

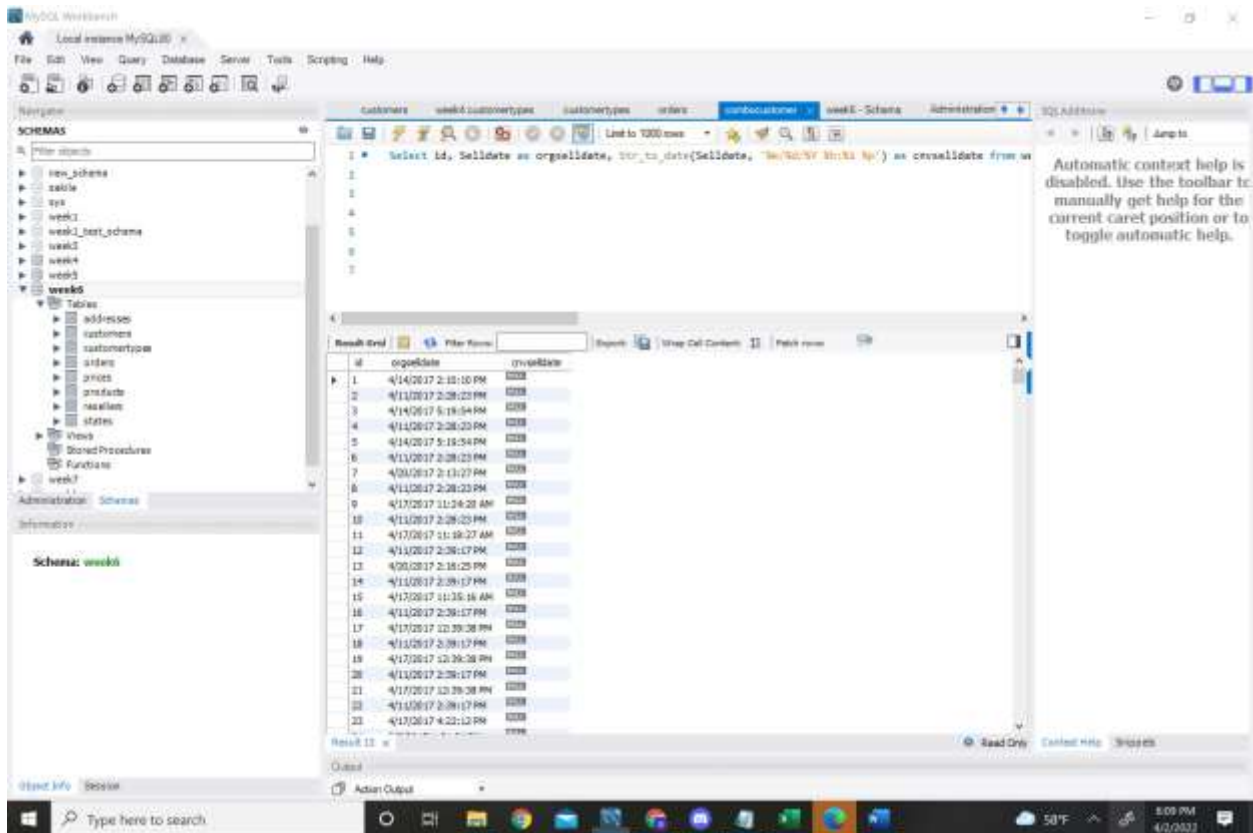
Week 6 Follow Along

```
Select * from week6.customers order by customerid asc;
```

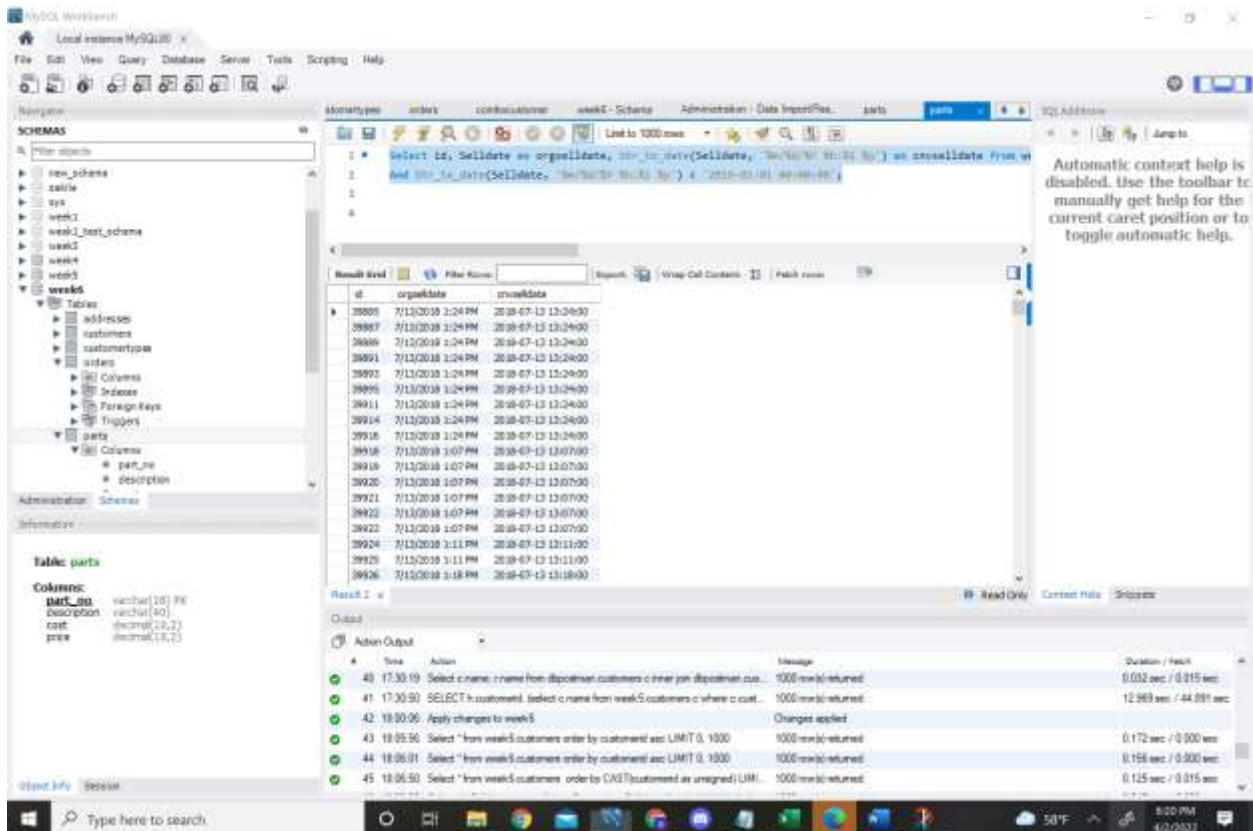
The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'week6' selected. The main query editor contains the SQL statement: `Select * from week6.customers order by customerid asc;`. The 'Results' pane shows a table with 6 columns: `customerid`, `name`, `searchname`, `address`, `address2`, and `city`. The results are ordered by `customerid` in ascending order. A tooltip on the right states: 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.'

customerid	name	searchname	address	address2	city
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					
31					
32					
33					
34					
35					
36					
37					
38					
39					
40					
41					
42					
43					
44					
45					
46					
47					
48					
49					
50					
51					
52					
53					
54					
55					
56					
57					
58					
59					
60					
61					
62					
63					
64					
65					
66					
67					
68					
69					
70					
71					
72					
73					
74					
75					
76					
77					
78					
79					
80					
81					
82					
83					
84					
85					
86					
87					
88					
89					
90					
91					
92					
93					
94					
95					
96					
97					
98					
99					
100					

```
Select * from week6.customers
order by CAST(customerid as unsigned);
```

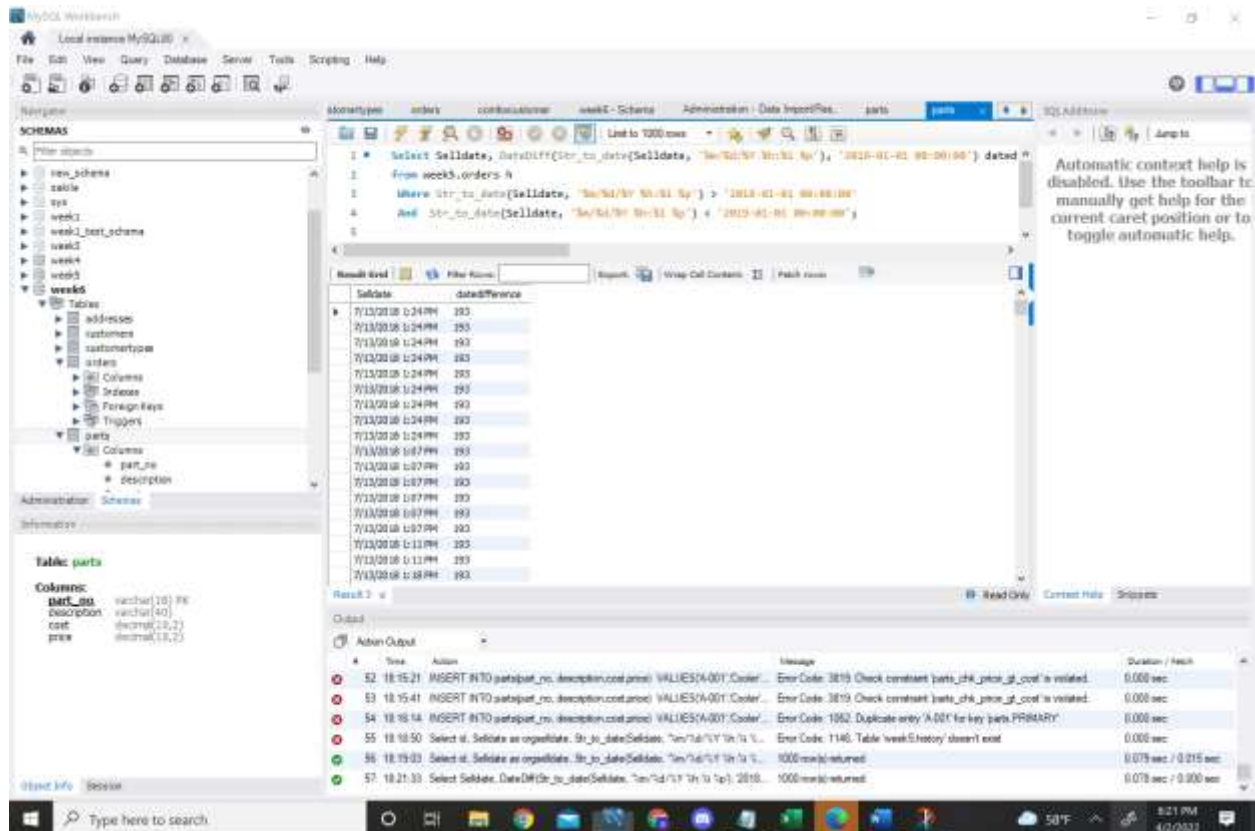
```
Select id, Selldate as orgselldate, Str_to_date(Selldate, '%m/%d/%Y %h:%i %p') as cnvselldate from week5.orders h Where Str_to_date(Selldate, '%m/%d/%Y %h:%i %p') > '2018-01-01 00:00:00'
And Str_to_date(Selldate, '%m/%d/%Y %h:%i %p') < '2019-01-01 00:00:00';
```



```

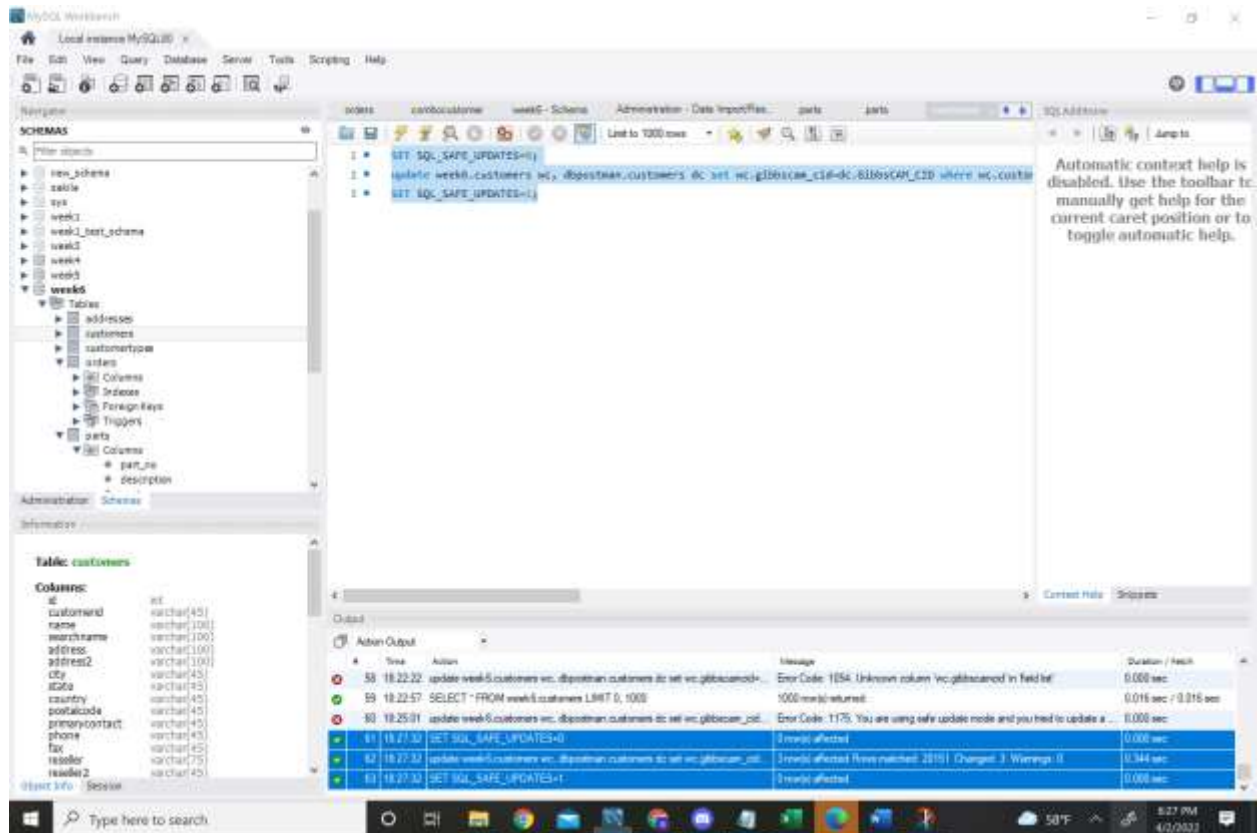
Select Selldate, DateDiff(Str_to_date(Selldate, '%m/%d/%Y %h:%i %p'), '2018-
01-01 00:00:00') datedifference
from week5.history h
Where Str_to_date(Selldate, '%m/%d/%Y %h:%i %p') > '2018-01-01 00:00:00'
And Str_to_date(Selldate, '%m/%d/%Y %h:%i %p') < '2019-01-01 00:00:00';

```

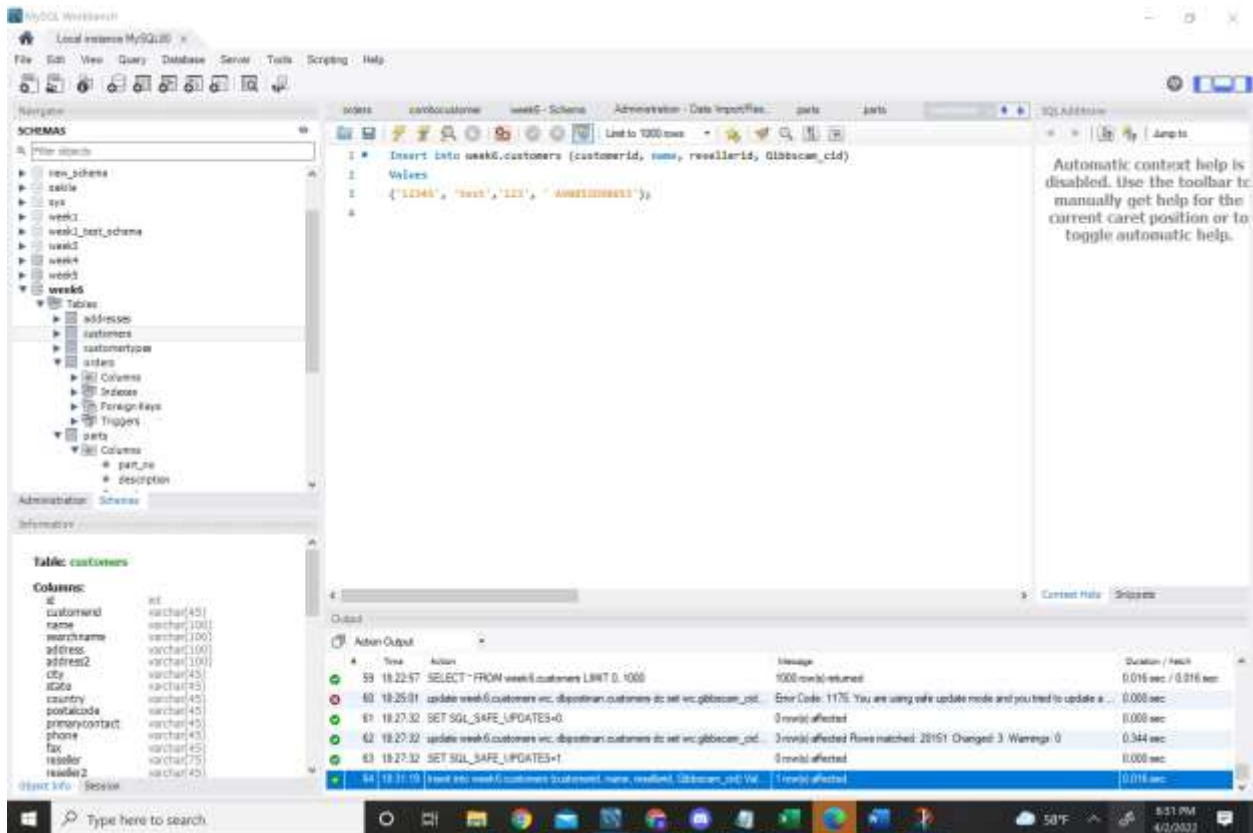


update week6.customers wc, dbpostman.customers dc set
wc.gibbscamcid=dc.GibbsCAM_CID where wc.customerid=dc.customerid;

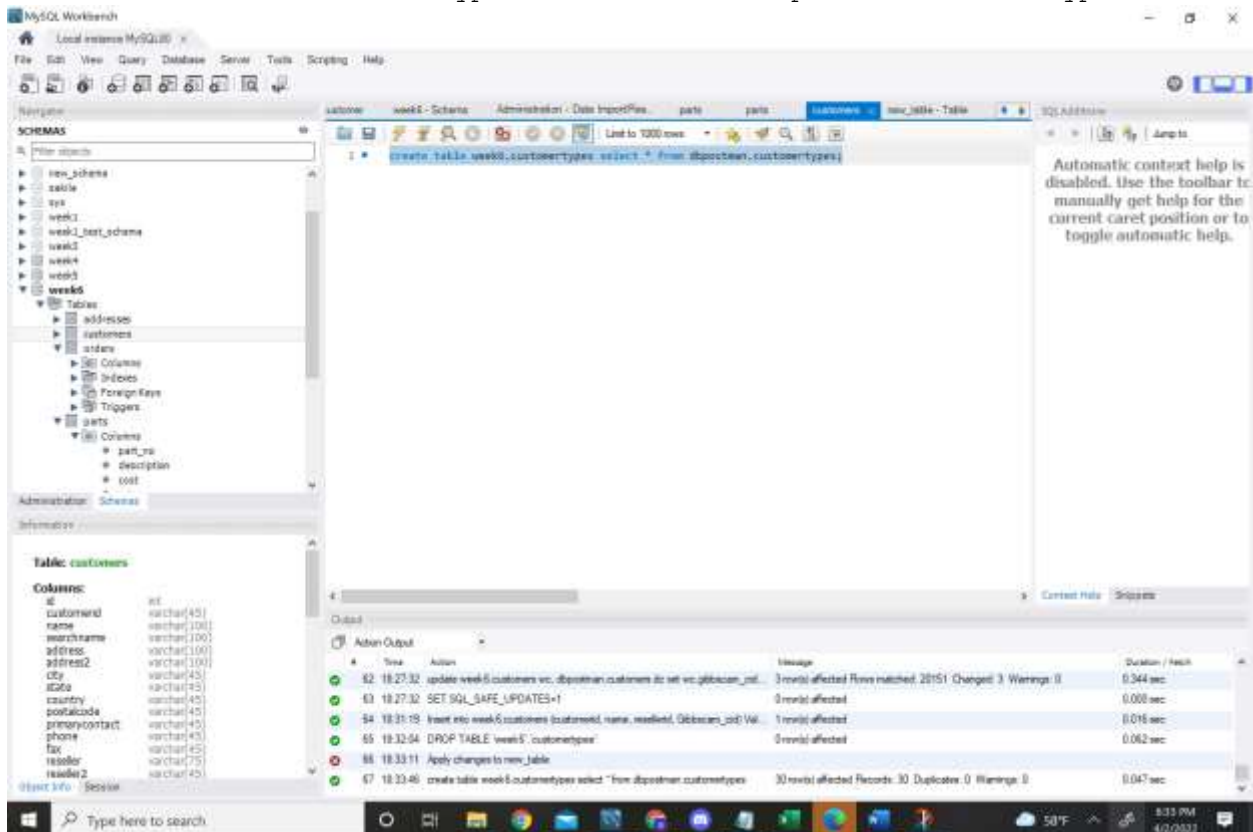
Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a KEY column. To disable safe mode, toggle the option in Preferences -> SQL Editor and reconnect.



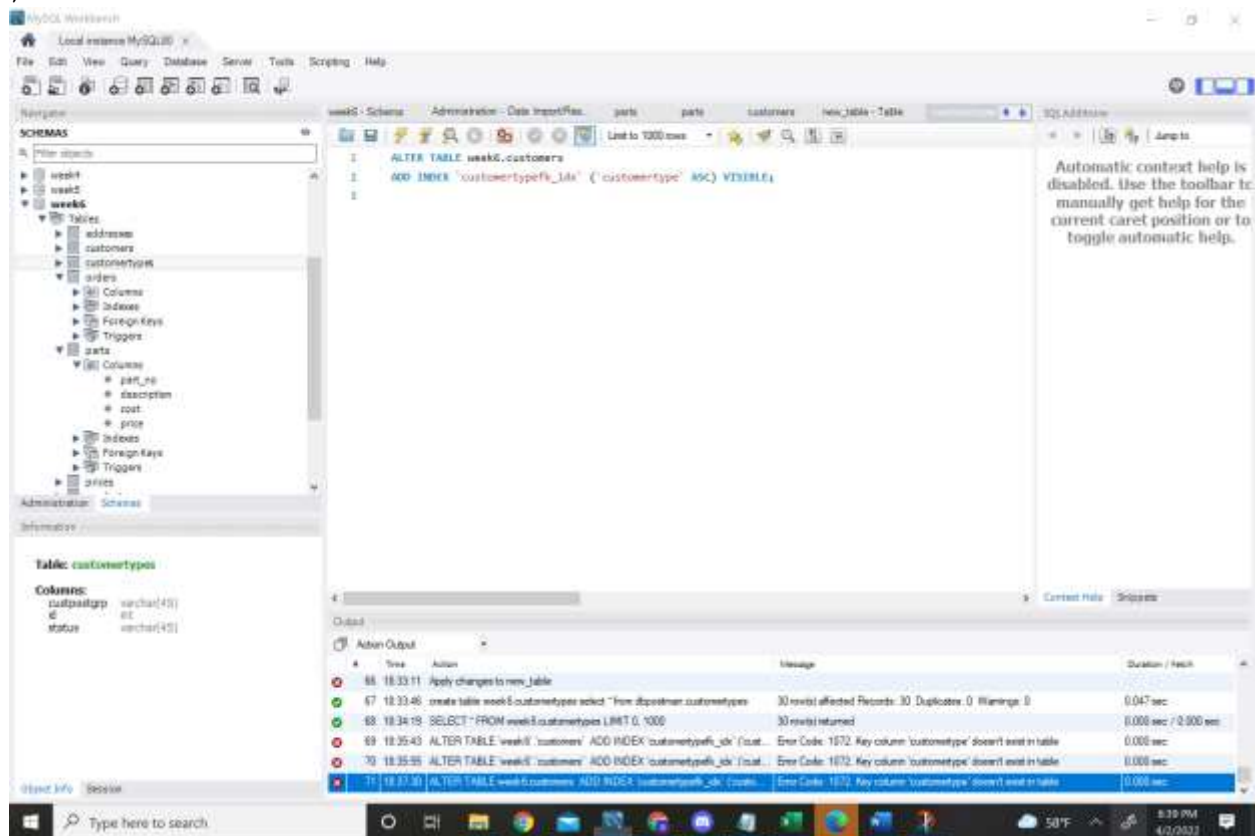
Insert into week6.customers (customerid, name, resellerid, Gibbscam_cid)
Values
('12345', 'test','123', ' A90853090853');
Note: Query had weird character so all text was replaced expect for the
unique Gibbscam_cid



create table week6.customertypes select * from dbpostman.customertypes;



```
ALTER TABLE `week6`.`customers`
ADD INDEX `customertypefk_idx` (`customertype` ASC) VISIBLE;
;
```



```
ALTER TABLE `week6`.`customers`
ADD CONSTRAINT `customertypefk`
FOREIGN KEY (`customertype`)
REFERENCES `week6`.`customertypes` (`id`)
ON DELETE NO ACTION
ON UPDATE NO ACTION;
Didn't work
```

Since customertype doesn't exist in customers
I'm going to run update queries

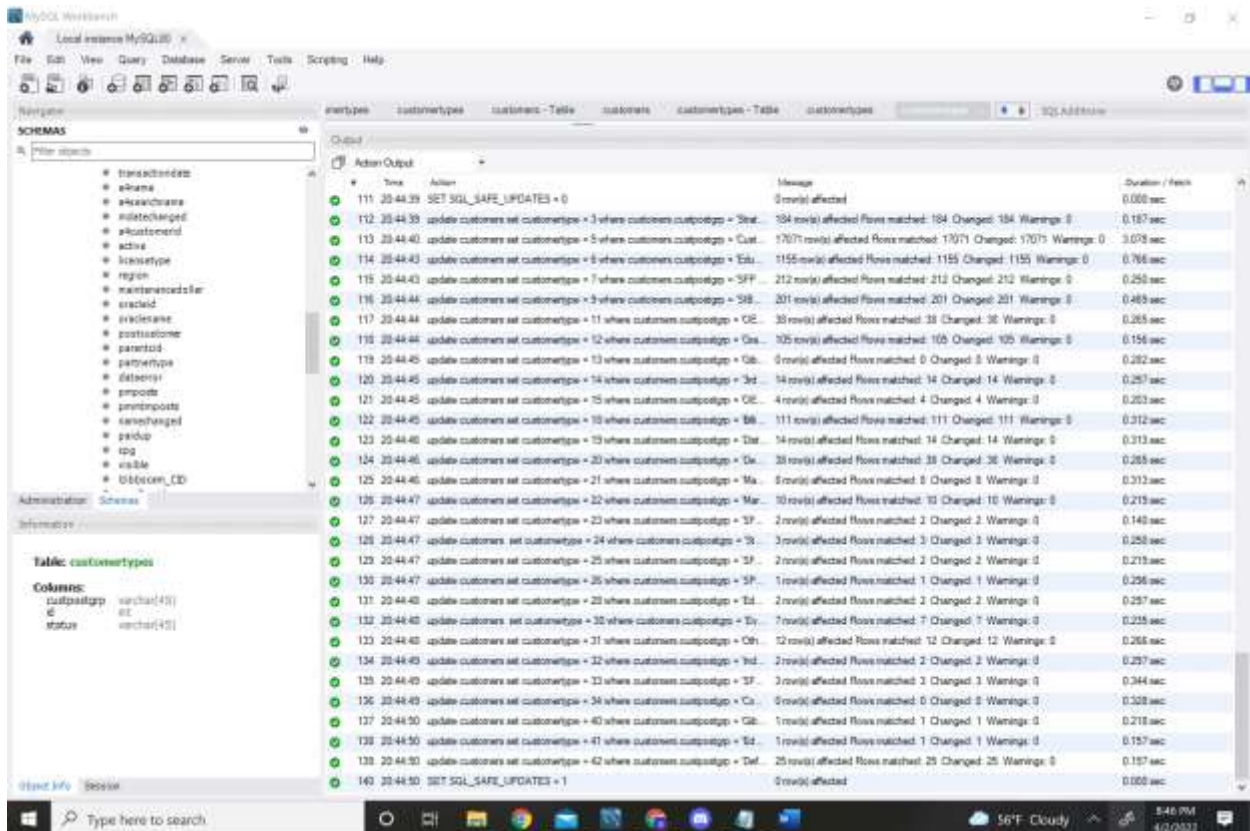
```
SET SQL_SAFE_UPDATES = 0;
update customers
set customertype = 1
where customers.custpostgrp = Reseller;
SET SQL_SAFE_UPDATES = 1;
```

```
SET SQL_SAFE_UPDATES = 0;
update customers
set customertype = 2
where customers.custpostgrp = 'Associate Reseller';
SET SQL_SAFE_UPDATES = 1;
```



```
SET SQL_SAFE_UPDATES = 0;
update customers
set customertype = 3
where customers.custpostgrp = 'Strategic Associate';
update customers set
customertype = 5
where customers.custpostgrp = 'Customer';
update customers
set customertype = 6
where customers.custpostgrp = 'Educational';
update customers
set customertype = 7
where customers.custpostgrp = 'SFP End User';
update customers
set customertype = 9
where customers.custpostgrp = 'SIBNUG';
update customers
set customertype = 11
where customers.custpostgrp = 'OEM/SFP';
update customers
set customertype = 12
where customers.custpostgrp = 'Grandfather';
update customers
set customertype = 13
where customers.custpostgrp = 'Gibbs Employee';
update customers
set customertype = 14
where customers.custpostgrp = '3rd Party Consultant';
update customers
set customertype = 15
where customers.custpostgrp = 'OEM Prospect';
update customers
set customertype = 18
where customers.custpostgrp = 'Billing Company';
update customers
set customertype = 19
where customers.custpostgrp = 'Distribution Manager';
update customers
set customertype = 20
where customers.custpostgrp = 'Development Partner';
update customers
set customertype = 21
where customers.custpostgrp = 'Machine Tool Dealer';
update customers
set customertype = 22
where customers.custpostgrp = 'Marketing Partner';
update customers
set customertype = 23
where customers.custpostgrp = 'SFP Associate';
update customers
set customertype = 24
where customers.custpostgrp = 'Student';
```

```
update customers
set customertype = 25
where customers.custpostgrp = 'SFP Assoc - CNM';
update customers
set customertype = 26
where customers.custpostgrp = 'SPP Partner';
update customers
set customertype = 28
where customers.custpostgrp = 'Educational Reseller';
update customers
set customertype = 30
where customers.custpostgrp = 'Eval - Customer';
update customers
set customertype = 31
where customers.custpostgrp = 'Other';
update customers
set customertype = 32
where customers.custpostgrp = 'Independent Contract';
update customers
set customertype = 33
where customers.custpostgrp = 'SFP Reseller';
update customers
set customertype = 34
where customers.custpostgrp = 'Contract Trainer';
update customers
set customertype = 40
where customers.custpostgrp = 'Gibbs A/E';
update customers
set customertype = 41
where customers.custpostgrp = 'Edu/SFP End User';
update customers
set customertype = 42
where customers.custpostgrp = 'Defunct';
SET SQL_SAFE_UPDATES = 1;
```



I then found 41 rows that could not be assigned a customertype id.

There corresponding custpostgrp were

'Phone Call'
'Client Employee'
'AE'
'CIMNET Reseller'
'SFP Assoc - FAD'
'OMAG Customer'

I added the others as customer type other

```
SET SQL_SAFE_UPDATES = 0;
UPDATE `customers`
SET
`customertype` = '32'
WHERE `custpostgrp` = 'Phone Call';
SET SQL_SAFE_UPDATES = 1;

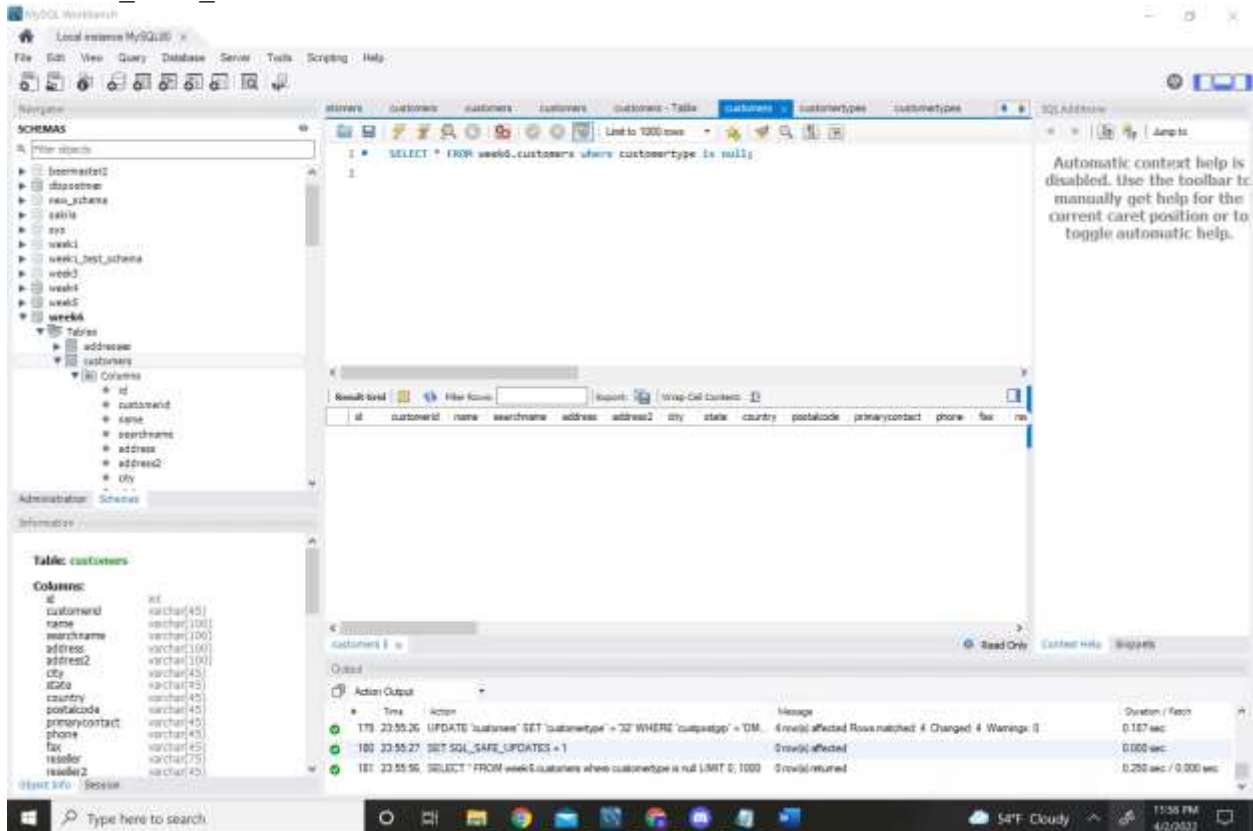
SET SQL_SAFE_UPDATES = 0;
UPDATE `customers`
SET
`customertype` = '32'
WHERE `custpostgrp` = 'Client Employee' ;
SET SQL_SAFE_UPDATES = 1;
SET SQL_SAFE_UPDATES = 0;
UPDATE `customers`
SET
`customertype` = '32'
```

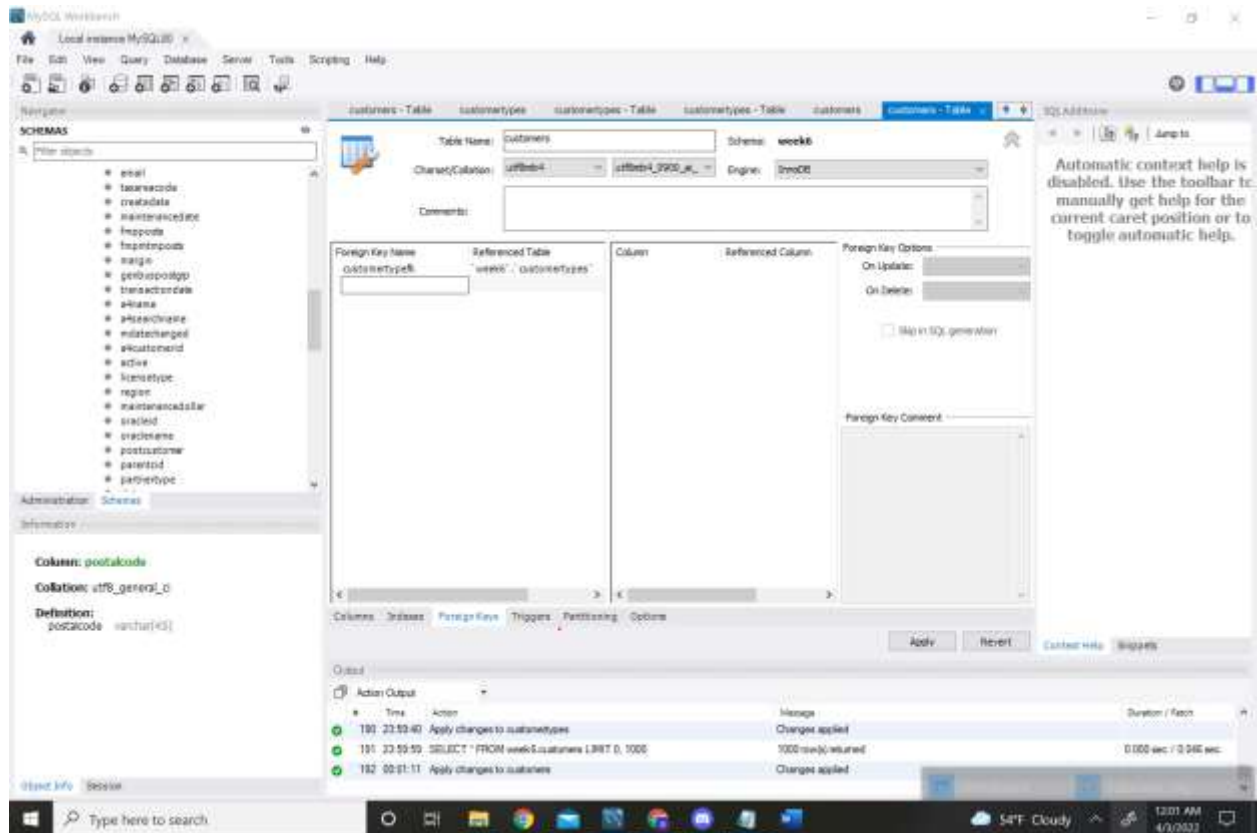
```

WHERE `custpostgrp` = 'AE' ;
SET SQL_SAFE_UPDATES = 1;
SET SQL_SAFE_UPDATES = 0;
UPDATE `customers`
SET
`customertype` = '32'
WHERE `custpostgrp` = 'CIMNET Reseller' ;
SET SQL_SAFE_UPDATES = 1;
SET SQL_SAFE_UPDATES = 0;
UPDATE `customers`
SET
`customertype` = '32'
WHERE `custpostgrp` = 'SFP Assoc - FAD' ;
SET SQL_SAFE_UPDATES = 1;

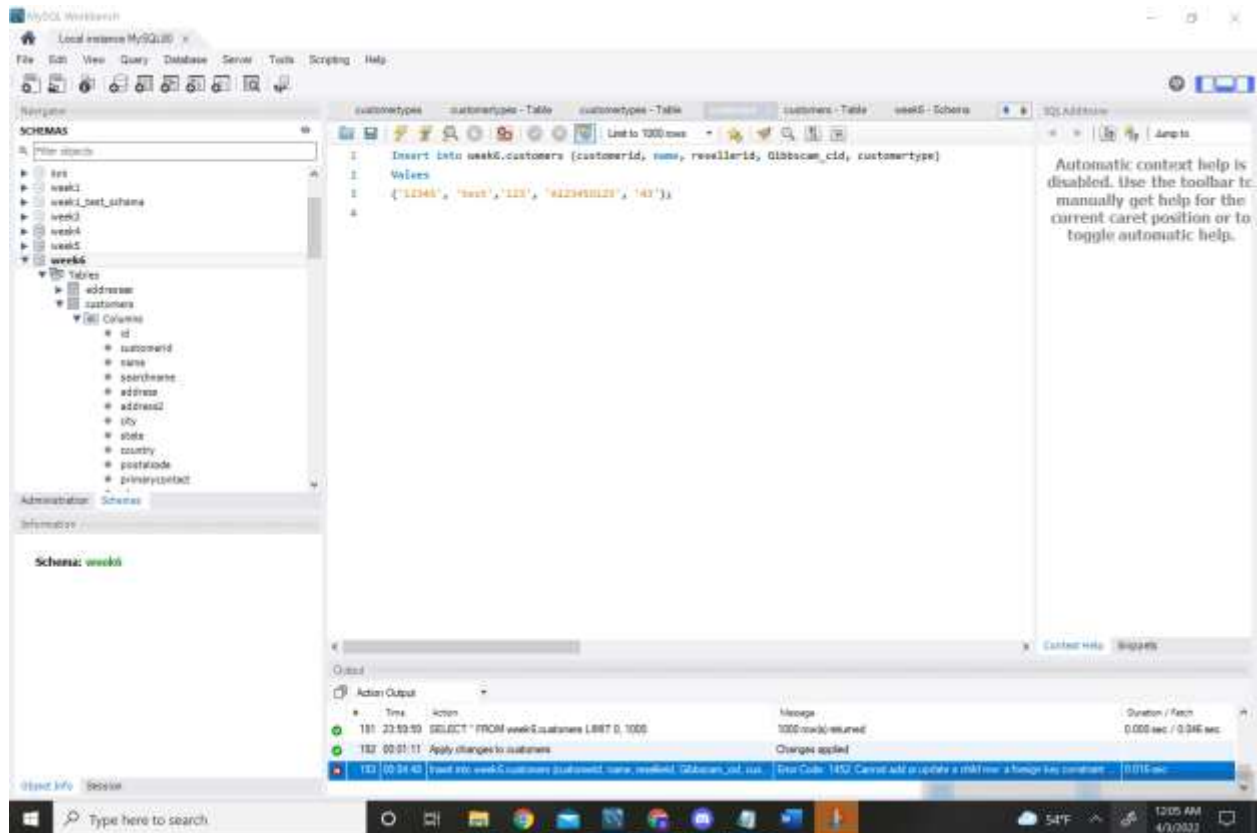
SET SQL_SAFE_UPDATES = 0;
UPDATE `customers`
SET
`customertype` = '32'
WHERE `custpostgrp` = 'OMAG Customer';
SET SQL_SAFE_UPDATES = 1;

```

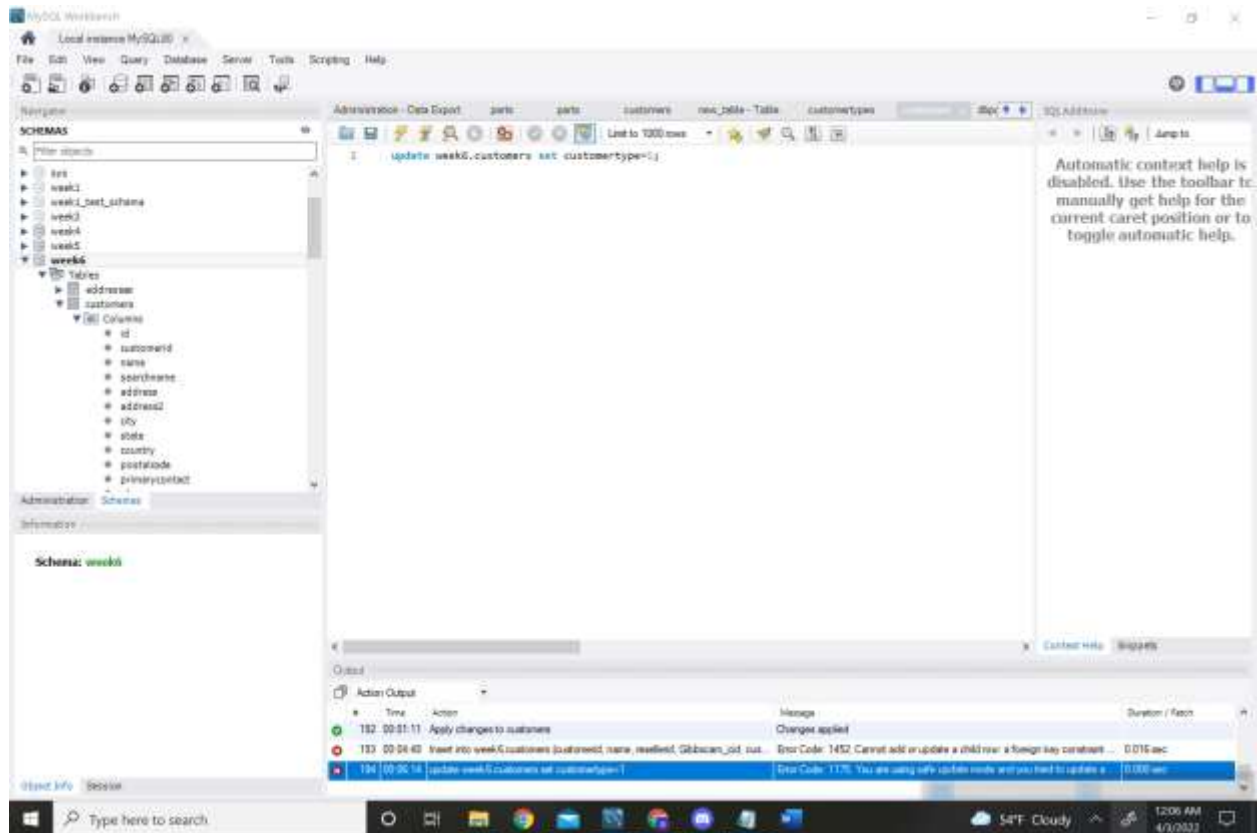




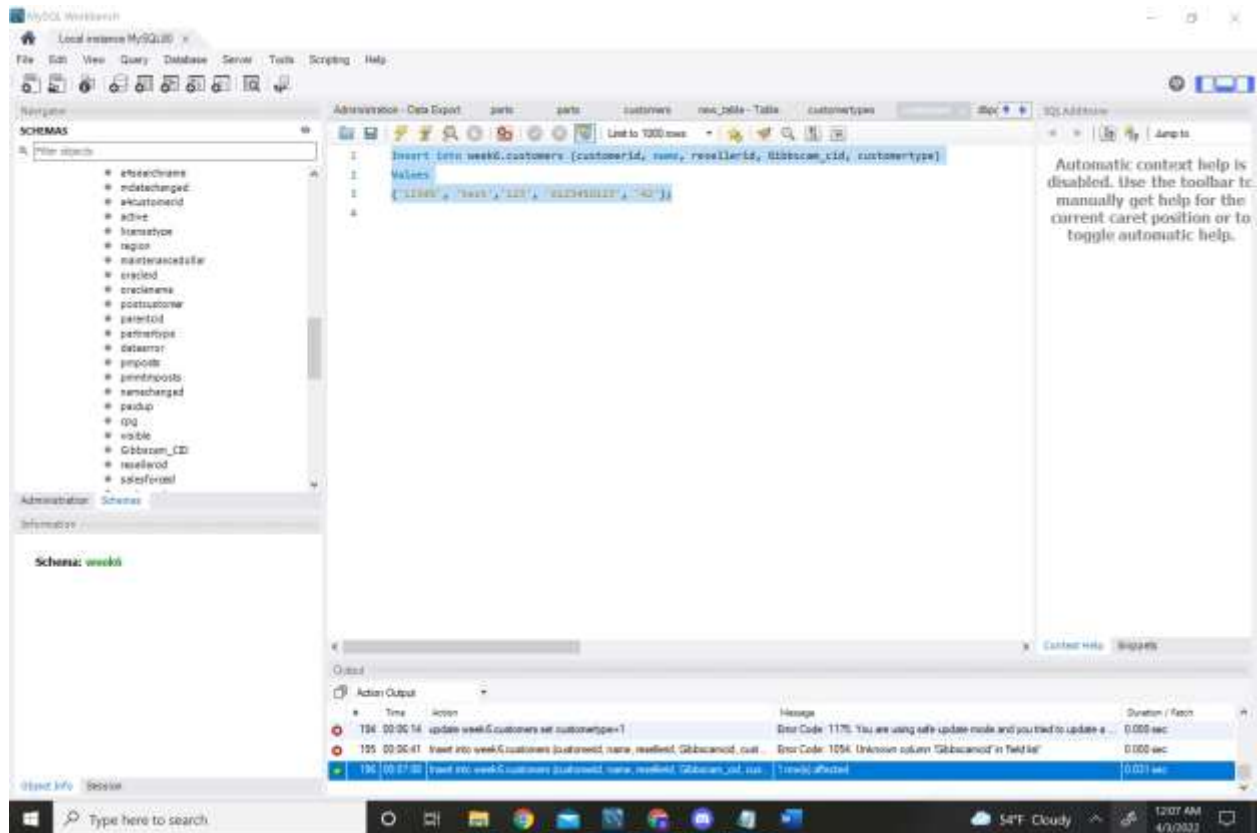
```
Insert into week6.customers (customerid, name, resellerid, Gibbscam_cid,
customertype)
Values
'12345', 'test','123', 'A123450123', '43');
```



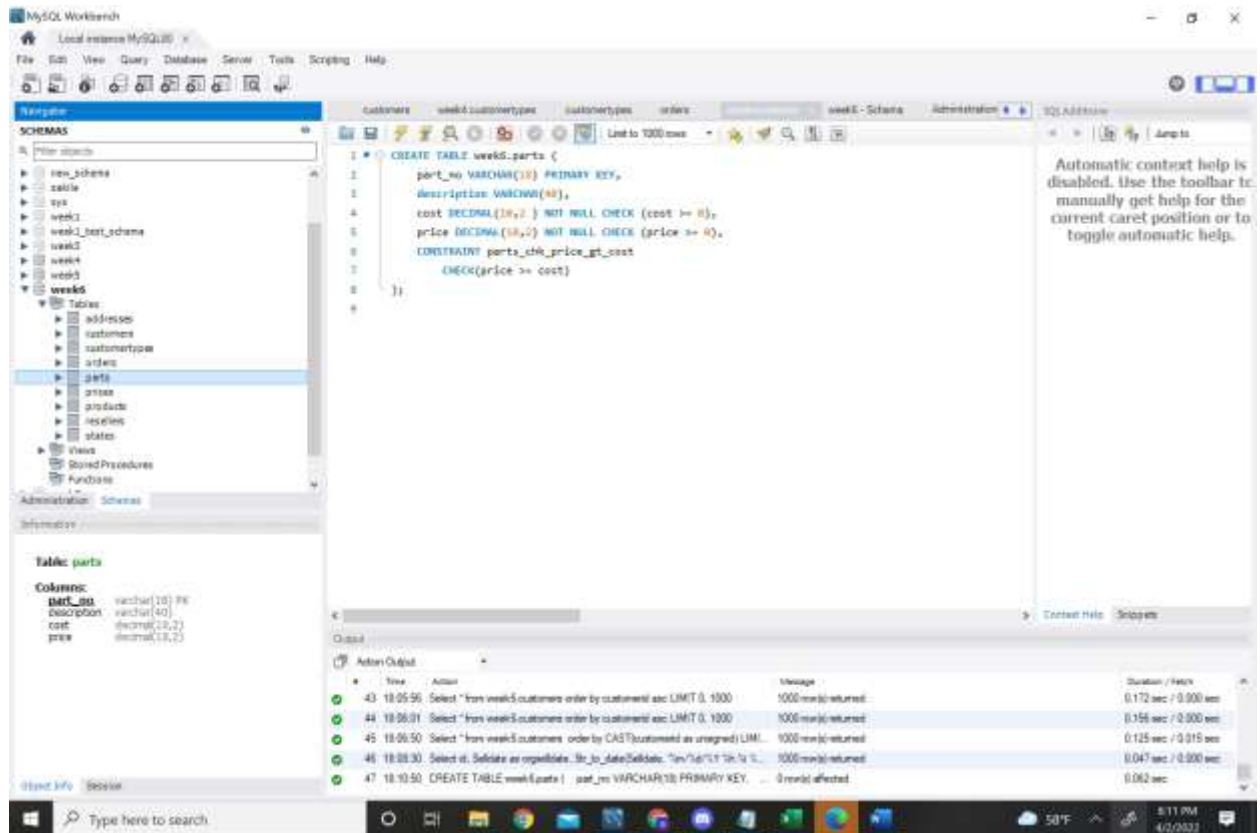
update week6.customers set customertype=1;



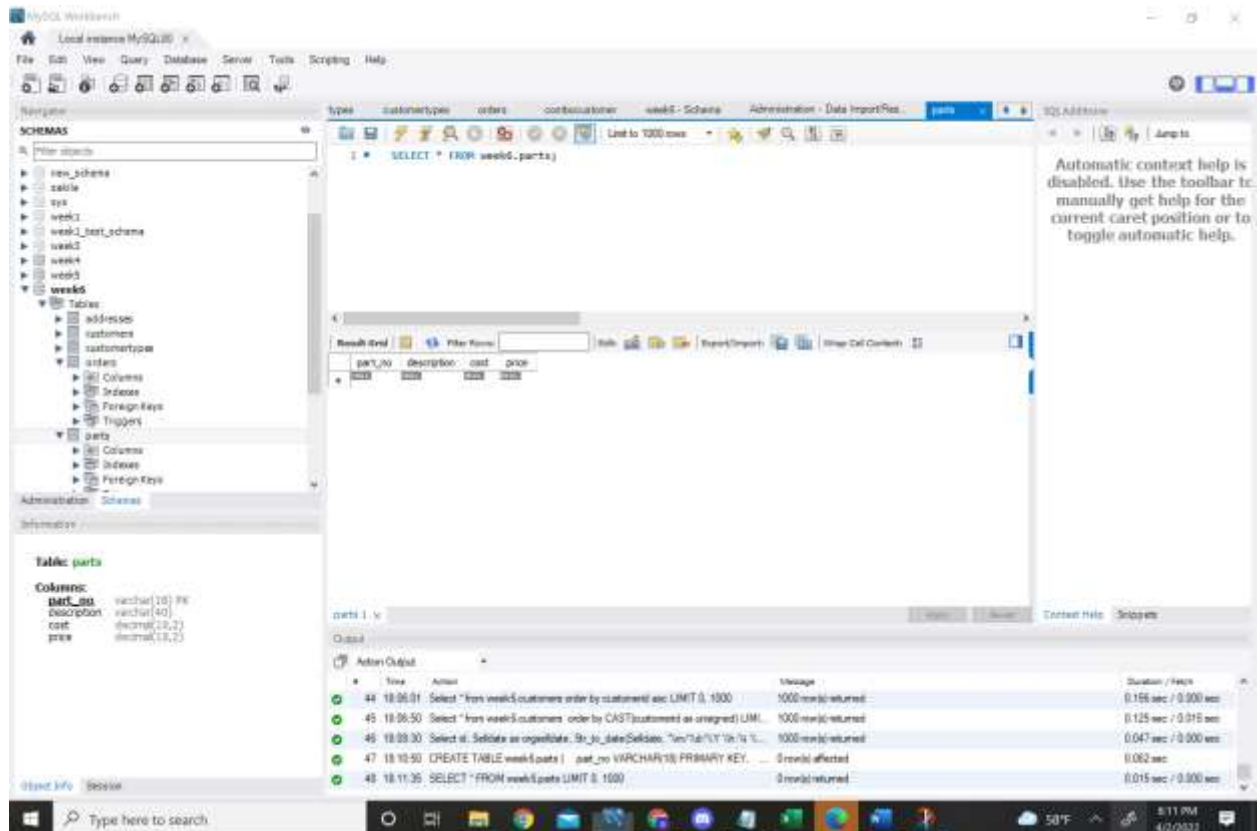
Insert into week6.customers (customerid, name, resellerid, gibbscamcid, customertype)
 Values
 ('12345', 'test', '123', 'A123450123', '42');



```
CREATE TABLE week6.parts (
    part_no VARCHAR(18) PRIMARY KEY,
    description VARCHAR(40),
    cost DECIMAL(10,2 ) NOT NULL CHECK (cost >= 0),
    price DECIMAL(10,2) NOT NULL CHECK (price >= 0),
    CONSTRAINT parts_chk_price_gt_cost
        CHECK(price >= cost)
);
```

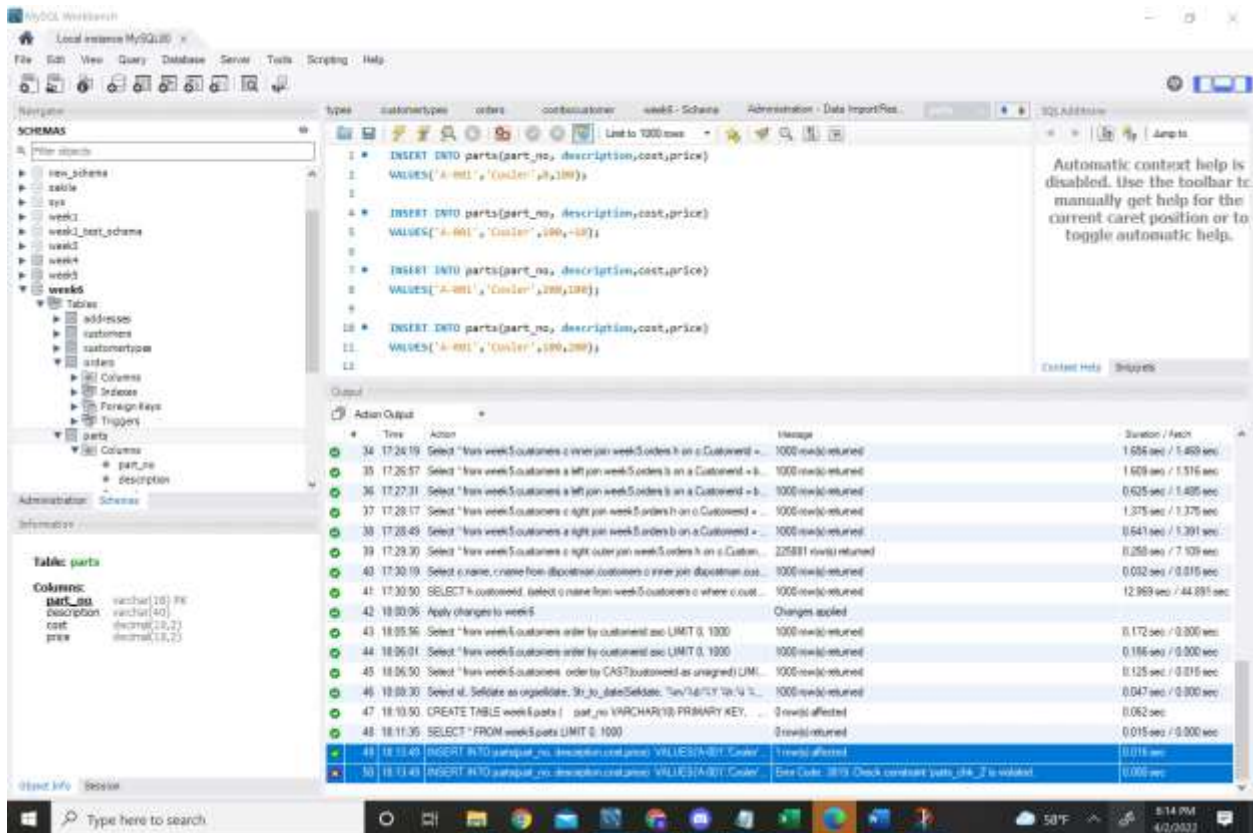


Error Code: 1054. Unknown column 'customertypes.custpostgrp' in 'where clause'



```
INSERT INTO parts(part_no, description, cost, price)
VALUES ('A-001', 'Cooler', 0, 100);
```

```
INSERT INTO parts(part_no, description, cost, price)
VALUES ('A-001', 'Cooler', 100, -10);
```



```

INSERT INTO parts(part_no, description, cost, price)
VALUES('A-001','Cooler',200,100);

```

MySQL Workbench

Local instance MySQL5.7

File Edit View Query Database Server Tools Scripting Help

Schemas

new_schema
sakila
sys
week1
week1_test_schema
week2
week3
week4
week5

Tables

addresses
customers
customer_type
orders
parts

Columns
Indices
Foreign Keys
Triggers

Administration Schemas

Information

Table: parts

Columns:

- part_no varchar(10) PK
- description varchar(40)
- cost decimal(28,2)
- price decimal(18,2)

SQL Editor

```
1 INSERT INTO parts(part_no, description, cost, price)
2 VALUES('A-001', 'Cooler', 100, 200);
```

Output

Time	Action	Message	Duration / Feat
37:17:28.17	Select * from week5 customers c right join week5 orders b on c Customerid =	1000 rows returned	1.375 sec / 1.375 sec
38:17:28.49	Select * from week5 customers c right join week5 orders b on c Customerid =	1000 rows returned	0.641 sec / 1.391 sec
39:17:29.30	Select * from week5 customers c right join week5 orders b on c Customerid =	225881 rows returned	0.250 sec / 7.109 sec
40:17:30.18	Select c.name, r.name from dispatchman customers c inner join dispatchman out	1000 rows returned	0.032 sec / 0.015 sec
41:17:30.90	SELECT h.customerid, select c.name from week5 customers c where c.cust...	1000 rows returned	12.969 sec / 44.881 sec
42:18:00.06	Apply changes to week5	Changes applied	
43:18:05.56	Select * from week5 customers order by customerid asc LIMIT 0, 1000	1000 rows returned	0.172 sec / 0.000 sec
44:18:06.01	Select * from week5 customers order by customerid asc LIMIT 0, 1000	1000 rows returned	0.186 sec / 0.000 sec
45:18:06.50	Select * from week5 customers order by CAST(customerid as unsigned) LIMI	1000 rows returned	0.125 sec / 0.015 sec
46:18:08.30	Select id, Setdate as originaldate, Set_in_dateSetdate: "on" id "LY" to "to "	1000 rows returned	0.047 sec / 0.000 sec
47:18:10.50	CREATE TABLE week6 parts (part_no VARCHAR(10) PRIMARY KEY,	0 rows affected	0.062 sec
48:18:11.35	SELECT * FROM week6 parts LIMIT 0, 1000	0 rows returned	0.015 sec / 0.000 sec
49:18:13.48	INSERT INTO parts(part_no, description, cost, price) VALUES('A-001', 'Cooler'	1 row(s) affected	0.016 sec
50:18:13.49	INSERT INTO parts(part_no, description, cost, price) VALUES('A-001', 'Cooler'	Error Code: 3819. Check constraint 'parts_pk_2' is violated.	0.008 sec
51:18:13.15	SELECT * FROM week6 parts LIMIT 0, 1000	1 row(s) returned	0.008 sec / 0.000 sec
52:18:15.21	INSERT INTO parts(part_no, description, cost, price) VALUES('A-001', 'Cooler'	Error Code: 3819. Check constraint 'parts_pk_2' is violated.	0.008 sec
53:18:15.41	INSERT INTO parts(part_no, description, cost, price) VALUES('A-001', 'Cooler'	Error Code: 3819. Check constraint 'parts_pk_2' is violated.	0.008 sec

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

Output Help Output

8:15 PM 4/2/2012

```
INSERT INTO parts(part_no, description, cost, price)
VALUES('A-001', 'Cooler', 100, 200);
```


MySQL Workbench

Local instance MySQL5

File Edit View Query Database Server Tools Scripting Help

Schemas

- new_schema
- sales
- sys
- week1
- week2_start_schema
- week3
- week4
- week5
 - Tables
 - addresses
 - customers
 - customer_type
 - orders
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - parts
 - Columns
 - part_no
 - description

Administration Schemas

Information

Table: parts

Columns:

- part_no varchar(10) PK
- description varchar(40)
- cost decimal(28,2)
- price decimal(18,2)

SQL Editor

```
1 INSERT INTO parts(part_no, description, cost, price)
2 VALUES('A-001', 'Cooler', 500, 200);
```

Output

Time	Action	Message	Duration / Affect
30 17:25:49	Select * from week5.customers a right join week5.orders b on a.Customer_id =	1000 rows returned	0.641 sec / 1.391 sec
39 17:29:30	Select * from week5.customers a right outer join week5.orders b on a.Custome	225811 rows returned	0.258 sec / 7.109 sec
40 17:30:19	Select a.name, c.name from disposition.customers a inner join disposition.cus	1000 rows returned	0.032 sec / 0.015 sec
41 17:30:50	SELECT h.customer_id, select c.name from week5.customers c where c.cust	1000 rows returned	12.969 sec / 44.891 sec
42 18:00:06	Apply changes to week5	Changes applied	
43 18:05:56	Select * from week5.customers order by customer_id asc LIMIT 0, 1000	1000 rows returned	0.172 sec / 0.000 sec
44 18:06:01	Select * from week5.customers order by customer_id asc LIMIT 0, 1000	1000 rows returned	0.156 sec / 0.000 sec
45 18:06:50	Select * from week5.customers order by CAST(customer_id as unsigned) LIM	1000 rows returned	0.125 sec / 0.015 sec
46 18:08:30	Select id, Setdate as originaldate, Sk_to_date=Setdate, 'mm/dd/yyyy' as %	1000 rows returned	0.047 sec / 0.000 sec
47 18:10:50	CREATE TABLE week5.parts (part_no VARCHAR(10) PRIMARY KEY,	0 rows affected	0.062 sec
48 18:11:35	SELECT * FROM week5.parts LIMIT 0, 1000	0 rows returned	0.015 sec / 0.000 sec
49 18:13:49	INSERT INTO partspart_no, description, cost, price) VALUES('A-001', 'Cooler'	1 rows affected	0.016 sec
50 18:13:49	INSERT INTO partspart_no, description, cost, price) VALUES('A-001', 'Cooler'	Error Code: 3819 Check constraint 'parts_chk_2' is violated.	0.000 sec
51 18:15:15	SELECT * FROM week5.parts LIMIT 0, 1000	1 rows returned	0.000 sec / 0.000 sec
52 18:15:21	INSERT INTO partspart_no, description, cost, price) VALUES('A-001', 'Cooler'	Error Code: 3819 Check constraint 'parts_chk_price_at_cost' is violated.	0.000 sec
53 18:15:41	INSERT INTO partspart_no, description, cost, price) VALUES('A-001', 'Cooler'	Error Code: 3819 Check constraint 'parts_chk_price_at_cost' is violated.	0.000 sec
54 18:16:14	INSERT INTO partspart_no, description, cost, price) VALUES('A-001', 'Cooler'	Error Code: 1062 Duplicate entry 'A-001' for key 'parts.PRIMARY'	0.000 sec

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

5:16 PM 4/2/2012