8924	30	17	$350	8/4/2020	14:30	890
NULL	NULL	NULL	NULL	NULL	NULL	NULL

➢ Aggregate Functions :

❖ COUNT :

SELECT COUNT(store_no) FROM transaction



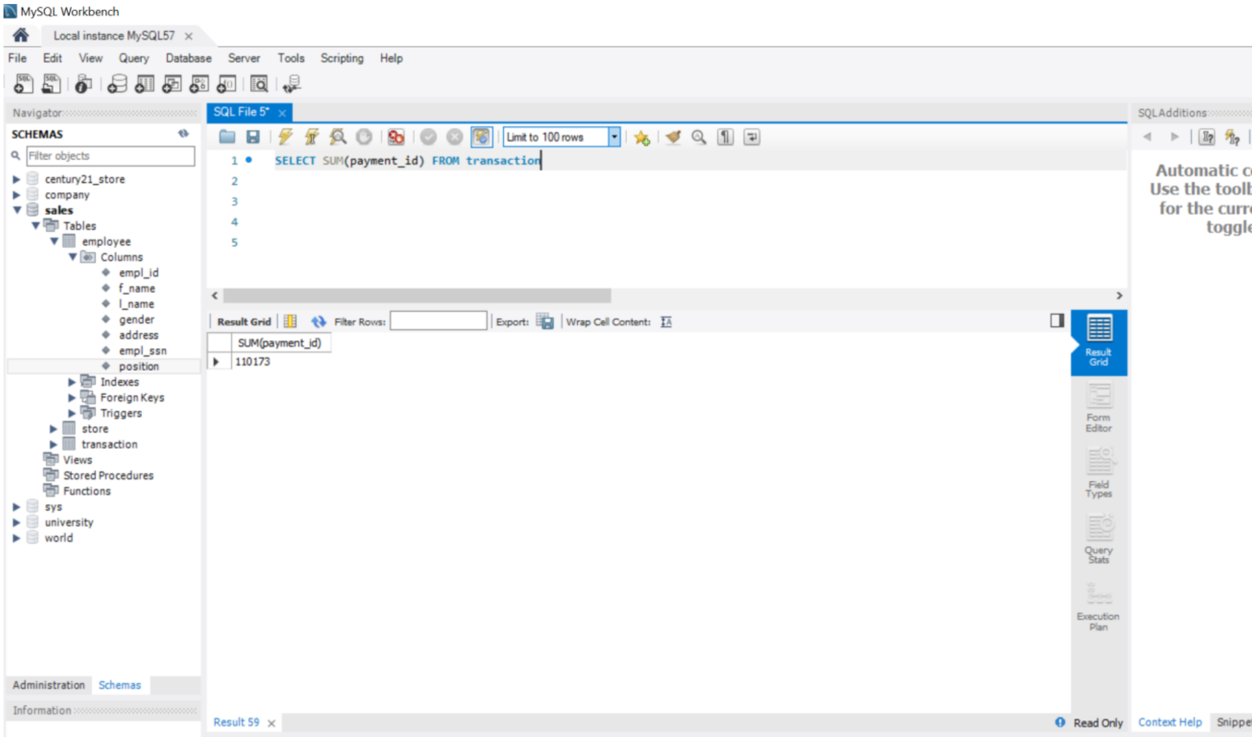
```
SELECT COUNT(store_no) FROM transaction;
```

```
COUNT(store_no)
21
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

❖ SUM :

SELECT SUM(payment_id) FROM transaction

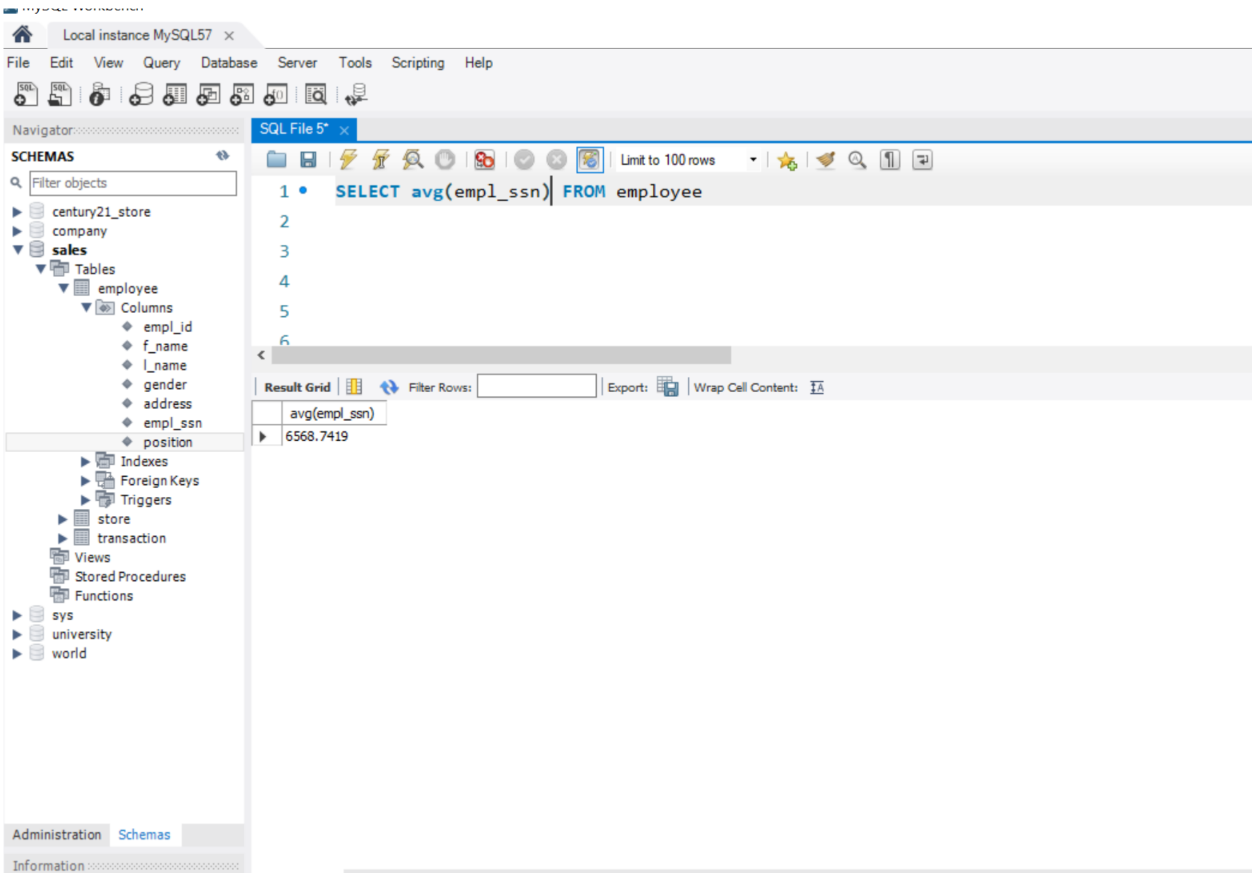


```
SELECT SUM(payment_id) FROM transaction
```

```
SUM(payment_id)
110173
```

❖ AVG :

SELECT avg(empl_ssn) FROM employee

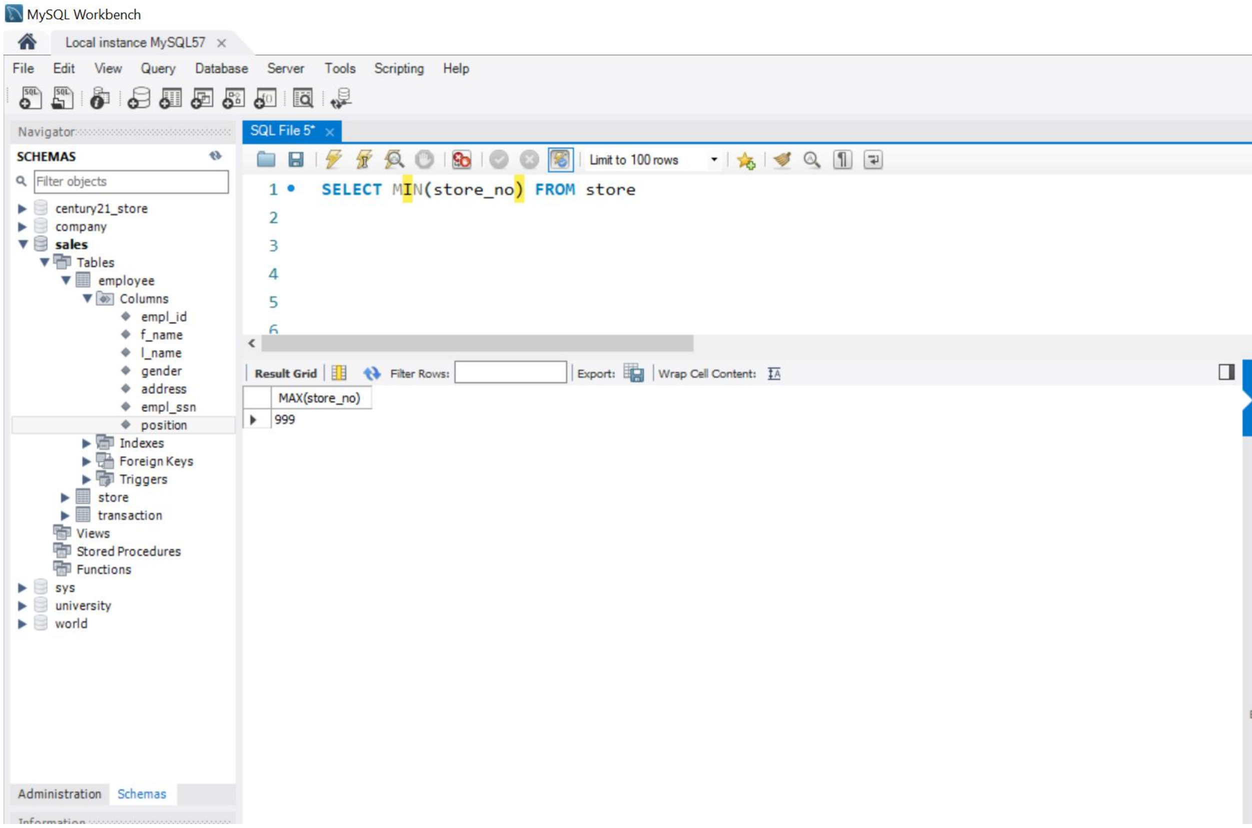


```
SELECT avg(empl_ssn) FROM employee
```

```
avg(empl_ssn)
6568.7419
```

❖ MIN :

SELECT MIN(store_no) FROM store

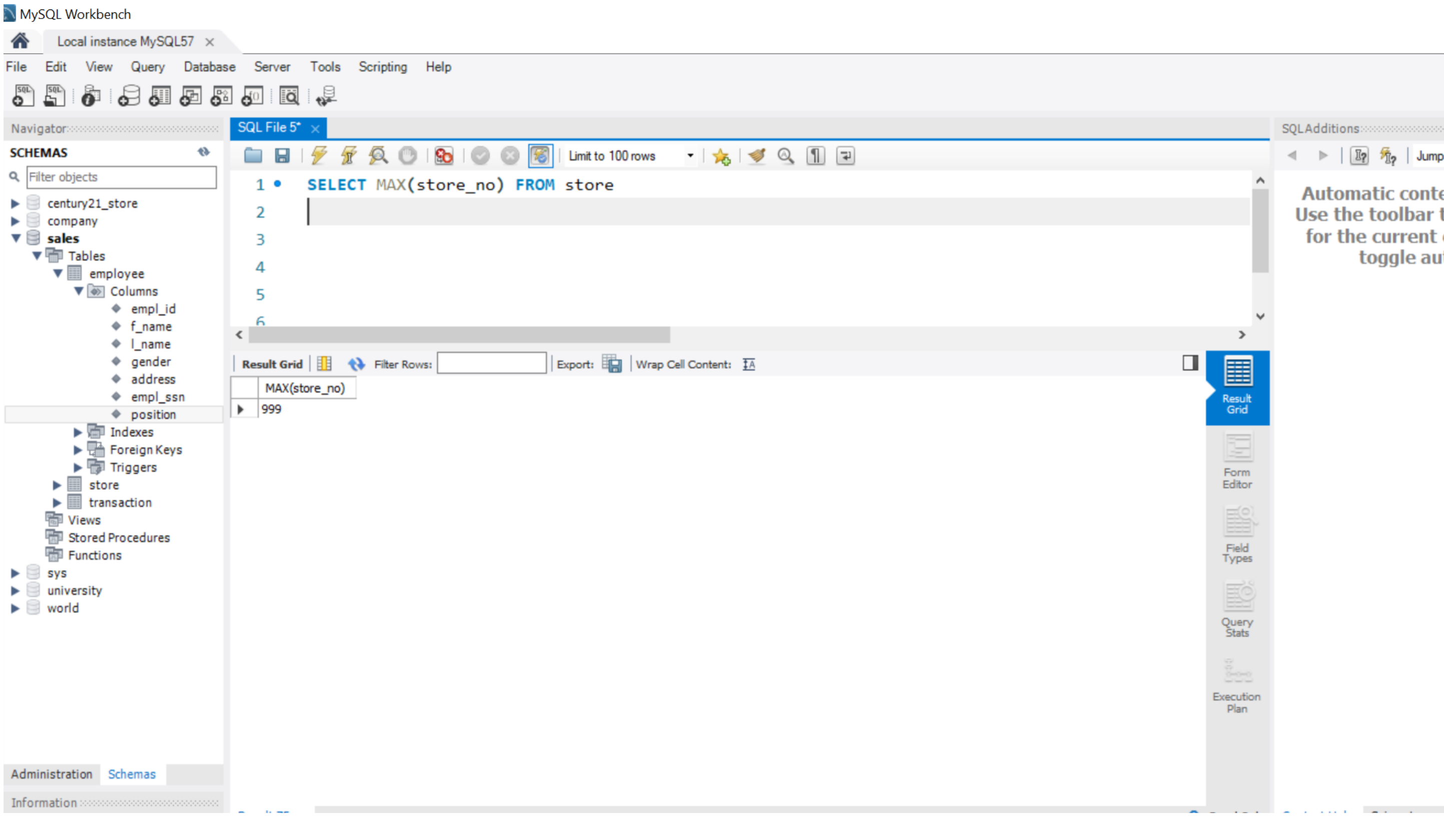


```
SELECT MIN(store_no) FROM store
```

```
MAX(store_no)
999
```

❖ MAX :

SELECT MAX(store_no) FROM store



```
SELECT MAX(store_no) FROM store
```

MAX(store_no)
999