

DBMS Project Report

PES University

Database Management Systems

UE20CS301

Submitted By

SRN: PES1UG20CS677

Name: Rachana R

SUMMARY:

In this project we developed a complete back end software in which we can update the stock, modify stock, we can forecast the stock, generate invoice. From this application we can get an update that if a particular inventory or stock is less than the some pre-fixed quantity then it'll be easy for the manager/owner to reorder the product from supplier to overcome the "Out of Stock" stage.

In addition to this it can also help us to manage the warehouses, add warehouses which can be proved as very useful feature. We can have complete customer details which can help us to retrieve the order details of regular customers.

From this program we can also keep a track of transactions performed by different customers/clients. We can also get an idea that how much fund we received from different payment methodologies.

This application will keep a high inventory turnover ratio to ensure our products aren't spoiling, becoming obsolete for our working capital. It'll help us to calculate how many times inventory sells in a year and see where we can make better use of our resources

Index

Introduction	2
Data Model	2
FD and Normalization	2
DDL	6
Triggers	3
SQL Queries improvement	14
	Execution plan and performance
	16

Introduction

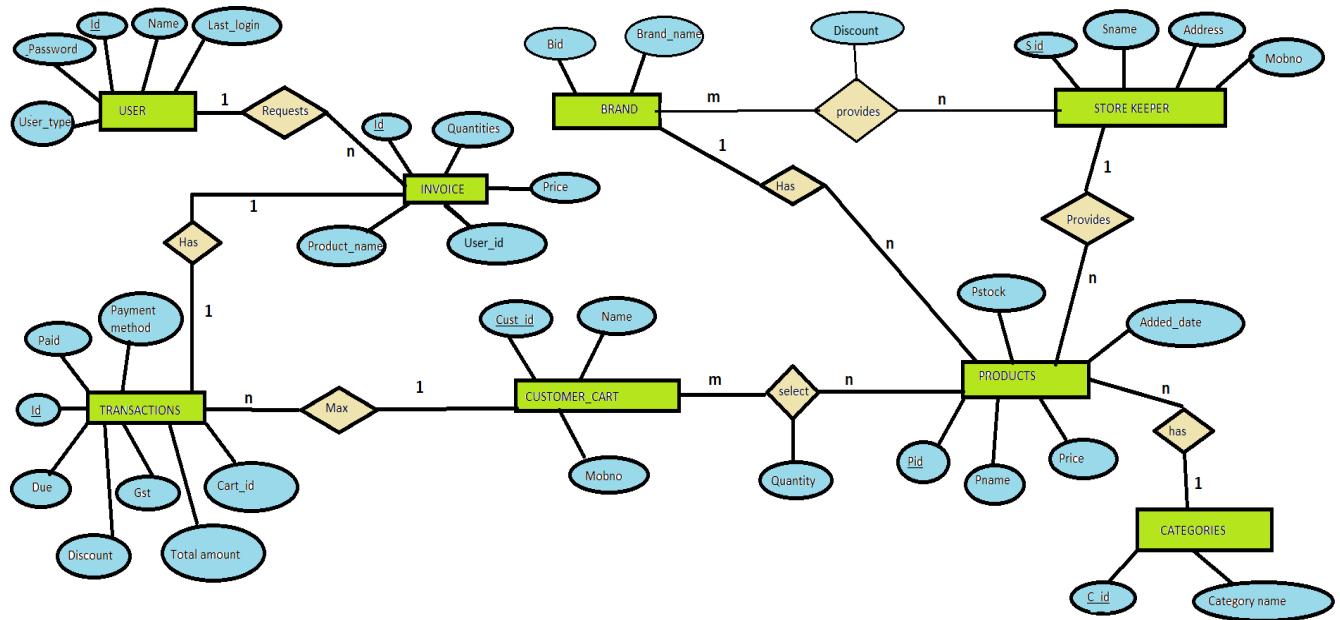
An inventory management system is the combination of technology (hardware and software) and processes and procedures that oversee the monitoring and maintenance of stocked products, whether those products are company assets, raw materials and supplies, or finished products ready to be sent to vendors or end consumers.

This system can widely be used by normal shops, departmental stores or MNCs for keeping a proper track of the stock. It also consists of information like manager details, customer details etc.

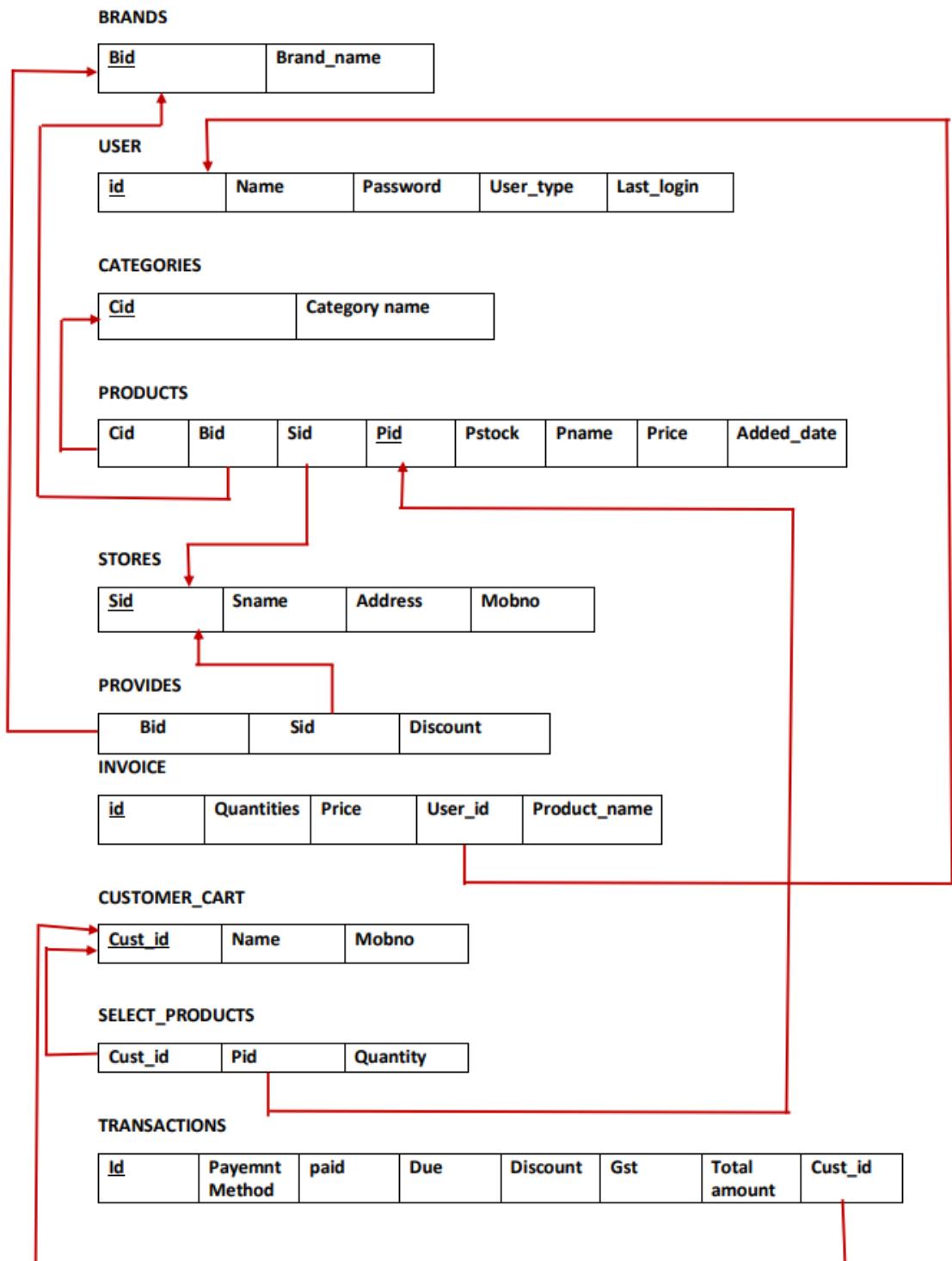
With the help of this system we can fix a minimum quantity of any inventory below which we need to place an order for that inventory. This will help us in good sales results and never the out of stock stage for any inventory.

Data Model

ER DIAGRAM



Relational Schema



FD and Normalization

BRANDS

FD: Bid → (Brand_name)

This relation os in 3NF from

Bid	Brand_name



USER

FD: id → (name,password,user_type,last_login)

This relation os in 3NF from

id	Name	Password	User_type	Last_login



CATEGORIES

FD: Cid → (category name)

This relation os in 3NF from

Cid	Category name



PRODUCT

FD: Pid → (cid,bid,sid,pstock,pname,price,added_date)

This relation os in 3NF from

Cid	Bid	Sid	Pid	Pstock	Pname	Price	Added_date



STORES

FD: Sid → (Sname, Address, Mobno)

This relation os in 3NF from

Sid	Sname	Address	Mobno



INVOICE

FD: id → (quantities, price, user_id, product_name)

This relation is in 3NF form

<u>id</u>	Quantities	Price	User_id	Product_name



CUSTOMER_CART

FD: Cust_id → (Name, Mobno)

This relation is in 3NF form

<u>Cust_id</u>	Name	Mobno

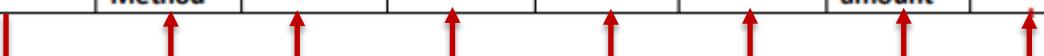


TRANSACTIONS

FD: Id → (payment method, paid, due, discount, Gst, total amount, cust_id)

This relation is in 3NF form

<u>Id</u>	Payemnt Method	paid	Due	Discount	Gst	Total amount	Cust_id



DDL

Creating tables

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons for database management.
- Navigator:** Shows the schema structure with tables: brands, categories, customer_cart, inv_user, invoice, product, provides, select_product, stores.
- Central Area:** SQL Editor containing the DDL code for creating tables. The code is color-coded for syntax highlighting.
- Right Panel:** Shows three identical messages: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."
- Status Bar:** Local instance MySQL80, various tabs like project, database_design, quiz_sql_file, and SQL.Additions.

```
1 • use project;
2 • create table brands(
3     bid varchar(5),
4     bname varchar(20),
5     primary key (bid)
6 );
7
8 • create table inv_user(
9     user_id varchar(20),
10    name varchar(20),
11    password varchar(20),
12    last_login timestamp,
13    user_type varchar(10) ,
14    primary key (user_id);
15
16 • create table categories(
17     cid varchar(5),
18     category_name varchar(20),
19     primary key (cid));
21
22
23 • create table stores(
24     sid varchar(5),
25     sname varchar(20),
26     address varchar(20),
27     mobno varchar(10),
28     primary key(sid));
29
30
31 • create table product(
32     pid varchar(5),
33     cid varchar(5) references categories(cid) ,
34     bid varchar(5) references brands(bid) ,
35     sid varchar(5) references stores(sid),
36     pname varchar(20),
37     p_stock varchar(5),
38     added_date date,
39     price varchar(5),
40     primary key (pid));
41
42 • create table provides(
43     bid varchar(5)references brands(bid),
44     sid varchar(5)references stores(sid),
45     discount varchar(5));
46
47 • create table customer_cart(
48     cust_id varchar(5) primary key,
49     name varchar(20),
50     mobno varchar(10) );
51
52 • create table select_product(
53     cust_id varchar(5) references customer_cart(cust_id),
54     pid varchar(5)references product(pid),
55     quantity varchar(4) );
56
```

```

57 • create table transactions(
58     id varchar(5) primary key,
59     total_amount bigint,
60     paid bigint,
61     due bigint,
62     gst varchar(10),
63     discount varchar(5),
64     payment_method varchar(50),
65     cust_id varchar(5) references customer_cart(cust_id) );
66
67 • create table invoice(
68     item_no varchar(5),
69     product_name varchar(20),
70     quantity varchar(5),
71     net_price varchar(5),
72     transaction_id varchar(5)references transaction(id) );
73
74
75
76

```

No object selected

Object Info Session Output

Type here to search

Windows Taskbar: Type here to search, File Explorer, Task View, Start, Taskbar icons, Desktop, Rachana R, 24°C, 18-11-22, 12:46

Inserting values:

```

76
77 /*insertion*/
78 • insert into brands(bid,bname)
79     values(1,'apple'),(2,'Samsung'),(3,'Nike'),(4,'Fortune'),(5,'Adiddas'),(6,'PUMA'),
80     (7,'oneplus'),(8,'figaro'),(9,'MTR'),(10,'GRB'),
81     (11,'Levis'),(12,'only');
82 • select * from brands;

```

bid	bname
1	apple
2	GRB
11	Levis
12	only
2	Samsung
3	Nike
4	Fortune
5	Adiddas
6	PUMA
7	oneplus
8	figaro
9	MTR
10	GRB

Result Grid | Filter Rows: [] | Edit: [] | Export/Import: [] | Wrap Cell Content: []

Action Output

Time	Action	Message	Duration / Fetch
80 19:11:07	insert into brands(bid,bname) values(1,apple),(2,Samsung),(3,Nike),(4,Fortune),(5,Adiddas),(6,PUMA),(7,oneplus),(8,figaro),(9,MTR),(10,GRB)	12 row(s) affected Records: 12 Duplicates: 0 Warnings: 0	0.000 sec
81 19:11:07	select * from brands LIMIT 0, 1000	12 row(s) returned	0.000 sec / 0.000 sec

Type here to search

Windows Taskbar: Type here to search, File Explorer, Task View, Start, Taskbar icons, Desktop, Rachana R, 23°C, 18-11-22, 19:12, 24

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- Tables
 - brands
 - categories
 - customer_cat
 - inv_user
 - invoice
 - product
 - provides
 - select_product
 - stores
- Views
- Stored Procedures
- Functions

Administration Schemas

Information

Multiple selected objects

```

86 /*INSERT INTO INV_USER*/
87 • insert into inv_user (user_id,name,password,last_login,user_type)
88 values('prashant@gmail.com','Prashant','0011tr','2022-10-29 10:20','Accountant'),
89 ('harsh@gmail.com','Harsh Khanelwal','1111re','2022-10-30 10:20','Manager'),
90 ('vikas66@gmail.com','vikas p','6626oiu','2022-10-17 11:20','Manager'),
91 ('ram872@gmail.com','Ram sharma','6574trh','2022-10-21 10:20','Manager');
92 • select * from inv_user;
93

```

Result Grid | Filter Rows: [] Edit: [] Export/Import: [] Wrap Cell Content: []

user_id	name	password	last_login	user_type
harsh@gmail.com	Harsh Khanelwal	1111re	2022-10-30 10:20:00	Manager
prashant@gmail.com	Prashant	0011tr	2022-10-29 10:20:00	Accountant
ram872@gmail.com	Ram sharma	6574trh	2022-10-21 10:20:00	Manager
vikas66@gmail.com	vikas p	6626oiu	2022-10-17 11:20:00	Manager
NULL	NULL	NULL	NULL	NULL

inv_user 25 x

Output

#	Time	Action	Message	Duration / Fetch
82	19:13:17	insert into inv_user (user_id,name,password,last_login,user_type) values('prashant@gmail.com','Prashant','0011tr','2022-10-29 10:20','Accountant')	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.016 sec
83	19:13:17	select * from inv_user LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- Tables
 - brands
 - categories
 - customer_cat
 - inv_user
 - invoice
 - product
 - provides
 - select_product
 - stores
- Views
- Stored Procedures
- Functions

Administration Schemas

Information

Multiple selected objects

```

95
96 /*insertion into categories*/
97 • insert into categories(cid,category_name)
98 values(1,'gadgets'),(2,'Clothing'),(3,'Grocery'),(4,'shoes');
99 • select * from categories;
100
101
102

```

Result Grid | Filter Rows: [] Edit: [] Export/Import: [] Wrap Cell Content: []

cid	category_name
1	gadgets
2	Clothing
3	Grocery
4	shoes
NULL	NULL

categories 26 x

Output

#	Time	Action	Message	Duration / Fetch
84	19:14:26	insert into categories(cid,category_name) values(1,'gadgets'),(2,'Clothing'),(3,'Grocery'),(4,'shoes')	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
85	19:14:26	select * from categories LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigators: pes1ug20cs677 cs677_wee8_1* cs677_week8_2* week9 SQL File 10* train SQL File 12* SQL File 13* project* database_design quiz_sql_file

SCHEMAS: Filter objects Tables brands categories customer_cart inv_user invoice product provider select_product stores Views Stored Procedures Functions

```

104 • insert into stores (sid,sname,address,mobno)
105   values(1,'RR super mart','chennai',8888555541),
106   (2,'SV Super mart','Haryana',7777555541),
107   (3,'Raj Super mart','Bengaluru',9876543620),
108   (4,'vishnu Super mart','Hyderabad',7684532190),
109   (5,'CC super mart','Hubli',6785436578),
110   (6,'Iyer super mart','Managaluru',9345654356);
111 • select * from stores;

```

Result Grid: sid sname address mobno

1	RR super mart	chennai	8888555541
2	SV Super mart	Haryana	7777555541
3	Raj Super mart	Bengaluru	9876543620
4	vishnu Super mart	Hyderabad	7684532190
5	CC super mart	Hubli	6785436578
6	Iyer super mart	Managaluru	9345654356

stores 27 x

Action Output:

- # Time Action Message Duration / Fetch
 - 86 19:15:00 insert into stores (sid,sname,address,mobno) values(1,'RR super mart','chennai',8888555541), (2,'SV Super mart','Haryana',7777555541); 6 row(s) affected Records: 6 Duplicates: 0 Warnings: 0 0.000 sec / 0.000 sec
 - 87 19:15:00 select * from stores LIMIT 0,1000 6 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigators: pes1ug20cs677 cs677_wee8_1* cs677_week8_2* week9 SQL File 10* train SQL File 12* SQL File 13* project* database_design quiz_sql_file

SCHEMAS: Filter objects Tables brands categories customer_cart inv_user invoice product provider select_product stores Views Stored Procedures Functions

```

114 • insert into product(pid,cid,bid,sid,pname,p_stock,added_date,price)
115   values(1,1,1,'iphone',4,'2018-10-30',65000),(2,1,1,1,'Airpods',3,'2018-10-27',120
116   (3,1,1,1,'Smart Watch',3,'2018-10-27',25000),
117   (4,4,3,2,'Air Max',6,'2018-10-20',35000),(5,3,4,3,'REFINED OIL',6,'2018-10-25',1200
118   (6,3,3,3,'sambar powder',100,'2022-10-21',10000),
119   (7,3,10,3,'gillab jamoon mix',50,'2022-09-1',5000),(8,4,5,6,'adiddas_sneakers',20,'2
120   (9,2,11,1,'Regular fit jeans',20,'2022-10-22',50000),(10,2,12,1,'t shirts',30,'2022
121 • select * from product;
122

```

Result Grid: pid cid bid sid pname p_stock added_date price

1	1	1	1	iphone	4	2018-10-30	65000
10	2	12	1	t shirts	30	2022-10-21	30000
2	1	1	1	Airpods	3	2018-10-20	12000
3	1	1	1	Smart Watch	3	2018-10-27	25000
4	4	3	2	Air Max	6	2018-10-20	35000
5	3	4	3	REFINED OIL	6	2018-10-25	1200
6	3	9	3	sambar powder	100	2022-10-21	10000
7	3	10	3	gillab jamoon mix	50	2022-09-1	5000
8	4	5	6	adiddas_sneakers	20	2022-11-01	40000
9	2	11	1	Regular fit jeans	20	2022-10-22	50000

product 28 x

Action Output:

- # Time Action Message Duration / Fetch
 - 88 19:15:42 insert into product(pid,cid,bid,sid,pname,p_stock,added_date,price) values(1,1,1,'iphone',4,'2018-10-30',65000); 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0 0.000 sec / 0.000 sec
 - 89 19:15:42 select * from product LIMIT 0,1000 10 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

```

123 /*insertion into provides*/
124 • insert into provides(bid,sid,discount)
125   values(1,1,'10%'),(2,2,'10%'),(3,1,'20%'),(4,3,'5%'),(5,4,'15%'),(6,6,'10%'),(7,2,
126   '5%');
127 • select * from provides;
128
129

```

Result Grid | Filter Rows: Export: Wrap Cell Content:

bid	sid	discount
1	1	10%
2	2	10%
3	1	20%
4	3	5%
5	4	15%
6	6	10%
7	2	10%
8	3	5%
9	5	10%
7	2	5%

Provides 29 x

Output

#	Time	Action	Message	Duration / Fetch
90	19:17:17	insert into provides(bid,sid,discount) values(1,1,'10%'),(2,2,'10%'),(3,1,'20%'),(4,3,'5%'),(5,4,'15%'),(6,6,'10%'),(7,2,'5%');	7 rows(s) affected Records: 10 Duplicates: 0 Warnings: 0	0.000 sec
91	19:17:17	select * from provides LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

```

129
130
131 /*insertion into customer_cart*/
132 • insert into customer_cart (cust_id,name,mobno)
133   values(1,'Ram',7896547534),(2,'Shyam',7777777777),(3,'Mohan',7777777775),(4,'Mohan',
134   (6,'veena',8764534567),(7,'sneha',9875232323);
135 • select * from customer_cart;
136

```

Result Grid | Filter Rows: Export/Import: Wrap Cell Content:

cust_id	name	mobno
1	Ram	7896547534
2	Shyam	7777777777
3	Mohan	7777777775
4	Mohan	7567777775
5	Raghav	9756457689
6	veena	8764534567
7	sneha	9875232323
8	MOHAN	9875232323

Customer_cart 30 x

Output

#	Time	Action	Message	Duration / Fetch
92	19:17:56	insert into customer_cart (cust_id,name,mobno) values(1,'Ram',7896547534),(2,'Shyam',7777777777),(3,'Mohan',7777777775),(4,'Mohan',7567777775),(6,'veena',8764534567),(7,'sneha',9875232323);	7 rows(s) affected Records: 7 Duplicates: 0 Warnings: 0	0.015 sec
93	19:17:56	select * from customer_cart LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Table: product

Columns:

- pid
- cd
- sd
- prname
- p_stock
- abuted_date
- price

```

154
155
156 • insert into select_product (cust_id,pid,quantity)
157   values(1,3,1),(2,3,3),(3,2,1),(2,6,2),(1,7,3),(7,5,3),(6,1,4),(5,9,3),
158   (2,3,1),(2,4,3),(3,1,3),(2,6,3),(7,7,3),(7,4,3),(2,1,3),(5,2,3);
159 • select * from select_product;
160

```

Result Grid

cust_id	pid	quantity
1	3	1
2	3	3
3	2	1
2	6	2
1	7	3
7	5	3
6	1	4
5	9	3
2	3	1
2	4	3
3	1	3
2	6	3
7	7	3
7	4	3
2	1	3
5	2	3

select_product 74 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
196	23:47:26	EXPLAIN FORMAT=JSON SELECT product.* FROM product JOIN select_product ON prod...	OK	0.000 sec
197	00:08:04	select * from select_product LIMIT 0, 1000	16 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Multiple selected objects

```

142
143
144 • insert into transactions (id,total_amount,paid,due,gst,discount,payment_method,cus
145   values(1,57000,57000,0,570,'5%','cash',1),(2,5700,5700,0,570,0,'UPI',2),
146   (3,19000,17000,2000,190,'5%','cash',3),(4,5700,5700,0,570,0,'UPI',4),
147   (5,145000,100000,40500,4500,10%,debit/credit card',6),(6,2000,1800,0,200,'10%','
148 • select * from transactions;
149 /*queries */

```

Result Grid

id	total_amount	paid	due	gst	discount	payment_method	cus
1	57000	57000	0	570	5%	cash	
2	5700	5700	0	570	0	UPI	2
3	19000	17000	2000	190	5%	cash	3
4	5700	5700	0	570	0	UPI	4
5	145000	100000	40500	4500	10%	debit/credit card	6
6	2000	1800	0	200	10%	credit/debit card	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

transactions 32 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
96	19:19:27	insert into transactions (id,total_amount,paid,due,gst,discount,payment_method,cus_id) values(1,57000,5700,0,570,'5%','cash',1)	Error Code: 1062: Duplicate entry '1' for key transactions.PRIMARY'	0.000 sec
97	19:19:53	select * from transactions LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Triggers

Update trigger on product table:

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** Local instance MySQL80
- Query Editor:** Contains the SQL code for creating an update trigger.

```
179 /*update trigger*/
180 • UPDATE product
181     SET p_stock = 5
182     WHERE pid = 1;
183 • drop trigger product_UPDATE;
184 delimiter //
185 • CREATE TRIGGER product_UPDATE before UPDATE on product
186     for each row
187     begin
188         insert into product_log values(new.pid,new.pname,new.p_stock,now());
189     end;;
190 delimiter ;
```
- Result Grid:** Shows the data from the product_log table after the trigger was tested.

pid	pname	p_stock	modified_time
1	iphone	10	2022-11-18 11:05:18
1	iphone	5	2022-11-18 18:44:38
- Output:** Shows the execution log of the trigger.

#	Time	Action	Message	Duration / Fetch
69	18:44:38	UPDATE product SET p_stock = 5 WHERE pid = 1	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
70	18:44:43	select * from product_log LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

Insert trigger on product table:

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** Local instance MySQL80
- Query Editor:** Contains the SQL code for creating an insert trigger.

```
189 /*INSERT trigger*/
190 • insert into product(pid,cid,bid,sid,pname,p_stock,added_date,price)
191     values(12,2,11,4,'skin fit jeans',5,'2018-10-30',12500);
192 • drop trigger product_INSERT;
193 delimiter //
194 • CREATE TRIGGER product_INSERT before insert on product
195     for each row
196     begin
197         insert into product_log values(new.pid,new.pname,new.p_stock,now());
198     end;;
199 delimiter ;
```
- Result Grid:** Shows the data from the product_log table after the trigger was tested.

pid	pname	p_stock	modified_time
12	skin fit jeans	5	2022-11-18 19:43:51
- Output:** Shows the execution log of the trigger.

#	Time	Action	Message	Duration / Fetch
130	19:43:51	insert into product(pid,cid,bid,sid,pname,p_stock,added_date,price) values(12,2,11,4,'skin fit jeans',5,'2018-10-30',12500)	1 row(s) affected	0.000 sec
131	19:43:56	select * from product_log LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

SQL Queries

1) Nested query

- Display all the product names from products under a particular category

The screenshot shows the MySQL Workbench interface with a SQL editor window. The code is:

```
151 /*queries */
152 /*display all the product names from products under a particular category*/
153 • select pname
   from product
154   where cid in (select cid
155     from product
156     where cid=1);
157
158
159
```

The result grid shows the following data:

pname
phone
Airpods
Smart Watch

The output pane shows the execution log:

Action	Time	Message	Duration / Fetch
select * from transactions LIMIT 0, 1000	97 19:19:53	6 row(s) returned	0.000 sec / 0.000 sec
select pname from product where cid in (select cid from product where cid=1) LIMIT 0, 1000	98 19:23:21	3 row(s) returned	0.000 sec / 0.000 sec

2) aggregate queries

- count the number of products who's stock is >3

The screenshot shows the MySQL Workbench interface with a SQL editor window. The code is:

```
159
160 /*aggregate queries*/
161 /*count the number of products who's stock is >3*/
162 • select count(pid)
   from product
163   where p_stock>3;
164
165
166
167
```

The result grid shows the following data:

count(pid)
8

The output pane shows the execution log:

Action	Time	Message	Duration / Fetch
select * from product LIMIT 0, 1000	100 19:26:23	10 row(s) returned	0.000 sec / 0.000 sec
select count(pid) from product where p_stock>3 LIMIT 0, 1000	101 19:27:17	1 row(s) returned	0.000 sec / 0.000 sec

- Give the count of transaction_id's who's due > 0 and group the same by payment method.

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Schemas:** Local instance MySQL80, showing tables like brands, categories, customer_cart, inv_user, invoice, product, product_log, providers, select_product, stores, transactions.
- SQL Editor:** Contains the following SQL code:


```

166 • select count(id),payment_method
167   from transactions
168   group by payment_method
169   having sum(due) >0;
170
      
```
- Result Grid:** Shows the output of the query:

count(id)	payment_method
2	cash
1	debit/credit card
- Output Panel:** Displays the execution log with two entries:
 - # 117 19:32:51 select * from transactions LIMIT 0, 1000 Message 6 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec
 - # 118 19:33:08 select count(id),payment_method from transactions group by payment_method having sum(due) >0 LIMIT 0, 10... 2 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec
- System Bar:** Shows the taskbar with various icons and the system tray indicating the date and time as 18-11-22.

3) Right outer join on stores table using pid(product_id)

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Schemas:** Local instance MySQL80, showing tables like customer_cart, inv_user, invoice, product, product_log, providers, select_product, stores, transactions, quiz.
- SQL Editor:** Contains the following SQL code:


```

189
190
191 /*right outer join*/
192 • SELECT product.pid,stores.sname
193   FROM product
194   RIGHT JOIN stores ON product.sid=stores.sid
195   ORDER BY product.pid;
196
      
```
- Result Grid:** Shows the output of the query:

pid	sname
1	Vera Super Mart
2	Vishal Super Mart
3	Vidya Super Mart
4	RR Super Mart
5	SV Super Mart
6	Raj Super Mart
7	Raj Super Mart
8	Iyer Super Mart
9	RR Super Mart
- Output Panel:** Displays the execution log with two entries:
 - # 192 23:45:02 select * from product LIMIT 0, 1000 Message 9 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec
 - # 193 23:45:20 SELECT product.pid,stores.sname FROM product RIGHT JOIN stores ON product.sid=stores.sid ORDER BY ... 29 row(s) returned Duration / Fetch 0.000 sec / 0.000 sec
- System Bar:** Shows the taskbar with various icons and the system tray indicating the date and time as 18-11-22.

Execution Plan before and after

BEFORE:

The screenshot shows the MySQL Workbench interface with the SQL editor tab active. The code being run is:

```
198
199
200 •  SELECT product.* ,select_product.*
  FROM product
  JOIN select_product ON product.pid=select_product.pid
 where select_product.quantity=3;
201
202
203
204
205
206 •  create index idx_sel on select_product(quantity);
207
```

The Result Grid shows the following data:

pid	cd	bid	sid	pname	p_stock	added_date	price	cust_id	pid	quantity
3	1	1	1	Smart Watch	3	2018-10-27	25000	2	3	3
7	3	10	3	glab jamoon mix	50	2012-09-01	5000	1	7	3
5	3	4	3	REFINED OIL	6	2018-10-25	1200	7	5	3
9	2	11	1	Regular fit jeans	20	2012-10-22	50000	5	9	3
4	4	3	2	Air Max	6	2018-10-20	35000	2	4	3
1	1	1	1	iphone	4	2018-10-30	65000	3	1	3
6	3	9	3	sambhar powder	100	2012-10-21	10000	2	6	3
7	3	10	3	glab jamoon mix	50	2012-09-01	5000	7	7	3
4	4	3	2	Air Max	6	2018-10-20	35000	7	4	3
1	1	1	1	iphone	4	2018-10-30	65000	2	1	3
2	1	1	1	Airpods	3	2018-10-27	12000	5	2	3

The Output pane shows the following messages:

- 193 23:45:20 SELECT product.pid,stores.aname FROM product RIGHT JOIN stores ON product.sid=stores.sid ORDER BY ... 29 rows returned Duration / Fetch 0.000 sec / 0.000 sec
- 194 23:46:39 SELECT product.* ,select_product.* FROM product JOIN select_product ON product.pid=select_product.pid where select_product.quantity=3; 11 rows returned Duration / Fetch 0.015 sec / 0.000 sec

The screenshot shows the MySQL Workbench interface with the SQL editor tab active. The code being run is the same as before:

```
198
199
200 •  SELECT product.* ,select_product.*
  FROM product
  JOIN select_product ON product.pid=select_product.pid
 where select_product.quantity=3;
201
202
203
204
205
206 •  create index idx_sel on select_product(quantity);
207
```

The Result Grid shows the same data as the previous screenshot.

The Execution Plan pane displays the following diagram:

```
graph TD
    subgraph query_block_1 [Query cost: 2.41]
        direction TB
        A[Full Table Scan  
selected_product] --> B[nested loop]
        B -- "1 row" --> C[Unique Key Lookup  
product  
PRIMARY]
        C -- "1 row" --> D[query_block_1]
    end
```

The Output pane shows the following messages:

- 195 23:47:26 EXPLAIN SELECT product.* ,select_product.* FROM product JOIN select_product ON product.pid=select_product.pid where select_product.quantity=3; OK Duration / Fetch 0.000 sec / 0.000 sec
- 196 23:47:26 EXPLAIN FORMAT=JSON SELECT product.* ,select_product.* FROM product JOIN select_product ON product.pid=select_product.pid where select_product.quantity=3; OK Duration / Fetch 0.000 sec / 0.000 sec

AFTER

MySQL Workbench

Local instance MySQL80

Schemas: project*, database_design, quiz_sql_file

Code Editor:

```
201 FROM product
202 JOIN select_product ON product.pid=select_product.pid
203 where select_product.quantity=3;
204
205
206 * create index idx_sel on select_product(quantity);
207
```

Visual Explain:

```
graph TD
    A[select_product] --> B[nested loop]
    B --> C[product]
    style A fill:#ff0000,color:#fff
    style C fill:#008000,color:#fff
```

Result 73:

Action	Time	Action	Message	Duration / Fetch
195	23:47:26	EXPLAIN SELECT product.*select_product.* FROM product JOIN select_product ON product.pid=select_product.pid	OK	0.000 sec
196	23:47:26	EXPLAIN FORMAT=JSON SELECT product.*select_product.* FROM product JOIN select_product ON product.pid=select_product.pid	OK	0.000 sec

Object Info | Session

Type here to search

File Edit View Query Database Server Tools Scripting Help

Navigator: pesTug20cs677, cs577_wee8_1*, cs577_week8_2, week9, SQL File 10*, train, SQL File 12*, SQL File 13*, project*, database_design, quiz_sql_file

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