



Database Systems Mini Project

A report submitted to the

Department of Electrical and Information Engineering

Faculty of Engineering

University of Ruhuna

Sri Lanka

On 9th of April 2024

In completing an assignment for the module EE4350

Database Systems

By

EG/2021/4613 – Kodikara .A.W.

EG/2021/4617 – Koshala .M.P.S.

EG/2021/4619 – Kularathna H.E.

Contents

1.	Requirement Analysis	5
1.1.	Functional Requirements	5
1.2	Data Requirements.....	6
2.	Conceptual Design	8
3.	Logical Design.....	9
4.	Implementation	10
4.1.	Create Table	10
4.2.	Inserting	16
4.3.	Update and Delete	24
5.	Transactions	32
5.1.	Simple Queries	32
5.2.	Complex Queries	36
5.3.	Nested Queries.....	41
6.	Database Tuning.....	43

Table of Figures

Figure 2-1 : ER Diagram	8
Figure 3-1 : Logical Database Design.....	9
Figure 4-1	10
Figure 4-2	10
Figure 4-3	11
Figure 4-4	11
Figure 4-5	12
Figure 4-6	12
Figure 4-7	13
Figure 4-8	13
Figure 4-9	14
Figure 4-10	14
Figure 4-11	15
Figure 4-12	16
Figure 4-13	16
Figure 4-14	17
Figure 4-15	17
Figure 4-16	18
Figure 4-17	18
Figure 4-18	19
Figure 4-19	19
Figure 4-20	20
Figure 4-21	20
Figure 4-22	21
Figure 4-23	21
Figure 4-24	22
Figure 4-25	22
Figure 4-26	23
Figure 4-27	23
Figure 4-28	24
Figure 4-29	24
Figure 4-30	25
Figure 4-31	25
Figure 4-32	26
Figure 4-33	26
Figure 4-34	27
Figure 4-35	27
Figure 4-36	28
Figure 4-37	28
Figure 4-38	29
Figure 4-39	29
Figure 4-40	30
Figure 4-41	30
Figure 4-42	31
Figure 4-43	31

Figure 5-1	32
Figure 5-2	32
Figure 5-3	33
Figure 5-4	33
Figure 5-5	34
Figure 5-6	34
Figure 5-7	35
Figure 5-8	36
Figure 5-9	36
Figure 5-10	37
Figure 5-11	37
Figure 5-12	38
Figure 5-13	38
Figure 5-14	39
Figure 5-15	39
Figure 5-16	40
Figure 5-17	40
Figure 5-18	41
Figure 5-19	41
Figure 5-20	42
Figure 6-1	43
Figure 6-2	43
Figure 6-3	44
Figure 6-4	44
Figure 6-5	45
Figure 6-6	45
Figure 6-7	46
Figure 6-8	46
Figure 6-9	47
Figure 6-10	47
Figure 6-11	48
Figure 6-12	48
Figure 6-13	49
Figure 6-14	49
Figure 6-15	50

1. Requirement Analysis

For the mini project of EE4350-Database System module we chose the Cruise Ship Database System as our topic name. Basically under this topic we are planning to build a data base for a company which is own multiple(let's say 5 ships) cruise ships. So that the company can get to know about whose are the employees that working on a particular ship, whose are the passengers that travelling on a particular ship, where is the particular ship located at a particular time and much more. So for the clear understand, we will analyse about functional requirement and data requirement of the data base system.

1.1. Functional Requirements

The company need to know about whose are the workers on a ship and which department particular worker belongs to, whose are the passengers travelling on a ship and there medical conditions, where is a ship located at a particular time and also whether there are maintainance available and so on. So that for the database we used 7 strong entities and 3 weak entities.

Strong Entities :- Ship, Employee, Department, Passenger, Cabin, Port, Cruise

Weak Entities :- Passenger medical history, Cabin service request, Cruise dinning reservation

- Employee is working on a department.
- Employee is responsible for cabin.
- Passenger is travelling on ship.
- Passenger has some health issues. Therefore the weak entity passenger medical history depends on passenger.
- Cabin may have maintainance. Therefore cabin incharge is responsible for cabin service request.
- Ship Is located at a particular port in a particular cruise.

Those are the functional requirements for the database system for the company.

1.2 Data Requirements

The data requirement of this database project under the name Shipping Company has 10 data categories. Basically 3 of them are classified as weak data entities and others are classified as strong Data entities. Entities and attributes can be defined as follows.

Strong entities and attributes.

Employee

- Name
- NIC
- Working ship
- Employee Id
- Permanent address
- Working Department
- Previous worked ships
- Position

Department

- Department head
- Department id
- Department name
- Ship id
- No.of workers

Passenger

- Name
- Contact number
- Passport
- Permanent address
- Passenger id

Ship

- Ship name
- No.of Departments
- Crew capacity
- Ship id

Cabin

- Cabin type
- Deck cabin number
- In charge id

Cruise

- Cruise ship
- Destination port
- Departure date
- Departure port
- Cruise id

Port

- Location
- Ship
- Port name
- Port id

Weak entities and attributes.

Passenger medical history

- Medication
- Medicated data
- Medical condition

Cabin service request

- Service request type
- Request date

Cruise dining reservation

- Number of guests
- Reservation date
- Reservation time

2. Conceptual Design

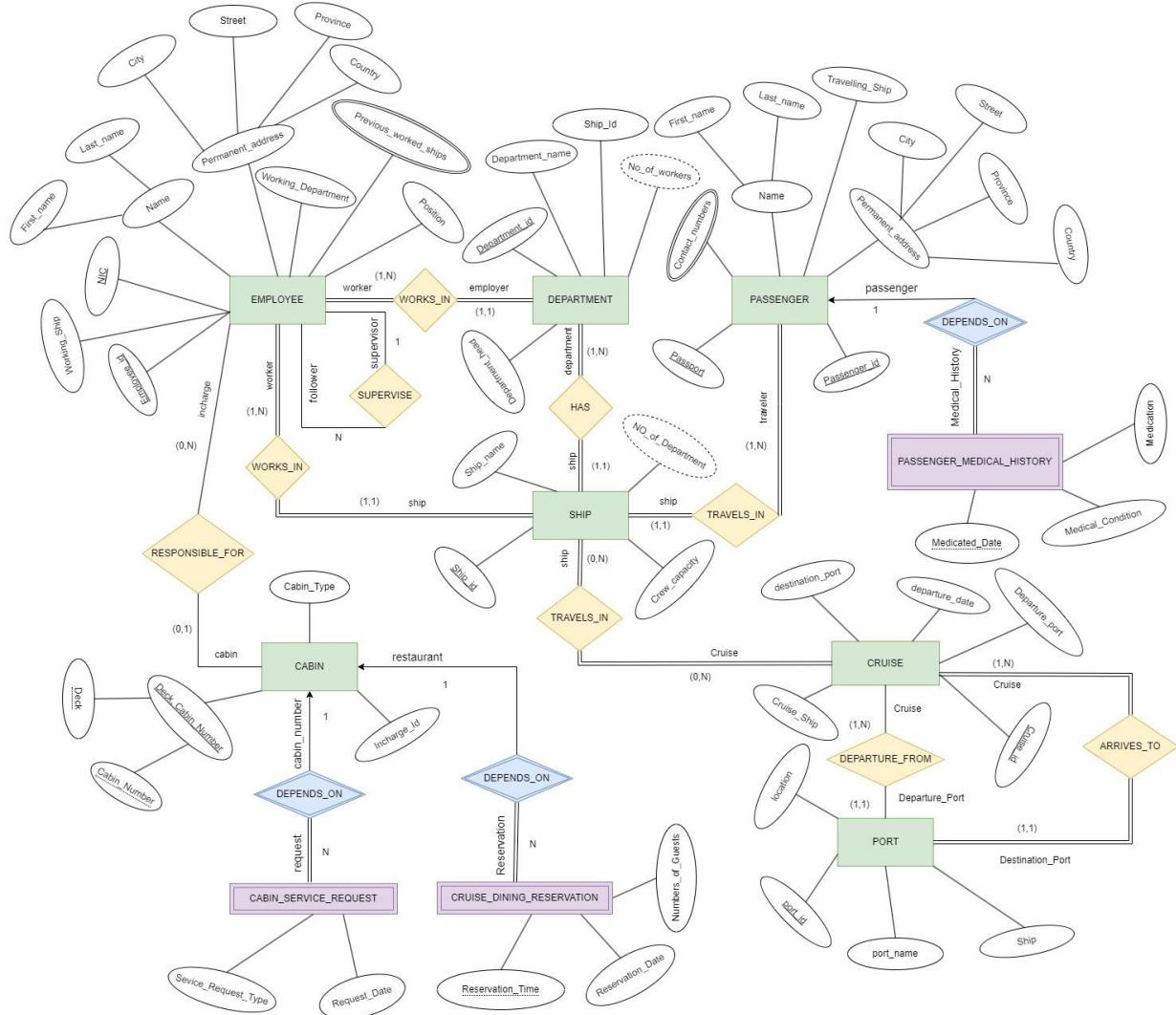


Figure 2-1 : ER Diagram

3. Logical Design

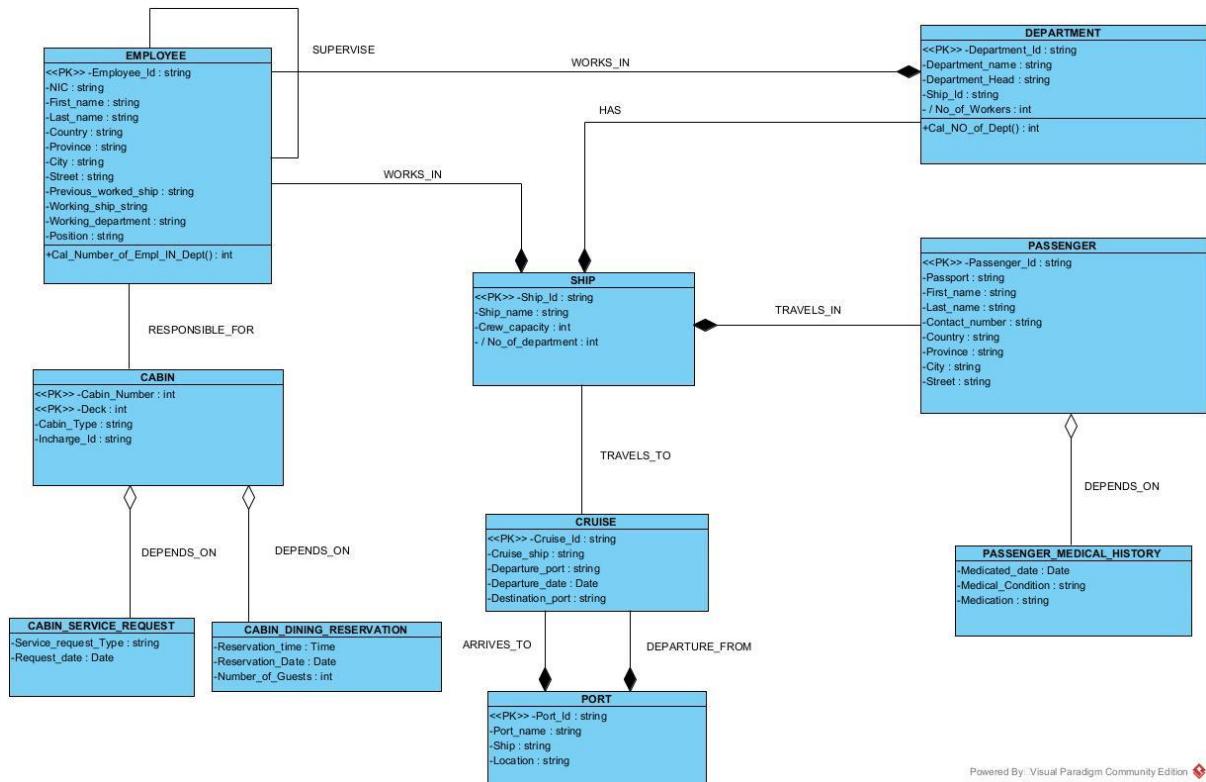
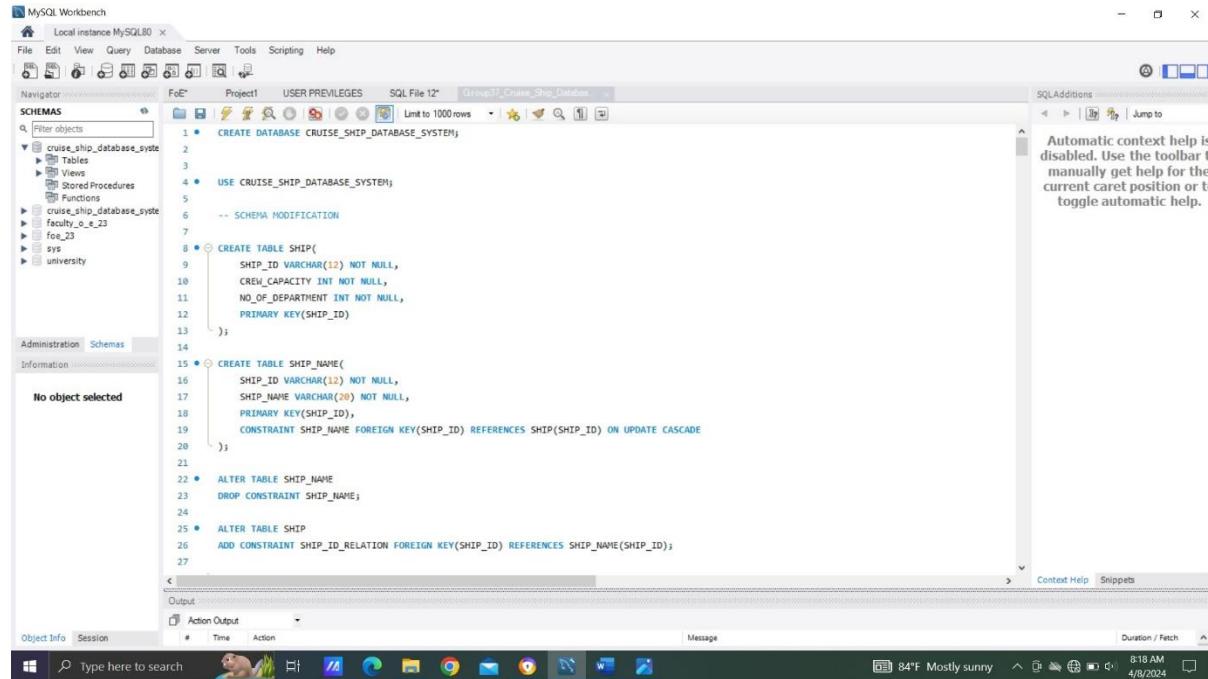


Figure 3-1 : Logical Database Design

4. Implementation

4.1. Create Table



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

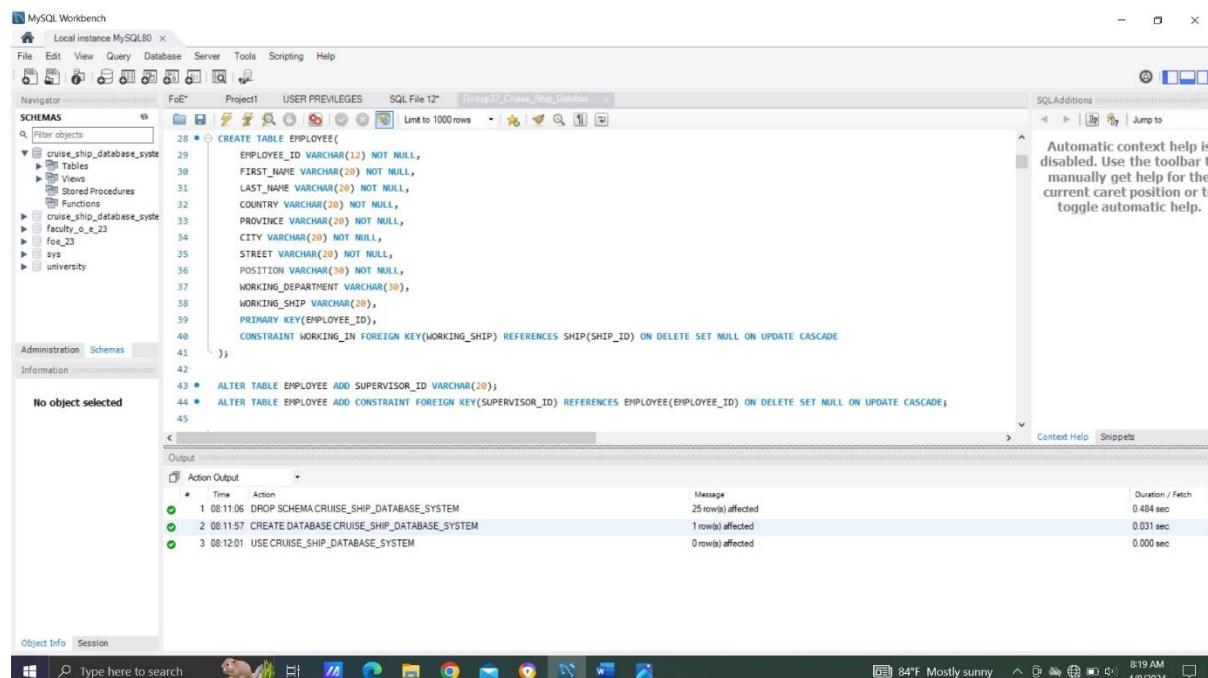
- Schemas:** The current schema is "cruise_ship_database_system".
- SQL Editor:** The code pane contains the following SQL script:

```
CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM;
USE CRUISE_SHIP_DATABASE_SYSTEM;
-- SCHEMA MODIFICATION
CREATE TABLE SHIP(
    SHIP_ID VARCHAR(12) NOT NULL,
    CREW_CAPACITY INT NOT NULL,
    NO_OF_DEPARTMENT INT NOT NULL,
    PRIMARY KEY(SHIP_ID)
);
CREATE TABLE SHIP_NAME(
    SHIP_ID VARCHAR(12) NOT NULL,
    SHIP_NAME VARCHAR(20) NOT NULL,
    PRIMARY KEY(SHIP_ID),
    CONSTRAINT SHIP_NAME FOREIGN KEY(SHIP_ID) REFERENCES SHIP(SHIP_ID) ON UPDATE CASCADE
);
ALTER TABLE SHIP_NAME
DROP CONSTRAINT SHIP_NAME;
ALTER TABLE SHIP
ADD CONSTRAINT SHIP_ID_RELATION FOREIGN KEY(SHIP_ID) REFERENCES SHIP_NAME(SHIP_ID);
```

Output: The output pane shows the results of the executed commands.

#	Time	Action	Message	Duration / Fetch
1	08:11:06	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec

Figure 4-1



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

- Schemas:** The current schema is "cruise_ship_database_system".
- SQL Editor:** The code pane contains the following SQL script:

```
CREATE TABLE EMPLOYEE(
    EMPLOYEE_ID VARCHAR(12) NOT NULL,
    FIRST_NAME VARCHAR(20) NOT NULL,
    LAST_NAME VARCHAR(20) NOT NULL,
    COUNTRY VARCHAR(20) NOT NULL,
    PROVINCE VARCHAR(20) NOT NULL,
    CITY VARCHAR(20) NOT NULL,
    STREET VARCHAR(20) NOT NULL,
    POSITION VARCHAR(30) NOT NULL,
    WORKING_DEPARTMENT VARCHAR(30),
    WORKING_SHIP VARCHAR(30),
    PRIMARY KEY(EMPLOYEE_ID),
    CONSTRAINT WORKING_IN FOREIGN KEY(WORKING_SHIP) REFERENCES SHIP(SHIP_ID) ON DELETE SET NULL ON UPDATE CASCADE
);
ALTER TABLE EMPLOYEE ADD SUPERVISOR_ID VARCHAR(20);
ALTER TABLE EMPLOYEE ADD CONSTRAINT FOREIGN KEY(SUPERVISOR_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON DELETE SET NULL ON UPDATE CASCADE;
```

Output: The output pane shows the results of the executed commands.

#	Time	Action	Message	Duration / Fetch
1	08:11:06	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec

Figure 4-2

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, cruise_ship_database_system (selected).
- SQL Editor:** Contains the following SQL code:


```

CREATE TABLE EMPLOYEE_NIC(
    EMPLOYEE_ID VARCHAR(20) NOT NULL,
    NIC VARCHAR(20) NOT NULL,
    PRIMARY KEY(EMPLOYEE_ID),
    CONSTRAINT EMPLOYEE_NIC FOREIGN KEY(EMPLOYEE_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON UPDATE CASCADE
);

ALTER TABLE EMPLOYEE_NIC DROP CONSTRAINT EMPLOYEE_NIC;

ALTER TABLE EMPLOYEE ADD CONSTRAINT FK_EMPL_ID FOREIGN KEY(EMPLOYEE_ID) REFERENCES EMPLOYEE_NIC(EMPLOYEE_ID);

CREATE TABLE EMPLOYEE_WORKED_SHIP(
    EMPLOYEE_ID VARCHAR(20) NOT NULL,
    PREVIOUS_WORKED_SHIP VARCHAR(20) NOT NULL,
    PRIMARY KEY(EMPLOYEE_ID, PREVIOUS_WORKED_SHIP),
    CONSTRAINT WORKED_SHIP FOREIGN KEY(EMPLOYEE_ID) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON UPDATE CASCADE
);
      
```
- Output Panel:** Shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	08:11:06	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec
- System Tray:** Shows the date and time as 4/8/2024, 8:20 AM.

Figure 4-3

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, cruise_ship_database_system (selected).
- SQL Editor:** Contains the following SQL code (continuation from Figure 4-3):


```

CREATE TABLE DEPARTMENT(
    DEPARTMENT_ID VARCHAR(20) NOT NULL,
    DEPARTMENT_NAME VARCHAR(20) NOT NULL,
    DEPARTMENT_HEAD VARCHAR(20),
    SHIP_ID VARCHAR(20) NOT NULL,
    NO_OF_WORKERS INT NOT NULL,
    PRIMARY KEY(DEPARTMENT_ID),
    CONSTRAINT FK_DEPARTMENT_HEAD FOREIGN KEY(DEPARTMENT_HEAD) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON DELETE SET NULL ON UPDATE CASCADE,
    CONSTRAINT FK_SHIP_ID FOREIGN KEY(SHIP_ID) REFERENCES SHIP(SHIP_ID) ON UPDATE CASCADE
);

ALTER TABLE DEPARTMENT
DROP CONSTRAINT FK_SHIP_ID;

ALTER TABLE DEPARTMENT
ADD CONSTRAINT FK_SHIP_ID FOREIGN KEY(SHIP_ID) REFERENCES SHIP_NAME(SHIP_ID);

ALTER TABLE EMPLOYEE
ADD CONSTRAINT FK_WORKING_DEPARTMENT FOREIGN KEY(WORKING_DEPARTMENT) REFERENCES DEPARTMENT(DEPARTMENT_ID)
ON DELETE SET NULL ON UPDATE CASCADE;
      
```
- Output Panel:** Shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	08:11:06	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec
- System Tray:** Shows the date and time as 4/8/2024, 8:20 AM.

Figure 4-4

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, cruise_ship_database_system.
- SQL Editor:**

```

82 • CREATE TABLE PASSENGER(
83     PASSENGER_ID VARCHAR(20) NOT NULL,
84     FIRST_NAME VARCHAR(20) NOT NULL,
85     LAST_NAME VARCHAR(20) NOT NULL,
86     COUNTRY VARCHAR(20) NOT NULL,
87     PROVINCE VARCHAR(20) NOT NULL,
88     CITY VARCHAR(20) NOT NULL,
89     STREET VARCHAR(20) NOT NULL,
90     TRAVELLING_SHIP VARCHAR(20) NOT NULL,
91     PRIMARY KEY(PASSENGER_ID),
92     CONSTRAINT FK_TRAVELLING_SHIP FOREIGN KEY(TRAVELLING_SHIP) REFERENCES SHIP(SHIP_ID) ON UPDATE CASCADE
93 );
94
95 • ALTER TABLE PASSENGER
96 DROP CONSTRAINT FK_TRAVELLING_SHIP;
97 • ALTER TABLE PASSENGER
98 ADD CONSTRAINT FK_TRAVELLING_SHIP FOREIGN KEY(TRAVELLING_SHIP) REFERENCES SHIP_NAME(SHIP_ID) ON UPDATE CASCADE;

```
- Output:**

#	Time	Action	Message	Duration / Fetch
1	08:11:05	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec
- System Taskbar:** 84°F Mostly sunny, 8:21 AM, 4/8/2024.

Figure 4-5

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, cruise_ship_database_system.
- SQL Editor:**

```

100 • CREATE TABLE PASSENGER_PASSPORT(
101     PASSENGER_ID VARCHAR(20) NOT NULL,
102     PASSPORT VARCHAR(20) NOT NULL,
103     PRIMARY KEY(PASSENGER_ID),
104     CONSTRAINT FK_PASSENGER_PASSPORT FOREIGN KEY(PASSENGER_ID) REFERENCES PASSENGER(PASSENGER_ID) ON UPDATE CASCADE
105 );
106
107 • ALTER TABLE PASSENGER_PASSPORT DROP CONSTRAINT FK_PASSENGER_PASSPORT;
108 • ALTER TABLE PASSENGER ADD CONSTRAINT FK_PSG_ID FOREIGN KEY(PASSENGER_ID) REFERENCES PASSENGER_PASSPORT(PASSENGER_ID);
109
110 • CREATE TABLE PASSENGER_CONTACT(
111     PASSENGER_ID VARCHAR(20) NOT NULL,
112     CONTACT_NUMBER VARCHAR(15) NOT NULL,
113     PRIMARY KEY(PASSENGER_ID, CONTACT_NUMBER),
114     CONSTRAINT FK_PASSENGER_CONTACT FOREIGN KEY(PASSENGER_ID) REFERENCES PASSENGER(PASSENGER_ID) ON UPDATE CASCADE
115 );
116
117 • ALTER TABLE PASSENGER_CONTACT
118 DROP CONSTRAINT FK_PASSENGER_CONTACT;
119 • ALTER TABLE PASSENGER_CONTACT
120 ADD CONSTRAINT FK_PASSENGER_CONTACT FOREIGN KEY(PASSENGER_ID) REFERENCES PASSENGER_PASSPORT(PASSENGER_ID);

```
- Output:**

#	Time	Action	Message	Duration / Fetch
1	08:11:05	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec
- System Taskbar:** 84°F Mostly sunny, 8:21 AM, 4/8/2024.

Figure 4-6

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

- Title Bar:** MySQL Workbench - Local instance MySQL80
- Toolbar:** Standard MySQL Workbench toolbar.
- Navigator:** Shows the schema tree under "cruise_ship_database_system".
- SQL Editor:** Contains the following SQL code (lines 121-340):


```

121 ADD CONSTRAINT FK_PASSENGER_CONTACT FOREIGN KEY(PASSENGER_ID) REFERENCES PASSENGER(PASSPORT(PASSENGER_ID));
122
123 • CREATE TABLE PASSENGER_MEDICAL_HISTORY(
124     PASSENGER VARCHAR(20) NOT NULL,
125     MEDICATED_DATE DATE NOT NULL,
126     MEDICAL_CONDITION VARCHAR(200) NOT NULL,
127     MEDICATION VARCHAR(200) NOT NULL,
128     PRIMARY KEY(PASSENGER, MEDICATED_DATE),
129     CONSTRAINT DEPENDS_ON FOREIGN KEY(PASSENGER) REFERENCES PASSENGER(PASSPORT(PASSENGER_ID)) ON UPDATE CASCADE
130
131 );
132 • ALTER TABLE PASSENGER_MEDICAL_HISTORY DROP FOREIGN KEY DEPENDS_ON;
133 • ALTER TABLE PASSENGER_MEDICAL_HISTORY DROP PRIMARY KEY;
134 • ALTER TABLE PASSENGER_MEDICAL_HISTORY ADD CONSTRAINT DEPENDS_ON FOREIGN KEY(PASSENGER) REFERENCES PASSENGER(PASSPORT(PASSENGER_ID)) ON UPDATE CASCADE;
135
136
137 • CREATE TABLE CABIN(
138     DECK INT NOT NULL,
139     CABIN_NUMBER INT NOT NULL,
140     CABIN_TYPE VARCHAR(10) NOT NULL,
141     PRIMARY KEY(DECK, CABIN_NUMBER)
142 );
143
144 • ALTER TABLE CABIN DROP PRIMARY KEY;
145 • ALTER TABLE CABIN ADD COLUMN SHIP VARCHAR(20) NOT NULL;
146 • ALTER TABLE CABIN ADD PRIMARY KEY(DECK,CABIN_NUMBER,SHIP);
147 • ALTER TABLE CABIN ADD CONSTRAINT FK_CABIN_BELONGS_SHIP FOREIGN KEY(SHIP) REFERENCES SHIP_NAME(SHIP_ID);
      
```
- Output Tab:** Shows the results of the executed commands.
- System Tray:** Displays weather (84°F Mostly sunny), time (8:22 AM), and date (4/8/2024).

Figure 4-7

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

- Title Bar:** MySQL Workbench - Local instance MySQL80
- Toolbar:** Standard MySQL Workbench toolbar.
- Navigator:** Shows the schema tree under "cruise_ship_database_system".
- SQL Editor:** Contains the following SQL code (lines 148-162):


```

148
149 • CREATE TABLE CABIN_INCHARGE(
150     DECK INT NOT NULL,
151     CABIN_NUMBER INT NOT NULL,
152     INCHARGE VARCHAR(20) NOT NULL,
153     PRIMARY KEY(DECK, CABIN_NUMBER, INCHARGE),
154     CONSTRAINT FK_INCHARGE FOREIGN KEY(INCHARGE) REFERENCES EMPLOYEE(EMPLOYEE_ID) ON UPDATE CASCADE,
155     CONSTRAINT FK_CABIN_INCHARGE FOREIGN KEY(DECK, CABIN_NUMBER) REFERENCES CABIN(DECK, CABIN_NUMBER)
156 );
157
158 • ALTER TABLE CABIN_INCHARGE DROP FOREIGN KEY FK_CABIN_INCHARGE;
159 • ALTER TABLE CABIN_INCHARGE DROP PRIMARY KEY;
160 • ALTER TABLE CABIN_INCHARGE ADD COLUMN SHIP VARCHAR(20) NOT NULL;
161 • ALTER TABLE CABIN_INCHARGE ADD PRIMARY KEY(DECK,CABIN_NUMBER, INCHARGE, SHIP);
162 • ALTER TABLE CABIN_INCHARGE ADD CONSTRAINT FK_CABIN_INCHARGE FOREIGN KEY(DECK, CABIN_NUMBER, SHIP) REFERENCES CABIN(DECK, CABIN_NUMBER, SHIP);
      
```
- Output Tab:** Shows the execution log with three entries:

#	Time	Action	Message	Duration / Fetch
1	08:11:06	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec
- System Tray:** Displays weather (84°F Mostly sunny), time (8:22 AM), and date (4/8/2024).

Figure 4-8

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

- Schemas:** The current schema is "cruise_ship_database_system".
- Tables:** A table named "CABIN_SERVICE_REQUEST" is being created.
- SQL Editor:** The SQL code for creating the table is displayed, including constraints like PRIMARY KEY and FOREIGN KEY.
- Output:** The execution log shows three commands:
 - 1 08:11:06 DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM
 - 2 08:11:57 CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM
 - 3 08:12:01 USE CRUISE_SHIP_DATABASE_SYSTEM
- System Bar:** Shows the date as 4/8/2024 and the time as 8:22 AM.

Figure 4-9

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

- Schemas:** The current schema is "cruise_ship_database_system".
- Tables:** A table named "CRUISE_DIMMING_RESERVATION" is being created.
- SQL Editor:** The SQL code for creating the table is displayed, including constraints like PRIMARY KEY and FOREIGN KEY.
- Output:** The execution log shows three commands:
 - 1 08:11:06 DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM
 - 2 08:11:57 CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM
 - 3 08:12:01 USE CRUISE_SHIP_DATABASE_SYSTEM
- System Bar:** Shows the date as 4/8/2024 and the time as 8:22 AM.

Figure 4-10

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

cruise_ship_database_system

Tables Views Stored Procedures Functions

faculty_o_e_23 foe_23 sys university

Administration Schemas Information

No object selected

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	08:11:06	DROP SCHEMA CRUISE_SHIP_DATABASE_SYSTEM	25 row(s) affected	0.484 sec
2	08:11:57	CREATE DATABASE CRUISE_SHIP_DATABASE_SYSTEM	1 row(s) affected	0.031 sec
3	08:12:01	USE CRUISE_SHIP_DATABASE_SYSTEM	0 row(s) affected	0.000 sec

SQLAdditions

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

Context Help Snippets

Object Info Session

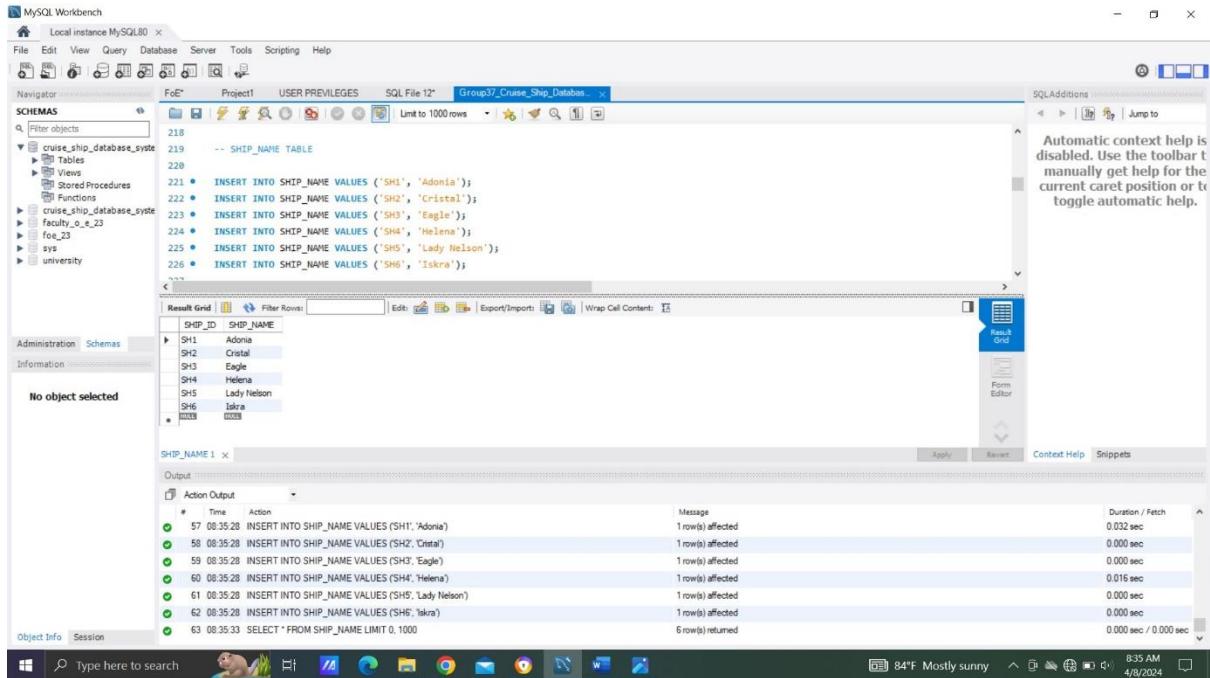
Type here to search

84°F Mostly sunny 8:23 AM 4/8/2024

```
196    CREATE TABLE PORT(
197        PORT_ID VARCHAR(20) NOT NULL,
198        PORT_NAME VARCHAR(30) NOT NULL,
199        LOCATION VARCHAR(100) NOT NULL,
200        PRIMARY KEY(PORT_ID)
201    );
202
203    CREATE TABLE CRUISE(
204        CRUISE_ID VARCHAR(20) NOT NULL,
205        CRUISE_SHIP VARCHAR(20) NOT NULL,
206        DEPARTURE_PORT VARCHAR(30),
207        DEPARTURE_DATE DATE NOT NULL,
208        DESTINATION_PORT VARCHAR(30),
209        PRIMARY KEY(CRUISE_ID),
210        CONSTRAINT FK_CRUISE_SHIP FOREIGN KEY(CRUISE_SHIP) REFERENCES SHIP(SHIP_ID) ON UPDATE CASCADE,
211        CONSTRAINT FK_DEPARTURE_PORT FOREIGN KEY(DEPARTURE_PORT) REFERENCES PORT(PORT_ID) ON DELETE SET NULL ON UPDATE CASCADE,
212        CONSTRAINT FK_DESTINATION_PORT FOREIGN KEY(DESTINATION_PORT) REFERENCES PORT(PORT_ID) ON DELETE SET NULL ON UPDATE CASCADE
213    );
214
215
```

Figure 4-11

4.2. Inserting



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

- Navigator:** Shows the schema `cruise_ship_database_stye` containing tables, views, stored procedures, and functions.
- SQL Editor:** Displays the following SQL code:

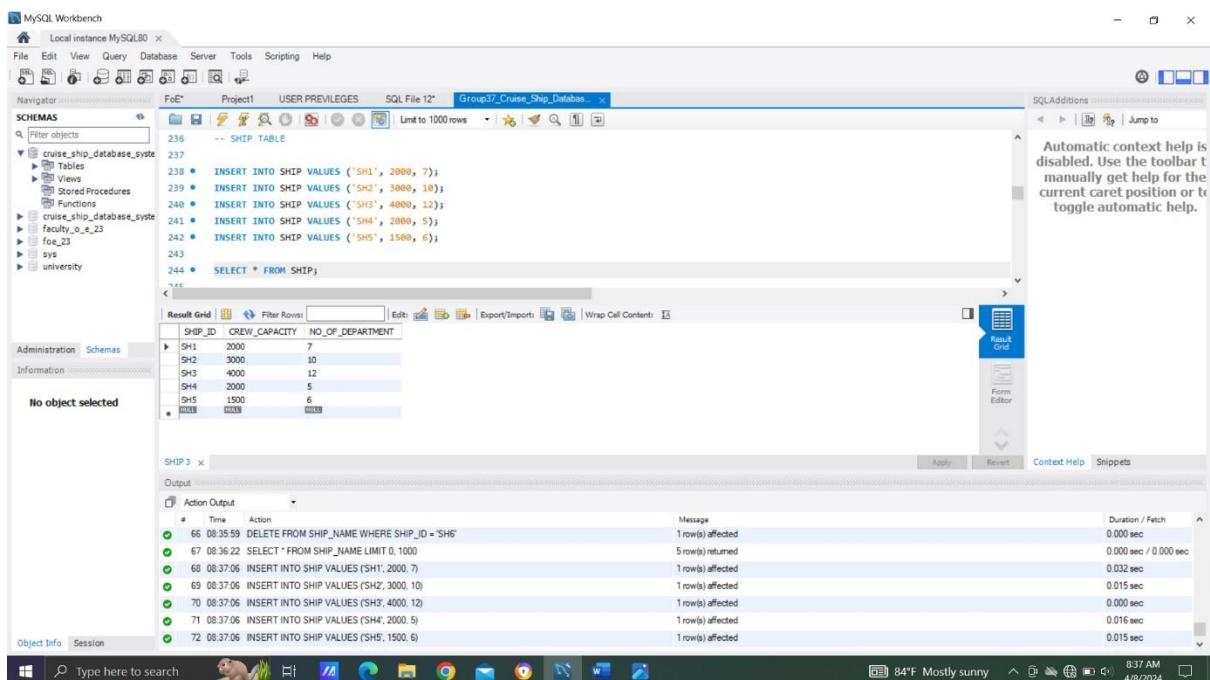

```

219 -- SHIP_NAME TABLE
220
221 • INSERT INTO SHIP_NAME VALUES ('SH1', 'Adonia');
222 • INSERT INTO SHIP_NAME VALUES ('SH2', 'Crystal');
223 • INSERT INTO SHIP_NAME VALUES ('SH3', 'Eagle');
224 • INSERT INTO SHIP_NAME VALUES ('SH4', 'Helena');
225 • INSERT INTO SHIP_NAME VALUES ('SH5', 'Lady Nelson');
226 • INSERT INTO SHIP_NAME VALUES ('SH6', 'Iskra');
      
```
- Result Grid:** Shows the data inserted into the `SHIP_NAME` table:

SHIP_ID	SHIP_NAME
SH1	Adonia
SH2	Crystal
SH3	Eagle
SH4	Helena
SH5	Lady Nelson
SH6	Iskra
NULL	NULL
- Action Output:** Shows the results of the executed statements:

#	Time	Action	Message	Duration / Fetch
57	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH1', 'Adonia')	1 row(s) affected	0.032 sec
58	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH2', 'Crystal')	1 row(s) affected	0.000 sec
59	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH3', 'Eagle')	1 row(s) affected	0.000 sec
60	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH4', 'Helena')	1 row(s) affected	0.015 sec
61	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH5', 'Lady Nelson')	1 row(s) affected	0.000 sec
62	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH6', 'Iskra')	1 row(s) affected	0.000 sec
63	08:35:33	SELECT * FROM SHIP_NAME LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Figure 4-12



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

- Navigator:** Shows the schema `cruise_ship_database_stye` containing tables, views, stored procedures, and functions.
- SQL Editor:** Displays the following SQL code:


```

236 -- SHIP TABLE
237
238 • INSERT INTO SHIP VALUES ('SH1', 2000, 7);
239 • INSERT INTO SHIP VALUES ('SH2', 3000, 10);
240 • INSERT INTO SHIP VALUES ('SH3', 4000, 12);
241 • INSERT INTO SHIP VALUES ('SH4', 2000, 5);
242 • INSERT INTO SHIP VALUES ('SH5', 1500, 6);
243
244 • SELECT * FROM SHIP;
      
```
- Result Grid:** Shows the data inserted into the `SHIP` table:

SHIP_ID	CREW_CAPACITY	NO_OF_DEPARTMENT
SH1	2000	7
SH2	3000	10
SH3	4000	12
SH4	2000	5
SH5	1500	6
NULL	NULL	NULL
- Action Output:** Shows the results of the executed statements:

#	Time	Action	Message	Duration / Fetch
66	08:35:59	DELETE FROM SHIP_NAME WHERE SHIP_ID = 'SH6'	1 row(s) affected	0.000 sec
67	08:36:22	SELECT * FROM SHIP_NAME LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
68	08:37:06	INSERT INTO SHIP VALUES ('SH1', 2000, 7)	1 row(s) affected	0.032 sec
69	08:37:06	INSERT INTO SHIP VALUES ('SH2', 3000, 10)	1 row(s) affected	0.015 sec
70	08:37:06	INSERT INTO SHIP VALUES ('SH3', 4000, 12)	1 row(s) affected	0.000 sec
71	08:37:06	INSERT INTO SHIP VALUES ('SH4', 2000, 5)	1 row(s) affected	0.016 sec
72	08:37:06	INSERT INTO SHIP VALUES ('SH5', 1500, 6)	1 row(s) affected	0.015 sec

Figure 4-13

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_stye

- Tables
- Views
- Stored Procedures
- Functions

faculty_o_e_23

foe_23

sys

university

254 • INSERT INTO EMPLOYEE_NIC VALUES ('SH1EMP1', '200145246325');

256 • INSERT INTO EMPLOYEE_NIC VALUES ('SH1EMP2', '200045786325');

257 • INSERT INTO EMPLOYEE_NIC VALUES ('SH2EMP1', '199945212525');

258 • INSERT INTO EMPLOYEE_NIC VALUES ('SH2EMP3', '199845636575');

259 • INSERT INTO EMPLOYEE_NIC VALUES ('SH2EMP1', '199745249225');

260 • INSERT INTO EMPLOYEE_NIC VALUES ('SH2EMP2', '200145462325');

261 • INSERT INTO EMPLOYEE_NIC VALUES ('SH2EMP2', '200045602325');

262

263 • SELECT * FROM EMPLOYEE_NIC;

Result Grid | Filter Rows | Edit: Export/Import: Wrap Cell Content: T3

EMPLOYEE_ID	NIC
SH1EMP1	200145246325
SH1EMP2	200045786325
SH1EMP3	199945636575
SH2EMP1	199945212525
SH2EMP2	200145462325
SH3EMP1	199745249225
SH3EMP2	200045602325
NULL	NULL

Administration Schemas

No object selected

EMPLOYEE_NIC 5 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
79	08:38:33	INSERT INTO EMPLOYEE_NIC VALUES ('SH1EMP2', '200045786325')	1 row(s) affected	0.00 sec
80	08:38:33	INSERT INTO EMPLOYEE_NIC VALUES ('SH2EMP1', '199945212525')	1 row(s) affected	0.015 sec
81	08:38:33	INSERT INTO EMPLOYEE_NIC VALUES ('SH1EMP3', '199845636575')	1 row(s) affected	0.00 sec
82	08:38:33	INSERT INTO EMPLOYEE_NIC VALUES ('SH3EMP1', '199745249225')	1 row(s) affected	0.00 sec
83	08:38:33	INSERT INTO EMPLOYEE_NIC VALUES ('SH2EMP2', '200145462325')	1 row(s) affected	0.016 sec

Object Info Session

Type here to search

84°F Mostly sunny 8:39 AM 4/8/2024

Figure 4-14

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_stye

- Tables
- Views
- Stored Procedures
- Functions

faculty_o_e_23

foe_23

sys

university

272 -- EMPLOYEE TABLE

273

274 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH1EMP1', 'Senura', 'Koshala', 'Sri Lanka', 'Eatern', 'Uhana', 'DS.Senanayake Street', 'Cjef Engineer', 'SH1');

275 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH1EMP2', 'Wasenith', 'Kodikara', 'Australia', 'Queensland', 'Leydens Hill', '20 Bayview Close', 'Captain', 'SH1');

276 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH1EMP3', 'Masanjith', 'Kodikara', 'Australia', 'Queensland', 'Leydens Hill', '20 Bayview Close', 'Captain', 'SH1');

277 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH2EMP1', 'Hirusha', 'Kularathne', 'Bangladesh', 'Dhaka', 'Dhaka', '7/2, north dhamondi', 'Chief Engineer', 'SH2');

278 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH2EMP2', 'Mruusha', 'Kularathne', 'Bangladesh', 'Dhaka', 'Dhaka', '7/2, north dhamondi', 'Chief Engineer', 'SH2');

279 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH2EMP3', 'David', 'Smith', 'Germany', 'Sachsen-Anhalt', 'Forderstedt', 'Feldstrasse 26', 'Cook', 'SH1');

280 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH3EMP1', 'Kris', 'Gall', 'Jamaica', 'Ocho Rios', 'Ocho Rios', '103 Main St', 'Seaman', 'SH3');

281 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH3EMP2', 'Virat', 'Kholl', 'India', 'Maharashtra', 'Mumbai', 'Tenkar Street', 'Cleaner', 'SH3');

282

283

284 • INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP) VALUES ('SH3EMP1', 'Kris', 'Gall', 'Jamaica', 'Ocho Rios', 'Ocho Rios', '103 Main St', 'Seaman', 'SH3');

285

286

287 • SELECT * FROM EMPLOYEE;

288

Result Grid | Filter Rows | Edit: Export/Import: Wrap Cell Content: T3

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	COUNTRY	PROVINCE	CITY	STREET	POSITION	WORKING_DEPARTMENT	WORKING_SHIP	SUPERVISOR_ID
SH1EMP1	Senura	Koshala	Sri Lanka	Eatern	Uhana	DS.Senanayake Street	Cjef Engineer	SH1	SH1	NULL
SH1EMP2	Wasenith	Kodikara	Australia	Queensland	Leydens Hill	20 Bayview Close	Captain	SH1	SH1	NULL
SH1EMP3	Wasenith	Kodikara	Australia	Queensland	Leydens Hill	20 Bayview Close	Captain	SH1	SH1	NULL
SH2EMP1	Hirusha	Kularathne	Bangladesh	Dhaka	Dhaka	7/2, north dhamondi	Chief Engineer	SH2	SH2	NULL
SH2EMP2	Mruusha	Kularathne	Bangladesh	Dhaka	Dhaka	7/2, north dhamondi	Chief Engineer	SH2	SH2	NULL
SH3EMP1	Kris	Gall	Jamaica	Ocho Rios	Ocho Rios	103 Main St	Seaman	SH3	SH3	NULL
SH3EMP2	Virat	Kholl	India	Maharashtra	Mumbai	Tenkar Street	Cleaner	SH3	SH3	NULL
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Administration Schemas

No object selected

EMPLOYEE 7 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
91	08:40:47	INSERT INTO EMPLOYEE (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, COUNTRY, PROVINCE, CITY, STREET, POSITION, WORKING_SHIP)	1 row(s) affected	0.00 sec

Object Info Session

Type here to search

84°F Mostly sunny 8:41 AM 4/8/2024

Figure 4-15

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS cruise_ship_database_system

Tables Views Stored Procedures Functions

Faculty_o_e_23 foe_23 sys university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database.x

Limit to 1000 rows

298 -- EMPLOYEE_WORKED_SHIP TABLE

300

301 • INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH1EMP1', 'Moby Baby');

302 • INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH1EMP1', 'Oceanic');

303 • INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH1EMP2', 'Pacific Dream');

304 • INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH1EMP2', 'Queen Mary 2');

305 • INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH2EMP1', 'Rawfath');

306 • INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH3EMP1', 'Rosella');

307

308 • SELECT * FROM EMPLOYEE_WORKED_SHIP;

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

Administration Schemas Information No object selected

EMPLOYEE_WORKED_SHIP #

Output

Action Output

#	Time	Action	Message	Duration / Fetch
106	08:43:33	INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH1EMP2', 'Pacific Dream')	1 row(s) affected	0.016 sec
107	08:43:33	INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH1EMP2', 'Queen Mary 2')	1 row(s) affected	0.015 sec
108	08:43:33	INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH2EMP1', 'Rawfath')	1 row(s) affected	0.000 sec
109	08:43:33	INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH3EMP1', 'Rosella')	1 row(s) affected	0.000 sec
110	08:43:33	SELECT * FROM EMPLOYEE_WORKED_SHIP LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 8:43 AM 4/8/2024

Figure 4-16

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS cruise_ship_database_system

Tables Views Stored Procedures Functions

Faculty_o_e_23 foe_23 sys university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database.x

Limit to 1000 rows

317 -- DEPARTMENT TABLE

318

319 • INSERT INTO DEPARTMENT VALUES ('SH1DP1', 'Engineering Department', 'SH1EMP2', 'SH1', 50);

320 • INSERT INTO DEPARTMENT VALUES ('SH1DP3', 'Food and Beverage Department', 'SH1EMP2', 'SH3', 75);

321 • INSERT INTO DEPARTMENT VALUES ('SH1DP2', 'Deck Department', 'SH1EMP2', 'SH1', 100);

322 • INSERT INTO DEPARTMENT VALUES ('SH2DP1', 'Entertainment Department', NULL, 'SH2', 50);

323 • INSERT INTO DEPARTMENT VALUES ('SH3DP2', 'Engineering Department', NULL, 'SH3', 60);

324 • INSERT INTO DEPARTMENT VALUES ('SH1DP4', 'Security Department', NULL, 'SH1', 50);

325

326 • SELECT * FROM DEPARTMENT;

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

Administration Schemas Information No object selected

DEPARTMENT #

Output

Action Output

#	Time	Action	Message	Duration / Fetch
115	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH1DP1', 'Engineering Department', 'SH1EMP2', 'SH1', 50)	1 row(s) affected	0.031 sec
116	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH1DP3', 'Food and Beverage Department', 'SH1EMP2', 'SH3', 75)	1 row(s) affected	0.016 sec
117	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH1DP2', 'Deck Department', 'SH1EMP2', 'SH1', 100)	1 row(s) affected	0.015 sec
118	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH2DP1', 'Entertainment Department', NULL, 'SH2', 50)	1 row(s) affected	0.000 sec
119	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH3DP2', 'Engineering Department', NULL, 'SH3', 60)	1 row(s) affected	0.000 sec
120	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH1DP4', 'Security Department', NULL, 'SH1', 50)	1 row(s) affected	0.015 sec

Object Info Session

Type here to search 84°F Mostly sunny 8:45 AM 4/8/2024

Figure 4-17

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

Tables

Views

Stored Procedures

Functions

faculty_o_e_23

foe_23

sys

university

Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database

Limit to 1000 rows

339 -- PASSENGER_PASSPORT TABLE

340 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH1P\$1', '199720013005');

343 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH1P\$2', '199562343005');

344 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH2P\$3', '894560013005');

345 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH3P\$2', '152100013005');

346 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH3P\$5', '178320013005');

347 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH3P\$8', '200020013005');

348 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH1P\$7', '100230013005');

349

350 • SELECT * FROM PASSENGER_PASSPORT;

Administration Schemas

Result Grid Filter Rows:

PASSENGER_ID	PASSPORT
SH1P\$1	199720013005
SH1P\$2	199562343005
SH1P\$7	100230013005
SH2P\$3	894560013005
SH3P\$2	152100013005
SH3P\$5	178320013005
SH3P\$8	200020013005
SH3P\$9	100000000000

No object selected

PASSENGER_PASSPORT 13

Action Output

#	Time	Action	Message	Duration / Fetch
131	09:50:37	INSERT INTO PASSENGER_PASSPORT VALUES ('SH1P\$2', '199562343005')	1 row(s) affected	0.000 sec
132	09:50:38	INSERT INTO PASSENGER_PASSPORT VALUES ('SH2P\$3', '894560013005')	1 row(s) affected	0.000 sec
133	09:50:38	INSERT INTO PASSENGER_PASSPORT VALUES ('SH3P\$2', '152100013005')	1 row(s) affected	0.016 sec

Object Info Session

Output

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

Tables

Views

Stored Procedures

Functions

faculty_o_e_23

foe_23

sys

university

Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database

Limit to 1000 rows

357 • SELECT * FROM PASSENGER_PASSPORT;

358

359 -- PASSENGER TABLE

360

361 • INSERT INTO PASSENGER VALUES ('SH1P\$1', 'Vidara', 'Supun', 'Sri Lanka', 'Central', 'Digana', 'Light Lane', 'sh1');

362 • INSERT INTO PASSENGER VALUES ('SH1P\$2', 'Udithe', 'Chavishan', 'India', 'Karnataka', 'Bangalore', '23, Hosur Main Road', 'sh1');

363 • INSERT INTO PASSENGER VALUES ('SH2P\$3', 'Mathew', 'David', 'Germany', 'Freistaat Bayern', 'Bad Steben', 'Billwerder Neuer', 'sh2');

364 • INSERT INTO PASSENGER VALUES ('SH3P\$2', 'Mitchel', 'Starc', 'Finland', 'Southern Savonia', 'Savonlinna', 'Liisankatu 39', 'sh3');

365 • INSERT INTO PASSENGER VALUES ('SH3P\$5', 'Tim', 'Southee', 'South Africa', 'Queensland', 'Uhana', '19 Creek Street', 'sh3');

366 • INSERT INTO PASSENGER VALUES ('SH3P\$8', 'David', 'Warner', 'Australia', 'Queensland', 'Jondaryan', '19 Creek Street', 'sh3');

367

368 • SELECT * FROM PASSENGER;

6 rows

No object selected

PASSENGER 15

Action Output

#	Time	Action	Message	Duration / Fetch
146	09:52:17	INSERT INTO PASSENGER VALUES ('SH3P\$8', 'Tim', 'Southee', 'South Africa', 'Queensland', 'Uhana', '19 Cre...')	1 row(s) affected	0.000 sec
147	09:52:17	INSERT INTO PASSENGER VALUES ('SH3P\$5', 'David', 'Warner', 'Australia', 'Queensland', 'Jondaryan', '19 Cr...')	1 row(s) affected	0.000 sec
148	09:52:17	SELECT * FROM PASSENGER LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Output

Figure 4-18

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

Tables

Views

Stored Procedures

Functions

faculty_o_e_23

foe_23

sys

university

Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database

Limit to 1000 rows

357 • SELECT * FROM PASSENGER_PASSPORT;

358

359 -- PASSENGER TABLE

360

361 • INSERT INTO PASSENGER VALUES ('SH1P\$1', 'Vidara', 'Supun', 'Sri Lanka', 'Central', 'Digana', 'Light Lane', 'sh1');

362 • INSERT INTO PASSENGER VALUES ('SH1P\$2', 'Udithe', 'Chavishan', 'India', 'Karnataka', 'Bangalore', '23, Hosur Main Road', 'sh1');

363 • INSERT INTO PASSENGER VALUES ('SH2P\$3', 'Mathew', 'David', 'Germany', 'Freistaat Bayern', 'Bad Steben', 'Billwerder Neuer', 'sh2');

364 • INSERT INTO PASSENGER VALUES ('SH3P\$2', 'Mitchel', 'Starc', 'Finland', 'Southern Savonia', 'Savonlinna', 'Liisankatu 39', 'sh3');

365 • INSERT INTO PASSENGER VALUES ('SH3P\$5', 'Tim', 'Southee', 'South Africa', 'Queensland', 'Uhana', '19 Creek Street', 'sh3');

366 • INSERT INTO PASSENGER VALUES ('SH3P\$8', 'David', 'Warner', 'Australia', 'Queensland', 'Jondaryan', '19 Creek Street', 'sh3');

367

368 • SELECT * FROM PASSENGER;

6 rows

No object selected

PASSENGER 15

Action Output

#	Time	Action	Message	Duration / Fetch
146	09:52:17	INSERT INTO PASSENGER VALUES ('SH3P\$8', 'Tim', 'Southee', 'South Africa', 'Queensland', 'Uhana', '19 Cre...')	1 row(s) affected	0.000 sec
147	09:52:17	INSERT INTO PASSENGER VALUES ('SH3P\$5', 'David', 'Warner', 'Australia', 'Queensland', 'Jondaryan', '19 Cr...')	1 row(s) affected	0.000 sec
148	09:52:17	SELECT * FROM PASSENGER LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Output

Figure 4-19

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Tables

Views

Stored Procedures

Functions

cruise_ship_database_system

faculty_o_e_23

foe_23

sys

university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database.x

375 • SELECT * FROM PASSENGER;

376

377 -- PASSENGER_CONTACT TABLE

378

379 • INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$1', '0766258105');

380 • INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$1', '0716258105');

381 • INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$2', '0756285105');

382 • INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$2', '0704558105');

383 • INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$2', '0766782105');

384 • INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$5', '0766250005');

385

386 • SELECT * FROM PASSENGER_CONTACT;

Result Grid

PASSENGER_ID CONTACT_NUMBER

SH1P\$1	0716258105
SH1P\$1	0766258105
SH1P\$2	0756285105
SH1P\$2	0704558105
SH1P\$2	0766782105
SH1P\$5	0766250005
NULL	NULL

No object selected

PASSENGER_CONTACT_17

Action Output

#	Time	Action	Message	Duration / Fetch
154	09:53:53	INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$1', '0716258105')	1 row(s) affected	0.016 sec
155	09:53:53	INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$2', '0756285105')	1 row(s) affected	0.000 sec
156	09:53:53	INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$2', '0704558105')	1 row(s) affected	0.000 sec
157	09:53:53	INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$2', '0766782105')	1 row(s) affected	0.016 sec
158	09:53:53	INSERT INTO PASSENGER_CONTACT VALUES ('SH1P\$5', '0766250005')	1 row(s) affected	0.000 sec

Object Info Session

Type here to search

84°F Mostly sunny 954 AM 4/8/2024

Figure 4-20

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Tables

Views

Stored Procedures

Functions

cruise_ship_database_system

faculty_o_e_23

foe_23

sys

university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database.x

393 • SELECT * FROM PASSENGER_CONTACT;

394

395 -- PASSENGER_MEDICAL_HISTORY

396

397

398 • INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$1', '2023-12-05', 'Depression', 'Anxiolytics or sedatives');

399 • INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$1', '2023-12-08', 'Motion Sickness', 'Dimenhydrinate (Dramamine), Meclizine (Bonine, Antivert)');

400 • INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$2', '2023-12-25', 'Dehydration', 'Oral rehydration solutions (e.g., Pedialyte)');

401 • INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$2', '2023-12-18', 'Gastrointestinal Issues', 'Antidiarrheal medications (e.g., loperamide/Iimo...');

402 • INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$2', '2023-12-20', 'Respiratory Infections', 'Decongestants');

403 • INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$5', '2023-12-30', 'Injuries', 'Analgesics for pain relief');

404

405 • SELECT * FROM PASSENGER_MEDICAL_HISTORY;

406

Result Grid

PASSENGER MEDICATED_DATE MEDICAL_CONDITION MEDICATION

SH1P\$1	2023-12-05	Depression	Anxiolytics or sedatives
SH1P\$1	2023-12-08	Motion Sickness	Dimenhydrinate (Dramamine), Meclizine (Bonine, Antivert)
SH1P\$2	2023-12-25	Dehydration	Oral rehydration solutions (e.g., Pedialyte)
SH1P\$2	2023-12-18	Gastrointestinal Issues	Antidiarrheal medications (e.g., loperamide/Iimo...)
SH1P\$2	2023-12-20	Respiratory Infections	Decongestants
SH1P\$5	2023-12-30	Injuries	Analgesics for pain relief

No object selected

PASSENGER_MEDICAL_HISTORY

Action Output

#	Time	Action	Message	Duration / Fetch
167	09:55:26	INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$1', '2023-12-18', 'Gastrointestinal Issues', 'Antidiarrheal medications (e.g., loperamide/Iimo...)	1 row(s) affected	0.000 sec
168	09:55:26	INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$2', '2023-12-20', 'Respiratory Infections', 'Decongestants')	1 row(s) affected	0.000 sec
169	09:55:26	INSERT INTO PASSENGER_MEDICAL_HISTORY VALUES ('SH1P\$5', '2023-12-30', 'Injuries', 'Analgesics for pain relief')	1 row(s) affected	0.015 sec
170	09:55:26	SELECT * FROM PASSENGER_MEDICAL_HISTORY LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

Near record 955 AM 4/8/2024

Figure 4-21

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS Filter objects

- cruise_ship_database_stye
 - Tables
 - Views
 - Stored Procedures
 - Functions
- faculty_o_e_23
- foe_23
- sys
- university

FoE Project1 USER PREVILEGES SQL File 12* Group37_Create_Ship_Database.x

414 -- CABIN TABLE

```

415 • INSERT INTO CABIN VALUES (2, 24, 'Dinning Room 1', 'SH1');
417 • INSERT INTO CABIN VALUES (2, 35, 'Dinning Room 2', 'SH1');
418 • INSERT INTO CABIN VALUES (3, 31, 'Ocean view cabin', 'SH3');
419 • INSERT INTO CABIN VALUES (3, 15, 'Family Cabin 1', 'SH3');
420 • INSERT INTO CABIN VALUES (4, 23, 'Suite', 'SH1');
421 • INSERT INTO CABIN VALUES (5, 30, 'Balcony Cabin', 'SH2');

423 • SELECT * FROM CABIN;
424

```

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

DECK	CABIN_NUMBER	CABIN_TYPE	SHIP
2	24	Dinning Room 1	SH1
2	35	Dinning Room 2	SH1
3	15	Family Cabin 1	SH3
3	31	Ocean view cabin	SH3
4	23	Suite	SH1
5	30	Balcony Cabin	SH2
*	NULL	NULL	NULL

SQLAdditions SQL Additions

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

Administration Schemas Information No object selected Object Info Session

CABIN 22 x Output

Action Output

#	Time	Action	Message	Duration / Fetch
183	09:57:15	INSERT INTO CABIN VALUES (4, 23, 'Suite', 'SH1')	1 row(s) affected	0.015 sec
184	09:57:15	INSERT INTO CABIN VALUES (5, 30, 'Balcony Cabin', 'SH2')	1 row(s) affected	0.016 sec
185	09:57:15	SELECT * FROM CABIN LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

84°F Mostly sunny 9:57 AM 4/8/2024

Figure 4-22

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS Filter objects

- cruise_ship_database_stye
 - Tables
 - Views
 - Stored Procedures
 - Functions
- faculty_o_e_23
- foe_23
- sys
- university

FoE Project1 USER PREVILEGES SQL File 12* Group37_Create_Ship_Database.x

433

```

434 • INSERT INTO CABIN_INCHARGE VALUES ('3', '35', 'SH1EMP1', 'SH1');
435 • INSERT INTO CABIN_INCHARGE VALUES ('10', '1', 'SH1EMP2', 'SH1');
436 • INSERT INTO CABIN_INCHARGE VALUES ('3', '31', 'SH2EMP1', 'SH3');
437 • INSERT INTO CABIN_INCHARGE VALUES ('3', '31', 'SH3EMP2', 'SH3');
438 • INSERT INTO CABIN_INCHARGE VALUES ('2', '35', 'SH1EMP2', 'SH1');
439 • INSERT INTO CABIN_INCHARGE VALUES ('5', '30', 'SH3EMP1', 'SH2');

441 • SELECT * FROM CABIN_INCHARGE;
442

```

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

DECK	CABIN_NUMBER	INCHARGE	SHIP
2	35	SH1EMP1	SH1
2	35	SH1EMP2	SH1
10	1	SH2EMP1	SH1
3	31	SH3EMP1	SH2
5	30	SH3EMP1	SH2
3	31	SH2EMP2	SH3
*	NULL	NULL	NULL

CABIN_INCHARGE 24 x Output

Action Output

#	Time	Action	Message	Duration / Fetch
192	09:58:47	INSERT INTO CABIN_INCHARGE VALUES ('3', '31', 'SH2EMP1', 'SH1')	1 row(s) affected	0.016 sec
193	09:58:47	INSERT INTO CABIN_INCHARGE VALUES ('3', '31', 'SH3EMP2', 'SH3')	1 row(s) affected	0.000 sec
194	09:58:47	INSERT INTO CABIN_INCHARGE VALUES ('2', '35', 'SH1EMP2', 'SH1')	1 row(s) affected	0.015 sec
195	09:58:47	INSERT INTO CABIN_INCHARGE VALUES ('5', '30', 'SH3EMP1', 'SH2')	1 row(s) affected	0.016 sec
196	09:58:47	SELECT * FROM CABIN_INCHARGE LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

84°F Mostly sunny 9:58 AM 4/8/2024

Figure 4-23

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

Filter objects

SCHEMAS

- cruise_ship_database_system
- Tables
- Views
- Stored Procedures
- Functions
- Faculty_o_e_23
- foe_23
- sys
- university

451 • INSERT INTO CABIN_SERVICE_REQUEST VALUES (2, 35, '2023-12-10', 'Table Broken', 'SH1');

452 • INSERT INTO CABIN_SERVICE_REQUEST VALUES (2, 35, '2023-12-25', 'Chair Broken', 'SH1');

453 • INSERT INTO CABIN_SERVICE_REQUEST VALUES (3, 15, '2023-12-20', 'Need to fix the cupboard', 'SH3');

454 • INSERT INTO CABIN_SERVICE_REQUEST VALUES (3, 31, '2023-12-22', 'Bed Broken', 'SH1');

455 • INSERT INTO CABIN_SERVICE_REQUEST VALUES (3, 31, '2023-12-30', 'Need a Chair', 'SH3');

456 • INSERT INTO CABIN_SERVICE_REQUEST VALUES (5, 15, '2023-12-05', 'Need to fix the cupboard', 'SH3');

457 • INSERT INTO CABIN_SERVICE_REQUEST VALUES (5, 30, '2023-12-10', 'Tile broken', 'SH2');

458

459 • SELECT * FROM CABIN_SERVICE_REQUEST;

460

461

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

DECK	CABIN_NUMBER	REQUEST_DATE	SERVICE_REQUEST_TYPE	SHIP
2	35	2023-12-10	Table Broken	SH1
2	35	2023-12-25	Chair Broken	SH1
3	15	2023-12-20	Need to fix the cupboard	SH3
3	31	2023-12-20	Need a Chair	SH3
5	30	2023-12-10	Tile broken	SH2
10	1	2023-12-22	Bed Broken	SH1
•	NULL	NULL	NULL	NULL

Administration Schemas Information No object selected

CABIN_SERVICE_REQUEST.27 ×

Output

Action Output

- Time Action Message Duration / Fetch
- 207 10:00:06 INSERT INTO CABIN_SERVICE_REQUEST VALUES (10, 1, '2023-12-22', 'Bed Broken', 'SH1') 1 row(s) affected 0.016 sec
- 208 10:00:06 INSERT INTO CABIN_SERVICE_REQUEST VALUES (3, 31, '2023-12-30', 'Need a Chair', 'SH3') 1 row(s) affected 0.000 sec
- 209 10:00:06 INSERT INTO CABIN_SERVICE_REQUEST VALUES (3, 15, '2023-12-05', 'Need to fix the cupboard', 'SH3') 1 row(s) affected 0.000 sec
- 210 10:00:06 INSERT INTO CABIN_SERVICE_REQUEST VALUES (5, 30, '2023-12-10', 'Tile broken', 'SH2') 1 row(s) affected 0.016 sec
- 211 10:00:06 SELECT * FROM CABIN_SERVICE_REQUEST LIMIT 0, 1000 6 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

Windows Taskbar: Type here to search, 84°F Mostly sunny, 10:00 AM, 4/8/2024

Figure 4-24

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

Filter objects

SCHEMAS

- cruise_ship_database_system
- Tables
- Views
- Stored Procedures
- Functions
- Faculty_o_e_23
- foe_23
- sys
- university

468 -- CRUISE_DINNING_RESERVATION TABLE

469

470 • INSERT INTO CRUISE_DINNING_RESERVATION VALUES (2, 35, '2023-12-11', '13:30:00', 5, 'SH1');

471 • INSERT INTO CRUISE_DINNING_RESERVATION VALUES (2, 35, '2023-12-12', '17:00:00', 5, 'SH1');

472 • INSERT INTO CRUISE_DINNING_RESERVATION VALUES (2, 35, '2023-12-13', '20:30:00', 5, 'SH1');

473 • INSERT INTO CRUISE_DINNING_RESERVATION VALUES (3, 31, '2023-12-11', '15:30:00', 10, 'SH3');

474 • INSERT INTO CRUISE_DINNING_RESERVATION VALUES (3, 31, '2023-12-25', '19:00:00', 12, 'SH3');

475 • INSERT INTO CRUISE_DINNING_RESERVATION VALUES (3, 31, '2023-12-02', '11:30:00', 7, 'SH3');

476

477 • SELECT * FROM CRUISE_DINNING_RESERVATION;

478

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

DECK	CABIN_NUMBER	RESERVATION_DATE	RESERVATION_TIME	NUMBER_OF_GUESTS	SHIP
2	35	2023-12-11	13:30:00	5	SH1
2	35	2023-12-12	17:00:00	5	SH1
2	35	2023-12-13	20:30:00	5	SH1
3	31	2023-12-02	11:30:00	7	SH3
3	31	2023-12-11	15:30:00	10	SH3
3	31	2023-12-25	19:00:00	12	SH3
•	NULL	NULL	NULL	NULL	NULL

CRUISE_DINNING_RESERVATION.28 ×

Output

Action Output

- Time Action Message Duration / Fetch
- 218 10:01:41 INSERT INTO CRUISE_DINNING_RESERVATION VALUES (2, 35, '2023-12-11', '13:30:00', 5, 'SH1') 1 row(s) affected 0.000 sec
- 219 10:01:41 INSERT INTO CRUISE_DINNING_RESERVATION VALUES (3, 31, '2023-12-11', '15:30:00', 10, 'SH3') 1 row(s) affected 0.000 sec
- 220 10:01:41 INSERT INTO CRUISE_DINNING_RESERVATION VALUES (3, 31, '2023-12-25', '19:00:00', 12, 'SH3') 1 row(s) affected 0.016 sec
- 221 10:01:41 INSERT INTO CRUISE_DINNING_RESERVATION VALUES (3, 31, '2023-12-02', '11:30:00', 7, 'SH3') 1 row(s) affected 0.015 sec
- 222 10:01:41 SELECT * FROM CRUISE_DINNING_RESERVATION LIMIT 0, 1000 6 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

Windows Taskbar: Type here to search, 84°F Mostly sunny, 10:02 AM, 4/8/2024

Figure 4-25

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

Tables Views Stored Procedures Functions

faculty_o_e_23 foe_23 sys university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database.x

486 -- PORT TABLE

```

488 • INSERT INTO PORT VALUES ('PT1', 'Miami', 'Florida, USA')
489 • INSERT INTO PORT VALUES ('PT2', 'Port Everglades', 'Florida, USA')
490 • INSERT INTO PORT VALUES ('PT3', 'Nassau', 'Bahamas')
491 • INSERT INTO PORT VALUES ('PT4', 'Cozumel', 'Mexico')
492 • INSERT INTO PORT VALUES ('PT5', 'St. Thomas', 'US Virgin Islands')
493 • INSERT INTO PORT VALUES ('PT6', 'Barcelona', 'Spain')
494
495 • SELECT * FROM PORT;
496

```

Result Grid

PORT_ID	PORT_NAME	LOCATION
PT1	Miami	Florida, USA
PT2	Port Everglades	Florida, USA
PT3	Nassau	Bahamas
PT4	Cozumel	Mexico
PT5	St. Thomas	US Virgin Islands
PT6	Barcelona	Spain

PORT 31 x

Output

Action Output

Time	Action	Message	Duration / Fetch
229	10:04:28 INSERT INTO PORT VALUES (PT3, 'Nassau', 'Bahamas')	1 row(s) affected	0.000 sec
230	10:04:28 INSERT INTO PORT VALUES (PT4, 'Cozumel', 'Mexico')	1 row(s) affected	0.016 sec
231	10:04:28 INSERT INTO PORT VALUES (PT5, 'St. Thomas', 'US Virgin Islands')	1 row(s) affected	0.000 sec
232	10:04:28 INSERT INTO PORT VALUES (PT6, 'Barcelona', 'Spain')	1 row(s) affected	0.015 sec
233	10:04:28 SELECT * FROM PORT LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

84°F Mostly sunny 10:04 AM 4/8/2024

Figure 4-26

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

Tables Views Stored Procedures Functions

faculty_o_e_23 foe_23 sys university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database.x

504 -- CRUISE TABLE

```

505
506 • INSERT INTO CRUISE VALUES ('CR1', 'SH1', 'PT1', '2023-11-30', 'PT2');
507 • INSERT INTO CRUISE VALUES ('CR2', 'SH1', 'PT6', '2023-10-28', 'PT2');
508 • INSERT INTO CRUISE VALUES ('CR3', 'SH3', 'PT5', '2023-12-01', 'PT3');
509 • INSERT INTO CRUISE VALUES ('CR4', 'SH3', 'PT3', '2023-12-30', 'PT4');
510 • INSERT INTO CRUISE VALUES ('CR5', 'SH2', 'PT1', '2023-11-25', 'PT6');
511 • INSERT INTO CRUISE VALUES ('CR6', 'SH2', 'PT3', '2023-12-10', 'PT2');
512
513 • SELECT * FROM CRUISE;
514

```

Result Grid

CRUISE_ID	CRUISE_SHIP	DEPARTURE_PORT	DEPARTURE_DATE	DESTINATION_PORT
CR1	SH1	PT1	2023-11-30	PT2
CR2	SH1	PT6	2023-10-28	PT2
CR3	SH3	PT2	2023-12-01	PT3
CR4	SH3	PT3	2023-12-30	PT4
CR5	SH2	PT1	2023-11-25	PT6
CR6	SH2	PT3	2023-12-10	PT2

CRUISE 33 x

Output

Action Output

Time	Action	Message	Duration / Fetch
240	10:05:25 INSERT INTO CRUISE VALUES (CR3, SH3, PT2, 2023-12-01, PT3)	1 row(s) affected	0.016 sec
241	10:05:25 INSERT INTO CRUISE VALUES (CR4, SH3, PT3, 2023-12-30, PT4)	1 row(s) affected	0.000 sec
242	10:05:25 INSERT INTO CRUISE VALUES (CR5, SH2, PT1, 2023-11-25, PT6)	1 row(s) affected	0.000 sec
243	10:05:25 INSERT INTO CRUISE VALUES (CR6, SH2, PT3, 2023-12-10, PT2)	1 row(s) affected	0.015 sec
244	10:05:25 SELECT * FROM CRUISE LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

84°F Mostly sunny 10:05 AM 4/8/2024

Figure 4-27

4.3. Update and Delete

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

- Navigator:** Shows the schema `cruise_ship_database_sys` with tables `SHIP_NAME`, `SHIP`, and `SHIP_TYPE`.
- SQL Editor:** Displays the following SQL code:


```

228 •  SELECT * FROM SHIP_NAME;
229
230 •  UPDATE SHIP_NAME SET SHIP_NAME = 'Elissa' WHERE SHIP_ID = 'SH3';
231 •  UPDATE SHIP_NAME SET SHIP_NAME = 'Gloria' WHERE SHIP_ID = 'SH1';
232
233 •  DELETE FROM SHIP_NAME WHERE SHIP_ID = 'SH6';
234 •  SELECT * FROM SHIP_NAME;
235
      
```
- Result Grid:** Shows the `SHIP_NAME` table with rows:

SHIP_ID	SHIP_NAME
SH1	Gloria
SH2	Cristal
SH3	Elissa
SH4	Helena
SH5	Lady Nelson
SH6	Unknown
- Action Output:** Shows the results of the executed queries:

#	Time	Action	Message	Duration / Fetch
61	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH5', 'Lady Nelson')	1 row(s) affected	0.000 sec
62	08:35:28	INSERT INTO SHIP_NAME VALUES ('SH6', 'Unknown')	1 row(s) affected	0.000 sec
63	08:35:33	SELECT * FROM SHIP_NAME LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
64	08:35:59	UPDATE SHIP_NAME SET SHIP_NAME = 'Elissa' WHERE SHIP_ID = 'SH3'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.015 sec
65	08:35:59	UPDATE SHIP_NAME SET SHIP_NAME = 'Gloria' WHERE SHIP_ID = 'SH1'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
66	08:35:59	DELETE FROM SHIP_NAME WHERE SHIP_ID = 'SH6'	1 row(s) affected	0.000 sec
67	08:36:22	SELECT * FROM SHIP_NAME LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Figure 4-28

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

- Navigator:** Shows the schema `cruise_ship_database_sys` with tables `SHIP`, `SHIP_TYPE`, and `SHIP_CREW`.
- SQL Editor:** Displays the following SQL code:


```

243
244 •  SELECT * FROM SHIP;
245
246 •  UPDATE SHIP SET CREW_CAPACITY = 3500, NO_OF_DEPARTMENT = 8 WHERE SHIP_ID = 'SH2';
247 •  UPDATE SHIP SET CREW_CAPACITY = 4500, NO_OF_DEPARTMENT = 15 WHERE SHIP_ID = 'SH3';
248
249 •  DELETE FROM SHIP WHERE SHIP_ID = 'SH5';
250
251 •  SELECT * FROM SHIP;
      
```
- Result Grid:** Shows the `SHIP` table with rows:

SHIP_ID	CREW_CAPACITY	NO_OF_DEPARTMENT
SH1	2000	8
SH2	3500	15
SH3	4500	15
SH4	2000	5
SH5	2000	5
- Action Output:** Shows the results of the executed queries:

#	Time	Action	Message	Duration / Fetch
71	08:37:05	INSERT INTO SHIP VALUES ('SH4', 2000, 5)	1 row(s) affected	0.016 sec
72	08:37:06	INSERT INTO SHIP VALUES ('SH5', 1500, 6)	1 row(s) affected	0.015 sec
73	08:37:22	SELECT * FROM SHIP LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
74	08:37:59	UPDATE SHIP SET CREW_CAPACITY = 3500, NO_OF_DEPARTMENT = 8 WHERE SHIP_ID = 'SH2'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.031 sec
75	08:37:59	UPDATE SHIP SET CREW_CAPACITY = 4500, NO_OF_DEPARTMENT = 15 WHERE SHIP_ID = 'SH3'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
76	08:37:59	DELETE FROM SHIP WHERE SHIP_ID = 'SH5'	1 row(s) affected	0.016 sec
77	08:38:04	SELECT * FROM SHIP LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

Figure 4-29

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Tables

Views

Stored Procedures

Functions

cruise_ship_database_system

Faculty_o_e_23

foe_23

sys

university

Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database

Limit to 1000 rows

261 • INSERT INTO EMPLOYEE_NIC VALUES ('SH1EMP2', '200045462325')

262 •

263 • SELECT * FROM EMPLOYEE_NIC;

264 •

265 • UPDATE EMPLOYEE_NIC SET NIC = '200145891325' WHERE EMPLOYEE_ID = 'SH2EMP2';

266 • UPDATE EMPLOYEE_NIC SET NIC = '199845636123' WHERE EMPLOYEE_ID = 'SH3EMP1';

267 •

268 • DELETE FROM EMPLOYEE_NIC WHERE EMPLOYEE_ID = 'SH2EMP2';

269 •

Result Grid Filter Rows Edit Export/Import Wrap Cell Content

EMPLOYEE_NIC 6

No object selected

Output

Action Output

Time Action Message Duration / Fetch

84 08:38:33 INSERT INTO EMPLOYEE_NIC VALUES ('SH3EMP2', '200045462325') 1 row(s) affected 0.000 sec

85 08:38:33 SELECT * FROM EMPLOYEE_NIC LIMIT 0, 1000 7 row(s) returned 0.000 sec / 0.000 sec

86 08:39:38 UPDATE EMPLOYEE_NIC SET NIC = '200145891325' WHERE EMPLOYEE_ID = 'SH2EMP2' 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.015 sec

87 08:39:38 UPDATE EMPLOYEE_NIC SET NIC = '199845636123' WHERE EMPLOYEE_ID = 'SH3EMP1' 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.015 sec

88 08:39:38 DELETE FROM EMPLOYEE_NIC WHERE EMPLOYEE_ID = 'SH2EMP2' 1 row(s) affected 0.000 sec

89 08:39:44 SELECT * FROM EMPLOYEE_NIC LIMIT 0, 1000 6 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 8:40 AM 4/8/2024

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

Figure 4-30

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Tables

Views

Stored Procedures

Functions

cruise_ship_database_system

Faculty_o_e_23

foe_23

sys

university

Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database

Limit to 1000 rows

287 • SELECT * FROM EMPLOYEE;

288 •

289 • UPDATE EMPLOYEE SET POSITION = 'Chief Engineer', SUPERVISOR_ID = 'SH1EMP2' WHERE EMPLOYEE_ID = 'SH1EMP1';

290 • UPDATE EMPLOYEE SET SUPERVISOR_ID = 'SH1EMP2' WHERE EMPLOYEE_ID = 'SH2EMP1';

291 • UPDATE EMPLOYEE SET POSITION = 'Seaman', STREET = 'North Diamond' WHERE EMPLOYEE_ID = 'SH2EMP1';

292 • UPDATE EMPLOYEE SET POSITION = 'Chief Cook' WHERE EMPLOYEE_ID = 'SH2EMP2';

293 • UPDATE EMPLOYEE SET POSITION = 'Cook', SUPERVISOR_ID = 'SH3EMP2' WHERE EMPLOYEE_ID = 'SH3EMP1';

294 •

295 • DELETE FROM EMPLOYEE WHERE EMPLOYEE_ID = 'SH1EMP3';

296 •

297 • SELECT * FROM EMPLOYEE;

Result Grid Filter Rows Edit Export/Import Wrap Cell Content

EMPLOYEE 8

No object selected

Output

Action Output

Time Action Message Duration / Fetch

99 08:42:11 UPDATE EMPLOYEE SET POSITION = 'Seaman', STREET = 'North Diamond' WHERE EMPLOYEE_ID = 'SH1EMP3' 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.000 sec

100 08:42:11 UPDATE EMPLOYEE SET POSITION = 'Chief Cook' WHERE EMPLOYEE_ID = 'SH3EMP2' 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.000 sec

101 08:42:11 UPDATE EMPLOYEE SET POSITION = 'Cook', SUPERVISOR_ID = 'SH3EMP2' WHERE EMPLOYEE_ID = 'SH3EMP1' 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.015 sec

102 08:42:11 DELETE FROM EMPLOYEE WHERE EMPLOYEE_ID = 'SH1EMP3' 1 row(s) affected 0.016 sec

103 08:42:11 SELECT * FROM EMPLOYEE LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 8:42 AM 4/8/2024

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

Figure 4-31

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

- Tables
- Views
- Stored Procedures
- Functions

faculty_o_e_23

foe_23

sys

university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database × Limit to 1000 rows

```

307
308 • SELECT * FROM EMPLOYEE_WORKED_SHIP;
309
310 • UPDATE EMPLOYEE_WORKED_SHIP SET PREVIOUS_WORKED_SHIP = 'Sea Lion' WHERE EMPLOYEE_ID = 'SH1EMP1' AND PREVIOUS_WORKED_SHIP = 'Noby Baby';
311 • UPDATE EMPLOYEE_WORKED_SHIP SET PREVIOUS_WORKED_SHIP = 'Unicorn' WHERE EMPLOYEE_ID = 'SH2EMP1' AND PREVIOUS_WORKED_SHIP = 'RawFaith';
312
313 • DELETE FROM EMPLOYEE_WORKED_SHIP WHERE EMPLOYEE_ID = 'SH1EMP2' AND PREVIOUS_WORKED_SHIP = 'Queen Mary 2';
314
315 • SELECT * FROM EMPLOYEE_WORKED_SHIP;

```

Result Grid | Filter Rows | Edit | Export/Import | Wrap Cell Content | Result Grid | Form Editor | Context Help | Snippets

EMPLOYEE_ID	PREVIOUS_WORKED_SHIP
SH1EMP1	Oceanic
SH1EMP1	Sea Lion
SH1EMP2	Pacific Dream
SH2EMP1	Unicorn
SH3EMP1	Rosella
SH3EMP1	RawFaith

Administration Schemas Information No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
108	08:43:33	INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH2EMP1', 'RawFaith')	1 row(s) affected	0.000 sec
109	08:43:33	INSERT INTO EMPLOYEE_WORKED_SHIP VALUES ('SH3EMP1', 'Rosella')	1 row(s) affected	0.000 sec
110	08:43:33	SELECT * FROM EMPLOYEE_WORKED_SHIP LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
111	08:43:59	UPDATE EMPLOYEE_WORKED_SHIP SET PREVIOUS_WORKED_SHIP = 'Sea Lion' WHERE EMPLOYEE_ID = 'SH1EMP1';	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
112	08:43:59	UPDATE EMPLOYEE_WORKED_SHIP SET PREVIOUS_WORKED_SHIP = 'Unicorn' WHERE EMPLOYEE_ID = 'SH2EMP1';	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.031 sec
113	08:43:59	DELETE FROM EMPLOYEE_WORKED_SHIP WHERE EMPLOYEE_ID = 'SH1EMP2' AND PREVIOUS_WORKED_SHIP = 'Queen Mary 2';	1 row(s) affected	0.015 sec
114	08:43:59	SELECT * FROM EMPLOYEE_WORKED_SHIP LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 8:44 AM 4/8/2024

Figure 4-32

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

- Tables
- Views
- Stored Procedures
- Functions

faculty_o_e_23

foe_23

sys

university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database × Limit to 1000 rows

```

326 • SELECT * FROM DEPARTMENT;
327
328 • UPDATE DEPARTMENT SET DEPARTMENT_HEAD = 'SH1EMP1' WHERE DEPARTMENT_ID = 'SH1DP1';
329 • UPDATE DEPARTMENT SET DEPARTMENT_NAME = 'housekeeping Department', NO_OF_WORKERS = 150 WHERE DEPARTMENT_ID = 'SH2DP1';
330
331 • DELETE FROM DEPARTMENT WHERE DEPARTMENT_ID = 'SH1DP4';
332
333 • SELECT * FROM DEPARTMENT;

```

Result Grid | Filter Rows | Edit | Export/Import | Wrap Cell Content | Result Grid | Form Editor | Context Help | Snippets

DEPARTMENT_ID	DEPARTMENT_NAME	DEPARTMENT_HEAD	SHIP_ID	NO_OF_WORKERS
SH1DP1	Engineering Department	SH1EMP1	SH1	50
SH1DP2	Deck Department	SH1EMP2	SH1	100
SH2DP1	Housekeeping Department	NULL	SH2	150
SH3DP2	Engineering Department	NULL	SH3	60
SH3DP3	Food and Beverage Department	SH3EMP2	SH3	75
SH3DP3	NULL	NULL	NULL	NULL

Administration Schemas Information No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
119	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH3DP2', 'Engineering Department', NULL, 'SH3', 60)	1 row(s) affected	0.000 sec
120	08:44:57	INSERT INTO DEPARTMENT VALUES ('SH1DP4', 'Security Department', NULL, 'SH1', 50)	1 row(s) affected	0.015 sec
121	08:44:57	SELECT * FROM DEPARTMENT LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
122	08:45:37	UPDATE DEPARTMENT SET DEPARTMENT_HEAD = 'SH1EMP1' WHERE DEPARTMENT_ID = 'SH1DP1';	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.031 sec
123	08:45:37	UPDATE DEPARTMENT SET DEPARTMENT_NAME = 'Housekeeping Department', NO_OF_WORKERS = 150 WHERE DEPARTMENT_ID = 'SH2DP1';	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.015 sec
124	08:45:37	DELETE FROM DEPARTMENT WHERE DEPARTMENT_ID = 'SH1DP4';	1 row(s) affected	0.000 sec
125	08:45:37	SELECT * FROM DEPARTMENT LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 8:46 AM 4/8/2024

Figure 4-33

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

Filter objects

Tables Views Stored Procedures Functions

cruise_ship_database_system

Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database ×

Limit to 1000 rows

348 • INSERT INTO PASSENGER_PASSPORT VALUES ('SH1PS7', '1808230013005');

349

350 • SELECT * FROM PASSENGER_PASSPORT;

351

352 • UPDATE PASSENGER_PASSPORT SET PASSPORT = '188952634561' WHERE PASSENGER_ID = 'SH1PS2';

353 • UPDATE PASSENGER_PASSPORT SET PASSPORT = '188822634561' WHERE PASSENGER_ID = 'SH1PS7';

354

355 • DELETE FROM PASSENGER_PASSPORT WHERE PASSENGER_ID = 'SH1PS7';

356

357 • SELECT * FROM PASSENGER_PASSPORT;

358

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

PASSENGER_ID PASSPORT

SH1PS1	199720013005
SH1PS2	188952634561
SH1PS3	89456013005
SH1PS2	152100013005
SH1PS5	178320013005
SH1PS8	200020013005

No object selected

PASSenger_Passport 14 ×

Action Output

#	Time	Action	Message	Duration / Fetch
139	09:51:26	UPDATE PASSENGER_PASSPORT SET PASSPORT = '188822634561' WHERE PASSENGER_ID = 'SH1PS7'	1 row(s) affected. 1 Changed. 1 Warnings: 0	0.031 sec
140	09:51:26	DELETE FROM PASSENGER_PASSPORT WHERE PASSENGER_ID = 'SH1PS7'	1 row(s) affected	0.000 sec
141	09:51:26	SELECT * FROM PASSENGER_PASSPORT LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search Morning Brief 9:51 AM 4/8/2024

Figure 4-34

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

Filter objects

Tables Views Stored Procedures Functions

cruise_ship_database_system

Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database ×

Limit to 1000 rows

369

370 • UPDATE PASSENGER SET FIRST_NAME = 'Hardik', LAST_NAME = 'Pandya' WHERE PASSENGER_ID = 'SH1PS2';

371 • UPDATE PASSENGER SET COUNTRY = 'Australia' WHERE PASSENGER_ID = 'SH1PS2';

372

373 • DELETE FROM PASSENGER WHERE PASSENGER_ID = 'SH1PS3';

374

375 • SELECT * FROM PASSENGER;

376

Result Grid Filter Rows Edits Export/Import Wrap Cell Content

PASSENGER_ID FIRST_NAME LAST_NAME COUNTRY PROVINCE CITY STREET TRAVELLING_SHIP

SH1PS1	Vidya	Supun	Sri Lanka	Central	Digana	LightLane	sh1
SH1PS2	Hardik	Pandya	India	Karnataka	Bengaluru	22, Mahatma Gandhi Main Road	sh1
SH1PS2	Michel	Stone	Australia	Southern Savonia	Lounaskatu 39	sh3	
SH1PS5	David	Warner	Australia	Queensland	Jondaryan	19 Creek Street	sh3
SH1PS8	Tin	Southee	South Africa	Queensland	Uhana	19 Greek Street	sh3

No object selected

PASSENGER 16 ×

Action Output

#	Time	Action	Message	Duration / Fetch
145	09:52:17	INSERT INTO PASSENGER VALUES ('SH1PS2', 'Michel', 'Stone', 'Finland', 'Southern Savonia', 'Savonlinna', 'U...')	1 row(s) affected	0.016 sec
146	09:52:17	INSERT INTO PASSENGER VALUES ('SH1PS2', 'Tin', 'Southee', 'South Africa', 'Queensland', 'Uhana', '19 Cre...')	1 row(s) affected	0.000 sec
147	09:52:17	INSERT INTO PASSENGER VALUES ('SH1PS5', 'David', 'Warner', 'Australia', 'Queensland', 'Jondaryan', '19 Cr...')	1 row(s) affected	0.000 sec
148	09:52:17	SELECT * FROM PASSENGER LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
149	09:52:17	UPDATE PASSENGER SET FIRST_NAME = 'Hardik', LAST_NAME = 'Pandya' WHERE PASSENGER_ID = 'SH1PS2';	1 row(s) affected. Rows matched: 1 Changed: 1 Warnings: 0	0.031 sec
150	09:52:17	UPDATE PASSENGER SET COUNTRY = 'Australia' WHERE PASSENGER_ID = 'SH1PS2';	1 row(s) affected. Rows matched: 1 Changed: 1 Warnings: 0	0.000 sec
151	09:52:17	DELETE FROM PASSENGER WHERE PASSENGER_ID = 'SH1PS3';	1 row(s) affected	0.000 sec
152	09:52:17	SELECT * FROM PASSENGER LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search Morning Brief 9:53 AM 4/8/2024

Figure 4-35

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS Filter objects

- cruise_ship_database_system
- Tables
- Views
- Stored Procedures
- Functions
- Faculty_o_e_23
- foe_23
- sys
- university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database ×

387

```

388 • UPDATE PASSENGER_CONTACT SET CONTACT_NUMBER = '0751246987' WHERE PASSENGER_ID = 'SH1P51' AND CONTACT_NUMBER = '0766258105';
389 • UPDATE PASSENGER_CONTACT SET CONTACT_NUMBER = '0751245687' WHERE PASSENGER_ID = 'SH3P52' AND CONTACT_NUMBER = '0704558105';
390
391 • DELETE FROM PASSENGER_CONTACT WHERE PASSENGER_ID = 'SH3P52' AND CONTACT_NUMBER = '0766782105';
392
393 • SELECT * FROM PASSENGER_CONTACT;
394
395 -- PASSENGER_MEDICAL_HISTORY
396
397

```

Result Grid | Filter Rows | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Field Types | Context Help | Snippets

PASSENGER_CONTACT_18 ×

PASSENGER_ID	CONTACT_NUMBER
SH1P51	0716258105
SH1P51	0751246987
SH1P52	0756285105
SH3P52	0751245687
SH3P55	0766250005
NULL	NULL

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
160	09:54:40	UPDATE PASSENGER_CONTACT SET CONTACT_NUMBER = '0751246987' WHERE PASSENGER_ID = 'S...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.031 sec
161	09:54:40	UPDATE PASSENGER_CONTACT SET CONTACT_NUMBER = '0751245687' WHERE PASSENGER_ID = 'S...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
162	09:54:40	DELETE FROM PASSENGER_CONTACT WHERE PASSENGER_ID = 'SH3P52' AND CONTACT_NUMBER = '0766782105';	1 row(s) affected	0.016 sec
163	09:54:40	SELECT * FROM PASSENGER_CONTACT LIMIT 0..1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

954 AM 4/8/2024

Figure 4-36

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS Filter objects

- cruise_ship_database_system
- Tables
- Views
- Stored Procedures
- Functions
- Faculty_o_e_23
- foe_23
- sys
- university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Database ×

485 • SELECT * FROM PASSENGER_MEDICAL_HISTORY;

486

```

487 • UPDATE PASSENGER_MEDICAL_HISTORY SET MEDICAL_CONDITION = 'Skin Conditions' WHERE PASSENGER = 'SH1P51' AND MEDICATED_DATE = '2023-12-08';
488 • UPDATE PASSENGER_MEDICAL_HISTORY SET MEDICATION = 'Antiseptic solutions or creams for wound cleaning' WHERE PASSENGER = 'SH3P55' AND MEDICATED_DA
489
490 • DELETE FROM PASSENGER_MEDICAL_HISTORY WHERE PASSENGER = 'SH3P52' AND MEDICATED_DATE = '2023-12-20';
491
492 • SELECT * FROM PASSENGER_MEDICAL_HISTORY;
493

```

PASSENGER_MEDICAL_HISTORY_18 ×

PASSENGER	MEDICATED_DATE	MEDICAL_CONDITION	MEDICATION
SH1P51	2023-12-05	Depression	Anxiolytics or sedatives
SH1P51	2023-12-08	Skin Conditions	Dimehydinate (Dramamine),Medizine (Bonine,...)
SH1P52	2023-12-25	Dehydration	Oral rehydration solutions (e.g., Pedialyte)
SH3P52	2023-12-18	Gastrointestinal Issues	Antidiarrheal medications (e.g., loperamide/Imo...
SH3P55	2023-12-30	Injuries	Antiseptic solutions or creams for wound cleaning

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
176	09:56:37	UPDATE PASSENGER_MEDICAL_HISTORY SET MEDICATION = 'Antiseptic solutions or creams for wound cl...	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.000 sec
177	09:56:37	DELETE FROM PASSENGER_MEDICAL_HISTORY WHERE PASSENGER = 'SH3P52' AND MEDICATED_DATE = '2023-12-20';	0 row(s) affected	0.000 sec
178	09:56:37	SELECT * FROM PASSENGER_MEDICAL_HISTORY LIMIT 0..1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search

84°F Mostly sunny 956 AM 4/8/2024

Figure 4-37

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas Administration Information

No object selected

Result Grid | Filter Rows | Edit: Export/Import: Wrap Cell Content: | SQLAdditions | SQL Additions | Jump to | Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

SCHEMAS

cruise_ship_database_system

Tables Views Stored Procedures Functions

Faculty_o_e_23 foe_23 sys university

424 ● UPDATE CABIN SET CABIN_TYPE = 'Dinning Room 1' WHERE DECK = 3 AND CABIN_NUMBER = 31 AND SHIP = 'SH3';
 425 ● UPDATE CABIN SET DECK = 10, CABIN_NUMBER = 1, CABIN_TYPE = "Captain's Quarters" WHERE DECK = 2 AND CABIN_NUMBER = 24 AND SHIP = 'SH1';
 426 ● DELETE FROM CABIN WHERE DECK = 4 AND CABIN_NUMBER = 23 AND SHIP = 'SH1';
 427
 428 ● SELECT * FROM CABIN;

Result Grid | Filter Rows | Edit: Export/Import: Wrap Cell Content: | SQLAdditions | SQL Additions | Jump to | Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

DECK CABIN_NUMBER CABIN_TYPE SHIP

2	35	Dinning Room 2	SH1
3	15	Family Cabin 1	SH3
3	31	Dinning Room 1	SH3
5	30	Balcony Cabin	SH2
10	1	Captain's Quarters	SH1
NULL	NULL	NULL	NULL

Result Grid | Form Editor | Field Types | Context Help | Snippets | CABIN 23 × | Output | Action Output | Time Action Message Duration / Fetch

185 09:57:15 SELECT * FROM CABIN LIMIT 0, 1000 6 row(s) returned 0.000 sec / 0.000 sec
 186 09:58:04 UPDATE CABIN SET CABIN_TYPE = 'Dinning Room 1' WHERE DECK = 3 AND CABIN_NUMBER = 31 AND SHIP = 'SH3'; 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.015 sec
 187 09:58:04 UPDATE CABIN SET DECK = 10, CABIN_NUMBER = 1, CABIN_TYPE = "Captain's Quarters" WHERE DECK = 2 AND CABIN_NUMBER = 24 AND SHIP = 'SH1'; 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.015 sec
 188 09:58:04 DELETE FROM CABIN WHERE DECK = 4 AND CABIN_NUMBER = 23 AND SHIP = 'SH1'; 1 row(s) affected 0.015 sec
 189 09:58:04 SELECT * FROM CABIN LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 9:58 AM 4/8/2024

Figure 4-38

MySQL Workbench

Local instance MySQL80 ×

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas Administration Information

No object selected

Result Grid | Filter Rows | Edit: Export/Import: Wrap Cell Content: | SQLAdditions | SQL Additions | Jump to | Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

SCHEMAS

cruise_ship_database_system

Tables Views Stored Procedures Functions

Faculty_o_e_23 foe_23 sys university

442 ● UPDATE CABIN_INCHARGE SET INCHARGE = 'SH3EMP1' WHERE DECK = 3 AND CABIN_NUMBER = 31 AND INCHARGE = 'SH2EMP1';
 443 ● UPDATE CABIN_INCHARGE SET INCHARGE = 'SH2EMP1' WHERE DECK = 5 AND CABIN_NUMBER = 30 AND INCHARGE = 'SH3EMP1';
 444 ● DELETE FROM CABIN_INCHARGE WHERE DECK = 5 AND CABIN_NUMBER = 30 AND INCHARGE = 'SH2EMP1';
 445
 446 ● SELECT * FROM CABIN_INCHARGE;

Result Grid | Filter Rows | Edit: Export/Import: Wrap Cell Content: | SQLAdditions | SQL Additions | Jump to | Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

DECK CABIN_NUMBER INCHARGE SHIP

2	35	SH1EMP1	SH1
2	35	SH1EMP2	SH1
10	1	SH1EMP2	SH1
3	31	SH2EMP1	SH3
3	31	SH2EMP2	SH3
NULL	NULL	NULL	NULL

Result Grid | Form Editor | Field Types | Context Help | Snippets | CABIN_INCHARGE 25 × | Output | Action Output | Time Action Message Duration / Fetch

200 09:59:13 SELECT * FROM CABIN_INCHARGE LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
 201 09:59:30 UPDATE CABIN_INCHARGE SET INCHARGE = 'SH3EMP1' WHERE DECK = 3 AND CABIN_NUMBER = 31 AND INCHARGE = 'SH2EMP1'; 0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0 0.000 sec
 202 09:59:30 UPDATE CABIN_INCHARGE SET INCHARGE = 'SH2EMP1' WHERE DECK = 5 AND CABIN_NUMBER = 30 AND INCHARGE = 'SH3EMP1'; 0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0 0.015 sec
 203 09:59:30 DELETE FROM CABIN_INCHARGE WHERE DECK = 5 AND CABIN_NUMBER = 30 AND INCHARGE = 'SH2EMP1'; 0 row(s) affected 0.000 sec
 204 09:59:30 SELECT * FROM CABIN_INCHARGE LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 9:59 AM 4/8/2024

Figure 4-39

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

- Tables
- Views
- Stored Procedures
- Functions

faculty_o_e_23

foe_23

sys

university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database.x

Limit to 1000 rows

458

```

459 • SELECT * FROM CABIN_SERVICE_REQUEST;
460
461 • UPDATE CABIN_SERVICE_REQUEST SET SERVICE_REQUEST_TYPE = 'Need a Chair' WHERE DECK = 10 AND CABIN_NUMBER = 1 AND SHIP = 'SH1';
462 • UPDATE CABIN_SERVICE_REQUEST SET SERVICE_REQUEST_TYPE = 'Wall broken' WHERE DECK = 5 AND CABIN_NUMBER = 30 AND SHIP = 'SH2';
463
464 • DELETE FROM CABIN_SERVICE_REQUEST WHERE DECK = 5 AND CABIN_NUMBER = 30 AND SHIP = 'SH2';
465
466 • SELECT * FROM CABIN_SERVICE_REQUEST;
467

```

Result Grid

DECK	CABIN_NUMBER	REQUEST_DATE	SERVICE_REQUEST_TYPE	SHIP
2	35	2023-12-10	Table Broken	SH1
2	35	2023-12-25	Chair Broken	SH1
3	15	2023-12-05	Need to fix the cupboard	SH3
3	31	2023-12-30	Need a Chair	SH3
10	1	2023-12-22	Need a Chair	SH1
•	•	•	•	•
•	•	•	•	•

Administration Schemas Information

No object selected

CABIN_SERVICE_REQUEST_28

Action Output

#	Time	Action	Message	Duration / Fetch
211	10:00:06	SELECT * FROM CABIN_SERVICE_REQUEST LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
212	10:00:37	UPDATE CABIN_SERVICE_REQUEST SET SERVICE_REQUEST_TYPE = 'Need a Chair' WHERE DECK = 1...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.031 sec
213	10:00:37	UPDATE CABIN_SERVICE_REQUEST SET SERVICE_REQUEST_TYPE = 'Wall broken' WHERE DECK = 5 A...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
214	10:00:37	DELETE FROM CABIN_SERVICE_REQUEST WHERE DECK = 5 AND CABIN_NUMBER = 30 AND SHIP = 'S...	1 row(s) affected	0.016 sec
215	10:00:37	SELECT * FROM CABIN_SERVICE_REQUEST LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

84°F Mostly sunny 10:00 AM 4/8/2024

Figure 4-40

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Schemas

Filter objects

cruise_ship_database_system

- Tables
- Views
- Stored Procedures
- Functions

faculty_o_e_23

foe_23

sys

university

FoE Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database.x

Limit to 1000 rows

477 • SELECT * FROM CRUISE_DINNING_RESERVATION;

```

478
479 • UPDATE CRUISE_DINNING_RESERVATION SET NUMBER_OF_GUESTS = 8 WHERE DECK = 2 AND CABIN_NUMBER = 35 AND SHIP = 'SH1' AND RESERVATION_DATE = '2023-12-11';
480 • UPDATE CRUISE_DINNING_RESERVATION SET NUMBER_OF_GUESTS = 15 WHERE DECK = 3 AND CABIN_NUMBER = 31 AND SHIP = 'SH3' AND RESERVATION_DATE = '2023-12-11';
481
482 • DELETE FROM CRUISE_DINNING_RESERVATION WHERE DECK = 2 AND CABIN_NUMBER = 35 AND SHIP = 'SH1' AND RESERVATION_DATE = '2023-12-11' AND RESERVATION_DATE > '2023-12-11';
483
484 • SELECT * FROM CRUISE_DINNING_RESERVATION;
485
486 -- PORT TABLE
487

```

Result Grid

DECK	CABIN_NUMBER	RESERVATION_DATE	RESERVATION_TIME	NUMBER_OF_GUESTS	SHIP
2	35	2023-12-12	17:00:00	8	SH1
2	35	2023-12-13	20:30:00	5	SH1
3	31	2023-12-02	11:30:00	15	SH3
3	31	2023-12-11	15:30:00	10	SH3
3	31	2023-12-25	19:00:00	12	SH3
•	•	•	•	•	•
•	•	•	•	•	•

CRUISE_DINNING_RESERVATION

Action Output

#	Time	Action	Message	Duration / Fetch
222	10:01:41	SELECT * FROM CRUISE_DINNING_RESERVATION LIMIT 0, 1000	6 row(s) returned	0.000 sec / 0.000 sec
223	10:04:01	UPDATE CRUISE_DINNING_RESERVATION SET NUMBER_OF_GUESTS = 8 WHERE DECK = 2 AND CABIN_NUMBER = 35 AND SHIP = 'SH1' AND RESERVATION_DATE = '2023-12-11';	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
224	10:04:01	UPDATE CRUISE_DINNING_RESERVATION SET NUMBER_OF_GUESTS = 15 WHERE DECK = 3 AND CABIN_NUMBER = 31 AND SHIP = 'SH3' AND RESERVATION_DATE = '2023-12-11';	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
225	10:04:01	DELETE FROM CRUISE_DINNING_RESERVATION WHERE DECK = 2 AND CABIN_NUMBER = 35 AND SHIP = 'SH1' AND RESERVATION_DATE = '2023-12-11' AND RESERVATION_DATE > '2023-12-11';	1 row(s) affected	0.000 sec
226	10:04:01	SELECT * FROM CRUISE_DINNING_RESERVATION LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

84°F Mostly sunny 10:04 AM 4/8/2024

Figure 4-41

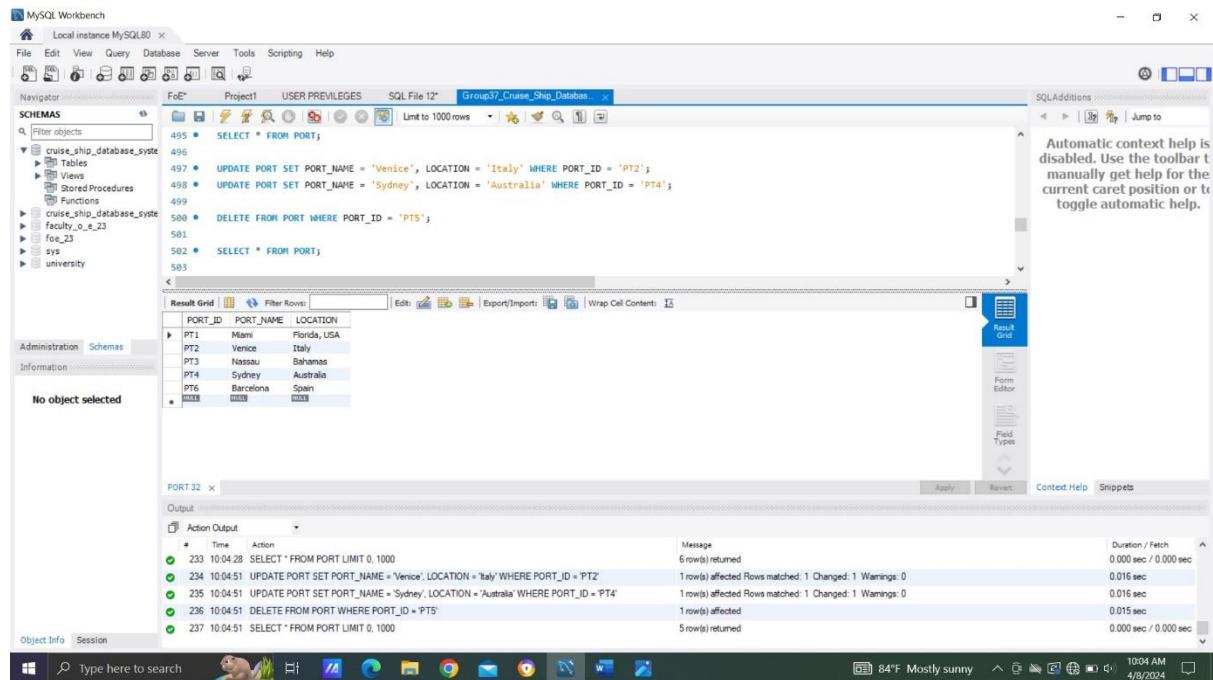


Figure 4-42

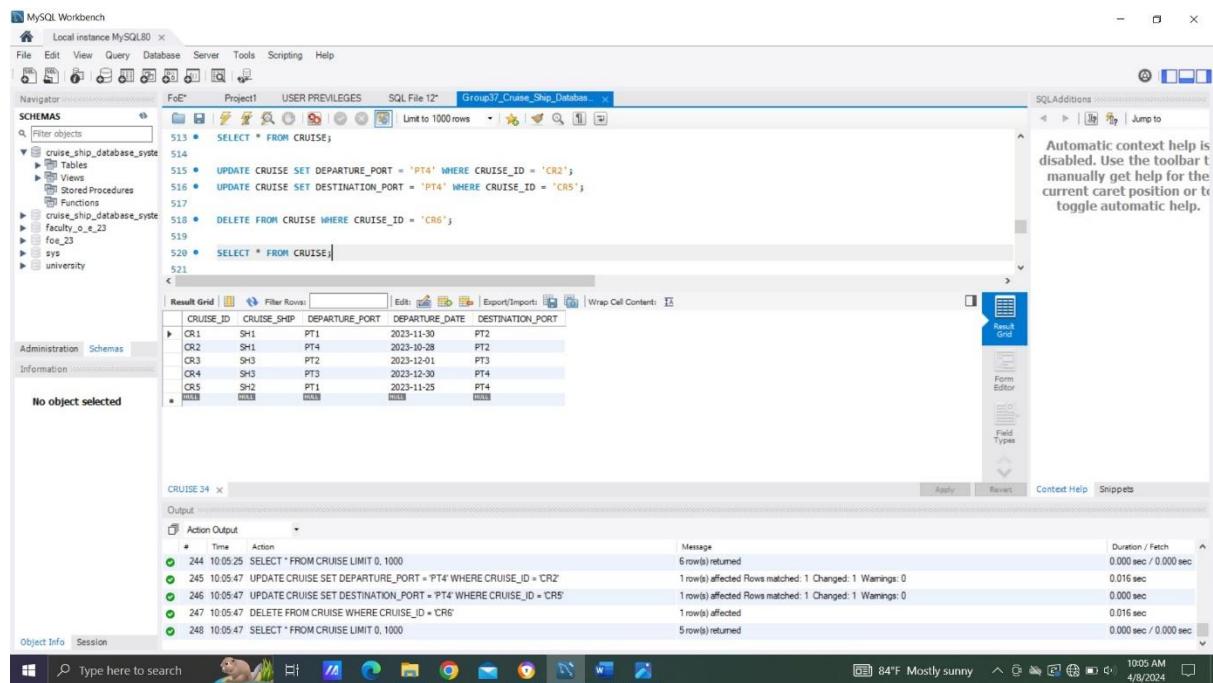
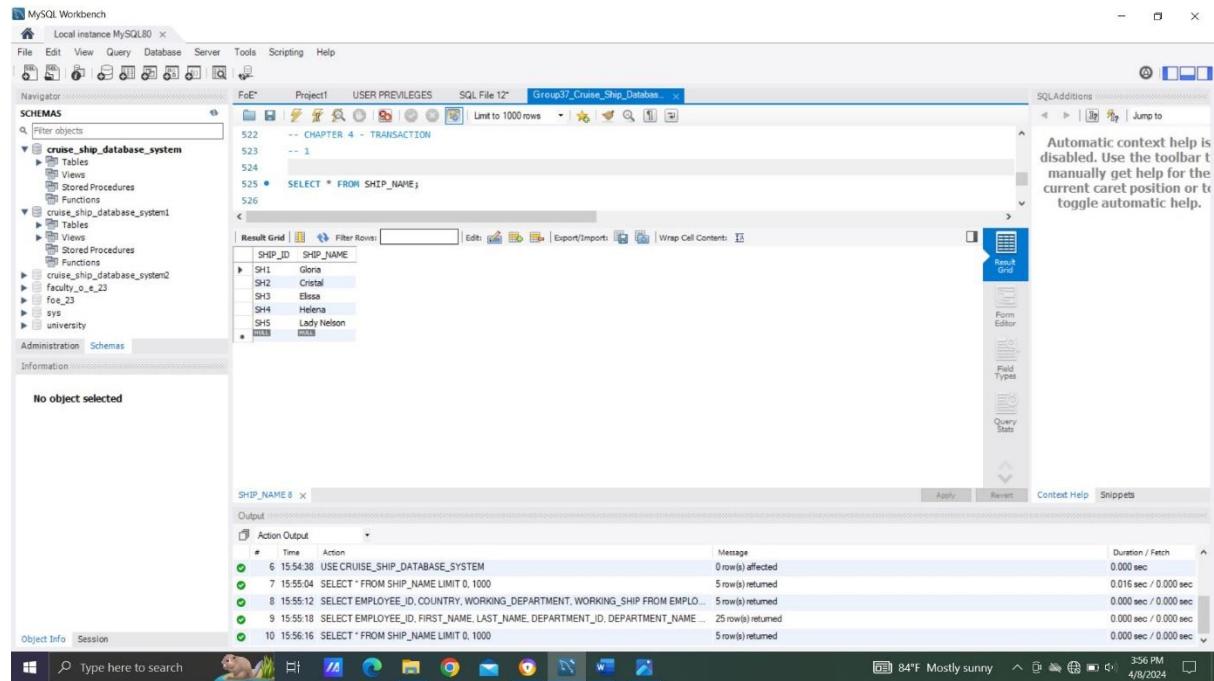


Figure 4-43

5. Transactions

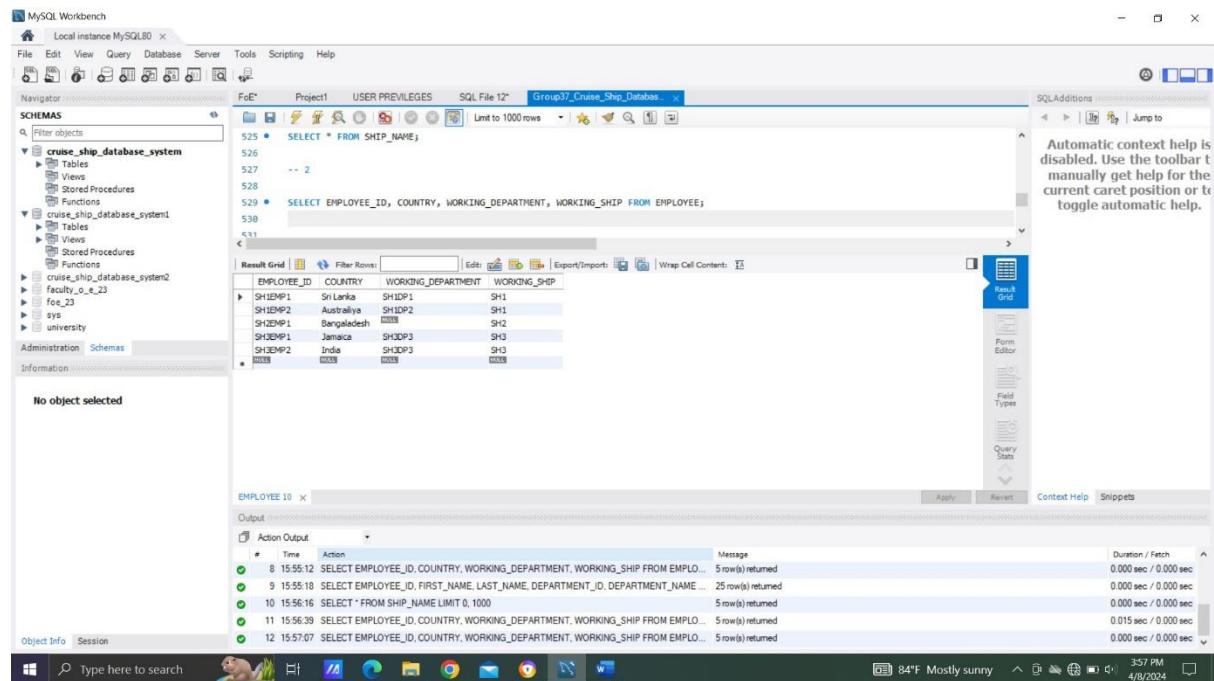
5.1. Simple Queries



The screenshot shows the MySQL Workbench interface. In the top-left pane, the 'Navigator' shows the database schema with several databases listed under 'cruise_ship_database_system'. The 'Result Grid' pane displays the results of a query: 'SELECT * FROM SHIP_NAME;'. The results show five rows of ship names: Gloria, Cristal, Elsa, Helena, and Lady Nelson. The bottom-right pane shows the 'Object Info' and 'Session' tabs, and the bottom status bar indicates the date and time.

SHIP_ID	SHIP_NAME
SH1	Gloria
SH2	Cristal
SH3	Elsa
SH4	Helena
SH5	Lady Nelson

Figure 5-1



This screenshot is similar to Figure 5-1, showing the MySQL Workbench interface. It displays the results of a query: 'SELECT * FROM SHIP_NAME;'. The results show five rows of ship names: Sri Lanka, Australia, Bangladesh, Jamaica, and India. The bottom-right pane shows the 'Object Info' and 'Session' tabs, and the bottom status bar indicates the date and time.

EMPLOYEE_ID	COUNTRY	WORKING_DEPARTMENT	WORKING_SHIP
SH1EMP1	Sri Lanka	SH1DP1	SH1
SH1EMP2	Australia	SH1DP2	SH1
SH2EMP1	Bangladesh	SH2DP1	SH2
SH3EMP1	Jamaica	SH3DP3	SH3
SH3EMP2	India	SH3DP3	SH3

Figure 5-2

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

- Navigator:** Shows the schema `cruise_ship_database_system` containing tables, views, stored procedures, and functions.
- SQL Editor:** Contains the following SQL code:


```

531 -- 3
532
533 *   SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, DEPARTMENT_ID, DEPARTMENT_NAME FROM EMPLOYEE CROSS JOIN DEPARTMENT;
534
      
```
- Result Grid:** Displays the results of the query, showing columns: EMPLOYEE_ID, FIRST_NAME, LAST_NAME, DEPARTMENT_ID, and DEPARTMENT_NAME. The data includes rows for employees like Virat, Kris, and Hirusha across various departments like SH1OP1, SH1OP2, SH2OP1, etc.
- Toolbar:** Includes icons for Run, Stop, Refresh, and Save.
- Right Panel:** Shows context help information: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."
- Bottom Status Bar:** Shows system status: 84°F Mostly sunny, 358 PM, 4/8/2024.

Figure 5-3

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

- Navigator:** Shows the schema `cruise_ship_database_system` containing tables, views, stored procedures, and functions.
- SQL Editor:** Contains the following SQL code:


```

534 *   SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, DEPARTMENT_ID, DEPARTMENT_NAME FROM EMPLOYEE CROSS JOIN DEPARTMENT;
535
536 -- 4
537
538 *   CREATE VIEW CABIN_SERVICE AS SELECT * FROM CABIN NATURAL JOIN CABIN_SERVICE_REQUEST;
539 *   SELECT * FROM CABIN_SERVICE;
540
      
```
- Result Grid:** Displays the results of the query, showing columns: DECK, CABIN_NUMBER, SHIP, CABIN_TYPE, REQUEST_DATE, and SERVICE_REQUEST_TYPE. The data includes rows for cabin requests like Dining Room 2, Family Cabin 1, etc.
- Output Tab:** Shows the execution history and message log, including actions like SELECT statements and the creation of the CABIN_SERVICE view.
- Toolbar:** Includes icons for Run, Stop, Refresh, and Save.
- Right Panel:** Shows context help information: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."
- Bottom Status Bar:** Shows system status: 84°F Mostly sunny, 358 PM, 4/8/2024.

Figure 5-4

The screenshot shows the MySQL Workbench interface with a query editor window titled "FoE" containing the following SQL code:

```

541 -- 5
542 *   SELECT PORT_ID, PORT_NAME, CRUISE_ID, DEPARTURE_PORT, DESTINATION_PORT
      FROM PORT AS P NATURAL JOIN CRUISE AS C;
544

```

The Result Grid displays the following data:

PORT_ID	PORT_NAME	CRUISE_ID	DEPARTURE_PORT	DESTINATION_PORT
PT1	Miami	CR1	PT1	PT2
PT2	Venice	CR1	PT1	PT2
PT3	Nassau	CR1	PT1	PT2
PT4	Sydney	CR1	PT1	PT2
PT5	Barcelona	CR2	PT2	PT2
PT6	Barcelona	CR3	PT2	PT3
PT7	Sydney	CR3	PT2	PT3
PT8	Nassau	CR3	PT2	PT3
PT9	Venice	CR3	PT2	PT3
PT10	Miami	CR3	PT2	PT3
PT11	Barcelona	CR4	PT3	PT4
PT12	Sydney	CR4	PT3	PT4
PT13	Nassau	CR4	PT3	PT4
PT14	Venice	CR4	PT3	PT4
PT15	Miami	CR4	PT3	PT4
PT16	Barcelona	CR5	PT1	PT4
PT17	Sydney	CR5	PT1	PT4
PT18	Nassau	CR5	PT1	PT4

The Output pane shows the following log entries:

```

# Time Action Message Duration / Fetch
1 15:56:18 SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, DEPARTMENT_ID, DEPARTMENT_NAME ... 25 row(s) returned 0.000 sec / 0.000 sec
2 15:56:16 SELECT * FROM SHIP_NAME LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
3 15:56:39 SELECT EMPLOYEE_ID, COUNTRY, WORKING_DEPARTMENT, WORKING_SHIP FROM EMPLO... 5 row(s) returned 0.015 sec / 0.000 sec
4 15:57:07 SELECT EMPLOYEE_ID, COUNTRY, WORKING_DEPARTMENT, WORKING_SHIP FROM EMPLO... 5 row(s) returned 0.000 sec / 0.000 sec
5 15:57:17 SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, DEPARTMENT_ID, DEPARTMENT_NAME FROM EMPLOYEE CROSS JOIN DEPARTMENT ... 5 row(s) returned 0.000 sec / 0.000 sec
6 15:58:00 CREATE VIEW CABIN_SERVICE AS SELECT * FROM CABIN_SERVICE LIMIT 0, 1000 0 rows affected 0.032 sec
7 15:58:02 SELECT * FROM CABIN_SERVICE LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
8 15:58:12 SELECT PORT_ID, PORT_NAME, CRUISE_ID, DEPARTURE_PORT, DESTINATION_PORT FROM ... 25 row(s) returned 0.000 sec / 0.000 sec
9 15:58:02 SELECT EMPLOYEE_ID AS EMPL_ID, COUNT(EMPLOYEE_ID) AS EXPERIENCE FROM EMPLOYEE_WORKED_SHIP GROUP BY EMPLOYEE_ID ORDER BY EXPERIENCE DESC 4 row(s) returned 0.000 sec / 0.000 sec

```

Figure 5-5

The screenshot shows the MySQL Workbench interface with a query editor window titled "FoE" containing the following SQL code:

```

543 *   SELECT PORT_ID, PORT_NAME, CRUISE_ID, DEPARTURE_PORT, DESTINATION_PORT
      FROM PORT AS P NATURAL JOIN CRUISE AS C;
544
545 -- 6
546
547 *   SELECT EMPLOYEE_ID AS EMPL_ID, COUNT(EMPLOYEE_ID) AS EXPERIENCE FROM EMPLOYEE_WORKED_SHIP GROUP BY EMPLOYEE_ID ORDER BY EXPERIENCE DESC;
548

```

The Result Grid displays the following data:

EMPL_ID	EXPERIENCE
SH1EMP1	2
SH1EMP2	1
SH2EMP1	1
SH3EMP1	1

The Output pane shows the following log entries:

```

# Time Action Message Duration / Fetch
1 15:56:18 SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, DEPARTMENT_ID, DEPARTMENT_NAME ... 25 row(s) returned 0.000 sec / 0.000 sec
2 15:56:16 SELECT * FROM SHIP_NAME LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
3 15:56:39 SELECT EMPLOYEE_ID, COUNTRY, WORKING_DEPARTMENT, WORKING_SHIP FROM EMPLO... 5 row(s) returned 0.015 sec / 0.000 sec
4 15:57:07 SELECT EMPLOYEE_ID, COUNTRY, WORKING_DEPARTMENT, WORKING_SHIP FROM EMPLO... 5 row(s) returned 0.000 sec / 0.000 sec
5 15:57:17 SELECT EMPLOYEE_ID, FIRST_NAME, LAST_NAME, DEPARTMENT_ID, DEPARTMENT_NAME FROM EMPLOYEE CROSS JOIN DEPARTMENT ... 5 row(s) returned 0.000 sec / 0.000 sec
6 15:58:00 CREATE VIEW CABIN_SERVICE AS SELECT * FROM CABIN_SERVICE LIMIT 0, 1000 0 rows affected 0.032 sec
7 15:58:02 SELECT * FROM CABIN_SERVICE LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec
8 15:58:12 SELECT PORT_ID, PORT_NAME, CRUISE_ID, DEPARTURE_PORT, DESTINATION_PORT FROM ... 25 row(s) returned 0.000 sec / 0.000 sec
9 15:58:02 SELECT EMPLOYEE_ID AS EMPL_ID, COUNT(EMPLOYEE_ID) AS EXPERIENCE FROM EMPLOYEE_WORKED_SHIP GROUP BY EMPLOYEE_ID ORDER BY EXPERIENCE DESC 4 row(s) returned 0.000 sec / 0.000 sec

```

Figure 5-6

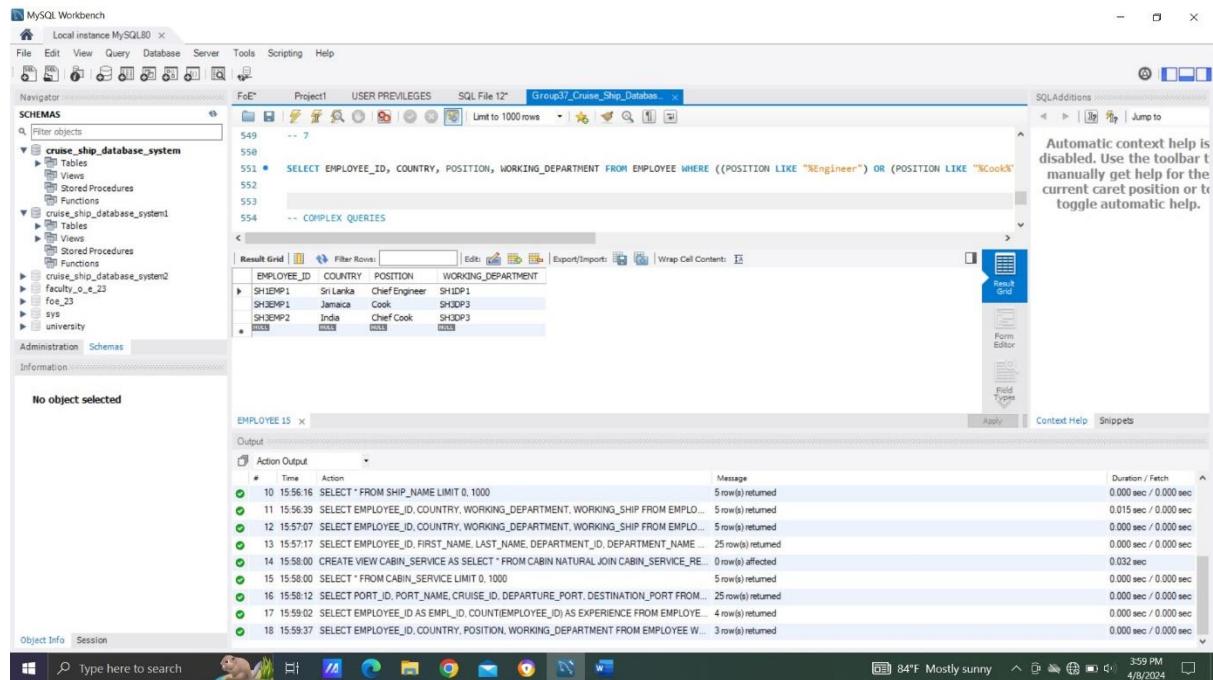


Figure 5-7

5.2. Complex Queries

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

- Navigator:** Shows the schema tree for the "cruise_ship_database_system" database, including tables like "Employees", "Ships", and "CabinServices".
- SQL Editor:** Displays a complex multi-table query involving "Employees", "Ships", and "CabinServices".
- Result Grid:** Shows the output of the query, listing employee IDs, working ships, and countries.
- Action Output:** Shows the execution log with 16 entries, detailing each SQL statement and its duration.

EMPLOYEE_ID	WORKING_SHIP	COUNTRY
SH1EMP1	SH3	Jamaica
SH1EMP2	SH3	India
SH1EMP1	SH1	Sri Lanka
SH1EMP2	SH1	Australia

Figure 5-8

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

- Navigator:** Shows the schema tree for the "cruise_ship_database_system" database.
- SQL Editor:** Displays a complex query involving multiple joins and subqueries.
- Result Grid:** Shows the output of the query, listing employee IDs and working ships.
- Action Output:** Shows the execution log with 19 entries, detailing each SQL statement and its duration.

EMPLOYEE_ID	WORKING_SHIP
SH1EMP1	SH1
SH1EMP2	SH1

Figure 5-9

The screenshot shows the MySQL Workbench interface with a SQL editor window titled "Group37_Create_Ship_Database". The code is a multi-line SQL script for creating tables and constraints. The results pane shows a table named "PASSENGER_ID" with three rows: SH1P52, SH3P52, and SH3P55. Below the results is an "Action Output" log showing the execution history of the queries.

```

567 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE E.WORK...
568
569 -- 3
570
571 • (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP NATURAL JOIN PASSENGER_CONTACT AS PC)
572 □ EXCEPT
573 (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP INNER JOIN PASSENGER_MEDICAL_HISTORY AS PMS ON PP.PASSENGER_ID = PMS.PASSENGER
574
575 -- 4
576
577 <-- SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMENT AS D ON E.EMPLOYEE_ID = D.DEPARTMENT_HEAD
578
579 □ INTERSECT
580
581 -- 5 NATURAL JOIN

```

PASSENGER_ID
SH1P52
SH3P52
SH3P55

Action Output:

- 17 15:59:02 SELECT EMPLOYEE_ID AS EMPL_ID, COUNT(EMPLOYEE_ID) AS EXPERIENCE FROM EMPLOYEE... 4 row(s) returned
- 18 15:59:37 SELECT EMPLOYEE_ID, COUNTRY, POSITION, WORKING_DEPARTMENT FROM EMPLOYEE W... 3 row(s) returned
- 19 16:02:23 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E INNER JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE CI...
- 20 16:02:43 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S O... 2 row(s) returned
- 21 16:03:00 (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP NATURAL JOIN PASSENG... 3 row(s) returned
- 22 16:03:13 (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTME... 2 row(s) returned

Figure 5-10

This screenshot shows the same MySQL Workbench environment as Figure 5-10, but with a different SQL script in the editor. The code includes an additional SELECT statement and a natural join clause. The results pane shows a table with columns FIRST_NAME, LAST_NAME, and CITY, containing data for Senara, Koshala, and Uhana. The action output log shows the execution of the modified queries.

```

573 (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP INNER JOIN PASSENGER_MEDICAL_HISTORY AS PMS ON PP.PASSENGER_ID = PMS.PASSENGER
574
575 -- 4
576
577 • (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMENT AS D ON E.EMPLOYEE_ID = D.DEPARTMENT_HEAD)
578 □ INTERSECT
579 (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE CI...
580
581 -- 5 NATURAL JOIN

```

FIRST_NAME	LAST_NAME	CITY
Senara	Koshala	Uhana
Wasenith	Kodikara	Leydens Hill

Action Output:

- 18 15:59:37 SELECT EMPLOYEE_ID, COUNTRY, POSITION, WORKING_DEPARTMENT FROM EMPLOYEE W... 3 row(s) returned
- 19 16:02:23 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E INNER JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE CI...
- 20 16:02:43 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S O... 2 row(s) returned
- 21 16:03:00 (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP NATURAL JOIN PASSENG... 3 row(s) returned
- 22 16:03:13 (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTME... 2 row(s) returned

Figure 5-11

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- cruise_ship_database_system
 - Tables
 - Views
 - Stored Procedures
 - Functions
- cruise_ship_database_system1
 - Tables
 - Views
 - Stored Procedures
 - Functions
- cruise_ship_database_system2
 - Tables
 - Views
 - Stored Procedures
 - Functions
- faculty_o_e_23
- foe_23
- sys
- university

Administration Schemas

No object selected

FoE* Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Databases

```

579  (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE CI.I.^
580
581  -- 5 NATURAL JOIN
582
583  * CREATE VIEW UV1 AS SELECT S.SHIP_ID, SN.SHIP_NAME FROM SHIP_NAME AS SN NATURAL JOIN SHIP AS S;
584  * SELECT * FROM UV1;
585
586  -- 6 INNER JOIN
587

```

Result Grid | Filter Rows | Export | Wrap Cell Contents |

SHIP_ID	SHIP_NAME
SH1	Gloria
SH2	Cristal
SH3	Eissa
SH4	Helena

UV1 20 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
20	16:02:43	(SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	2 row(s) returned	0.000 sec / 0.000 sec
21	16:03:00	(SELECT PP.PASSENGER_ID FROM PASSENGER_PP AS PP NATURAL JOIN PASSENGER AS P)	3 row(s) returned	0.000 sec / 0.000 sec
22	16:03:13	(SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMENT AS D ON E.DEPARTMENT_ID = D.DEPARTMENT_ID)	2 row(s) returned	0.000 sec / 0.000 sec
23	16:03:44	CREATE VIEW UV1 AS SELECT S.SHIP_ID, SN.SHIP_NAME FROM SHIP_NAME AS SN NATURAL JOIN SHIP AS S;	0 row(s) affected	0.016 sec
24	16:03:44	SELECT * FROM UV1 LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 4:03 PM 4/8/2024

Figure 5-12

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- cruise_ship_database_system
 - Tables
 - Views
 - Stored Procedures
 - Functions
- cruise_ship_database_system1
 - Tables
 - Views
 - Stored Procedures
 - Functions
- cruise_ship_database_system2
 - Tables
 - Views
 - Stored Procedures
 - Functions
- faculty_o_e_23
- foe_23
- sys
- university

Administration Schemas

No object selected

FoE* Project1 USER PRIVILEGES SQL File 12* Group37_Create_Ship_Databases

```

585
586  -- 6 INNER JOIN
587
588  * CREATE VIEW UV2 AS SELECT EN.EMPLOYEE_ID, EN.NIC, D.DEPARTMENT_NAME FROM EMPLOYEE_NIC AS EN INNER JOIN
589  DEPARTMENT AS D ON EN.EMPLOYEE_ID = D.DEPARTMENT_HEAD;
590  * SELECT * FROM UV2;
591
592  -- 7 LEFT OUTER JOIN
593

```

Result Grid | Filter Rows | Export | Wrap Cell Contents |

EMPLOYEE_ID	NIC	DEPARTMENT_NAME
SHIEMP1	200145246325	Engineering Department
SHIEMP2	200045786325	Deck Department
SHIEMP2	200045462325	Food and Beverage Department

UV2 21 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
22	16:03:13	(SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMENT AS D ON E.DEPARTMENT_ID = D.DEPARTMENT_ID)	2 row(s) returned	0.000 sec / 0.000 sec
23	16:03:44	CREATE VIEW UV1 AS SELECT S.SHIP_ID, SN.SHIP_NAME FROM SHIP_NAME AS SN NATURAL JOIN SHIP AS S;	0 row(s) affected	0.016 sec
24	16:03:44	SELECT * FROM UV1 LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec
25	16:03:59	CREATE VIEW UV2 AS SELECT EN.EMPLOYEE_ID, EN.NIC, D.DEPARTMENT_NAME FROM EMPLOYEE_NIC AS EN INNER JOIN DEPARTMENT AS D ON EN.EMPLOYEE_ID = D.DEPARTMENT_HEAD;	0 row(s) affected	0.015 sec
26	16:03:59	SELECT * FROM UV2 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Type here to search 84°F Mostly sunny 4:04 PM 4/8/2024

Figure 5-13

```

591 -- 7 LEFT OUTER JOIN
592
593
594 • CREATE VIEW UV3 AS SELECT * FROM SHIP_NAME AS SN LEFT JOIN CABIN AS C ON SN.SHIP_ID = C.SHIP WHERE SN.SHIP_ID != 'SH1';
595 • SELECT * FROM UV3;
596
597 -- 8 RIGHT OUTER JOIN
598
599 • CREATE VIEW UV4 AS SELECT PC.PASSENGER_ID, PC.CONTACT_NUMBER, PP.PASSPORT FROM PASSENGER_CONTACT AS PC RIGHT JOIN

```

SHIP_ID	SHIP_NAME	DECK	CABIN_NUMBER	CABIN_TYPE	SHIP
SH2	Cristal	5	30	Balcony Cabin	SH2
SH3	Ella	3	31	Dining Room 1	SH3
SH3	Ella	3	15	Family Cabin 1	SH3
SH4	Helena	MALE	MALE	MALE	SH3
SH5	Lady Nelson	MALE	MALE	MALE	SH3

UV3 22 x

Action Output

#	Time	Action	Message	Duration / Fetch
24	16:03:44	SELECT * FROM UV1 LIMIT 0, 1000	4 row(s) returned	0.015 sec / 0.000 sec
25	16:03:59	CREATE VIEW UV2 AS SELECT EN.EMPLOYEE_ID, EN.NIC, D.DEPARTMENT_NAME FROM EMPLOYEE AS EN, DEPARTMENT AS D WHERE EN.DEPARTMENT_ID = D.DEPARTMENT_ID	0 row(s) affected	0.015 sec
26	16:03:59	SELECT * FROM UV2 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
27	16:04:08	CREATE VIEW UV3 AS SELECT * FROM SHIP_NAME AS SN LEFT JOIN CABIN AS C ON SN.SHIP_ID = C.SHIP WHERE SN.SHIP_ID != 'SH1'	0 row(s) affected	0.015 sec
28	16:04:08	SELECT * FROM UV3 LIMIT 0, 1000	5 row(s) returned	0.016 sec / 0.000 sec

Figure 5-14

```

597 -- 8 RIGHT OUTER JOIN
598
599 • CREATE VIEW UVA AS SELECT PC.PASSENGER_ID, PC.CONTACT_NUMBER, PP.PASSPORT FROM PASSENGER_CONTACT AS PC RIGHT JOIN
600 PASSENGER_PASSPORT AS PP ON PC.PASSENGER_ID = PP.PASSENGER_ID;
601 • SELECT * FROM UVA;
602
603 -- 9 FULL OUTER JOIN
604
605 • CREATE VIEW UVB AS SELECT EN.EMPLOYEE_ID, EN.NIC, CI.INCHARGE FROM EMPLOYEE NIC AS EN LEFT OUTER JOIN CABIN INCHARGE AS CI ON EN.EMPLOYEE_ID = CI.CABIN_ID

```

PASSENGER_ID	CONTACT_NUMBER	PASSPORT
SH1P51	0716258105	199720013005
SH1P51	0751246987	199720013005
SH1P52	0756285105	188952634561
SH1P52	0751245687	152100013005
SH3P55	0766250005	178320013005
SH3P55	0766250005	200020013005

UV4 23 x

Action Output

#	Time	Action	Message	Duration / Fetch
26	16:03:59	SELECT * FROM UV2 LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
27	16:04:08	CREATE VIEW UV3 AS SELECT * FROM SHIP_NAME AS SN LEFT JOIN CABIN AS C ON SN.SHIP_ID = C.SHIP WHERE SN.SHIP_ID != 'SH1'	0 row(s) affected	0.015 sec
28	16:04:08	SELECT * FROM UV3 LIMIT 0, 1000	5 row(s) returned	0.016 sec / 0.000 sec
29	16:05:00	CREATE VIEW UV4 AS SELECT PC.PASSENGER_ID, PC.CONTACT_NUMBER, PP.PASSPORT FROM PASSENGER_CONTACT AS PC RIGHT JOIN	0 row(s) affected	0.031 sec
30	16:05:00	SELECT * FROM UV4 LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec

Figure 5-15

```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Databases
683 -- 9 FULL OUTER JOIN
684
685 • CREATE VIEW UV5 AS SELECT EN.EMPLOYEE_ID, EN.NIC, CI.INCHARGE FROM EMPLOYEE_NIC AS EN LEFT OUTER JOIN CABIN_INCHARGE AS CI ON EN.EMPLOYEE_ID = CI.CABIN_INCHARGE_ID
686 • CREATE VIEW UV6 AS SELECT CI.INCHARGE, EN.EMPLOYEE_ID, EN.NIC FROM CABIN_INCHARGE AS CI RIGHT OUTER JOIN EMPLOYEE_NIC AS EN ON EN.EMPLOYEE_ID = CI.CABIN_INCHARGE_ID
687 • (SELECT * FROM UV5) UNION (SELECT * FROM UV6)
688
689 -- 10
690
691 • CREATE VIEW UV7 AS SELECT EN.EMPLOYEE_ID, EN.NIC FROM EMPLOYEE_NIC AS EN NATURAL JOIN EMPLOYEE_WORKED_SHIP AS EWS;
Result Grid | Filter Rows: Export: Wrap Cell Content: 
EMPLOYEE_ID NIC INCHARGE
SH1EMP1 200145246325 SH1EMP1
SH1EMP2 200045786325 SH1EMP2
SH1EMP3 199845636575 SH1EMP3
SH2EMP1 199945212525 SH2EMP1
SH2EMP2 200045462325 SH2EMP2
SH2EMP3 200145246325 SH2EMP3
SH3EMP1 199845636575 SH3EMP1
SH3EMP2 199945212525 SH3EMP2
SH3EMP3 199845636123 SH3EMP3
SH4EMP1 199845462325 SH4EMP1
SH4EMP2 200145462325 SH4EMP2
SH4EMP3 199945212525 SH4EMP3
SH5EMP1 199845636123 SH5EMP1
SH5EMP2 200145462325 SH5EMP2
SH5EMP3 199945212525 SH5EMP3
Result 24 ×
Output
Action Output
# Time Action Message Duration / Fetch
29 16:05:00 CREATE VIEW UV4 AS SELECT PC.PASSENGER_ID, PC.CONTACT_NUMBER, PP.PASSPORT_NUMBER, PC.FIRST_NAME, PC.LAST_NAME, PC.GENDER, PC.BIRTH_DATE, PC.CREATED_BY, PC.CREATED_DATE, PC.UPDATED_BY, PC.UPDATED_DATE, PC.IS_DELETED, PC.REASON_IS_DELETED, PC.CREATED_IP, PC.UPDATED_IP, PC.IS_ACTIVE, PC.IS_DELETED AS IS_DELETED, PC.CREATED_BY AS CREATED_BY, PC.UPDATED_BY AS UPDATED_BY, PC.IS_DELETED AS IS_DELETED, PC.REASON_IS_DELETED AS REAS... 0 row(s) affected 0.031 sec
30 16:05:00 SELECT * FROM UV4 LIMIT 0, 1000 7 row(s) returned 0.000 sec / 0.000 sec
31 16:05:19 CREATE VIEW UV5 AS SELECT EN.EMPLOYEE_ID, EN.NIC, CI.INCHARGE FROM EMPLOYEE_NIC AS EN INNER JOIN CABIN_INCHARGE AS CI ON EN.EMPLOYEE_ID = CI.CABIN_INCHARGE_ID 0 row(s) affected 0.016 sec
32 16:05:19 CREATE VIEW UV6 AS SELECT CI.INCHARGE, EN.EMPLOYEE_ID, EN.NIC FROM CABIN_INCHARGE AS CI RIGHT OUTER JOIN EMPLOYEE_NIC AS EN ON EN.EMPLOYEE_ID = CI.CABIN_INCHARGE_ID 0 row(s) affected 0.016 sec
33 16:05:19 (SELECT * FROM UV5) UNION (SELECT * FROM UV6) 12 row(s) returned 0.015 sec / 0.000 sec

```

Figure 5-16

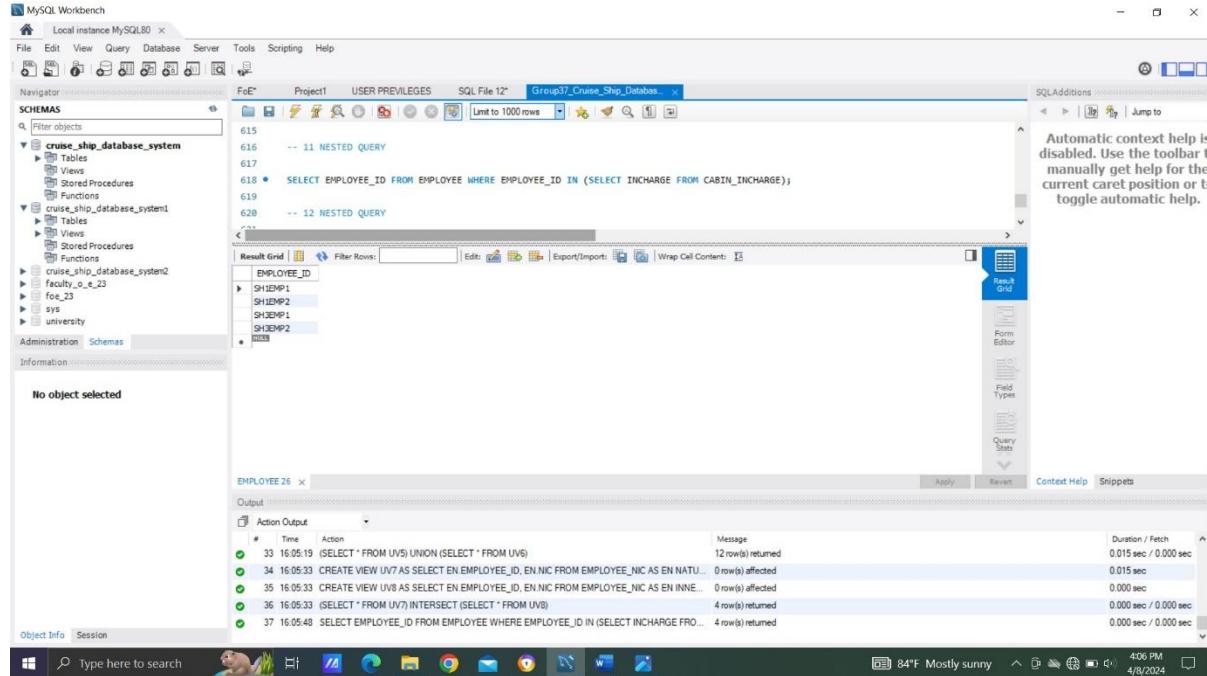
```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Databases
690 -- 10
691
692
693 • CREATE VIEW UV7 AS SELECT EN.EMPLOYEE_ID, EN.NIC FROM EMPLOYEE_NIC AS EN NATURAL JOIN EMPLOYEE_WORKED_SHIP AS EWS;
694 • CREATE VIEW UV8 AS SELECT EN.EMPLOYEE_ID, EN.NIC FROM EMPLOYEE_NIC AS EN INNER JOIN EMPLOYEE_WORKED_SHIP AS EWS ON EN.EMPLOYEE_ID = EWS.EMPLOYEE_ID
695 • (SELECT * FROM UV7) INTERSECT (SELECT * FROM UV8);
696
697 -- 11 NESTED QUERY
Result Grid | Filter Rows: Export: Wrap Cell Content: 
EMPLOYEE_ID NIC
SH1EMP1 200145246325
SH1EMP2 200045786325
SH1EMP3 199945212525
SH2EMP1 199845636575
Result 25 ×
Output
Action Output
# Time Action Message Duration / Fetch
32 16:05:19 CREATE VIEW UV7 AS SELECT CI.INCHARGE, EN.EMPLOYEE_ID, EN.NIC FROM CABIN_INCHARGE AS CI LEFT OUTER JOIN EMPLOYEE_NIC AS EN ON CI.CABIN_INCHARGE_ID = EN.EMPLOYEE_ID 0 row(s) affected 0.015 sec
33 16:05:19 (SELECT * FROM UV7) UNION (SELECT * FROM UV6) 12 row(s) returned 0.015 sec / 0.000 sec
34 16:05:33 CREATE VIEW UV7 AS SELECT EN.EMPLOYEE_ID, EN.NIC FROM EMPLOYEE_NIC AS EN NATURAL JOIN EMPLOYEE_WORKED_SHIP AS EWS; 0 row(s) affected 0.015 sec
35 16:05:33 CREATE VIEW UV8 AS SELECT EN.EMPLOYEE_ID, EN.NIC FROM EMPLOYEE_NIC AS EN INNER JOIN EMPLOYEE_WORKED_SHIP AS EWS ON EN.EMPLOYEE_ID = EWS.EMPLOYEE_ID 0 row(s) affected 0.000 sec
36 16:05:33 (SELECT * FROM UV7) INTERSECT (SELECT * FROM UV8) 4 row(s) returned 0.000 sec / 0.000 sec

```

Figure 5-17

5.3. Nested Queries



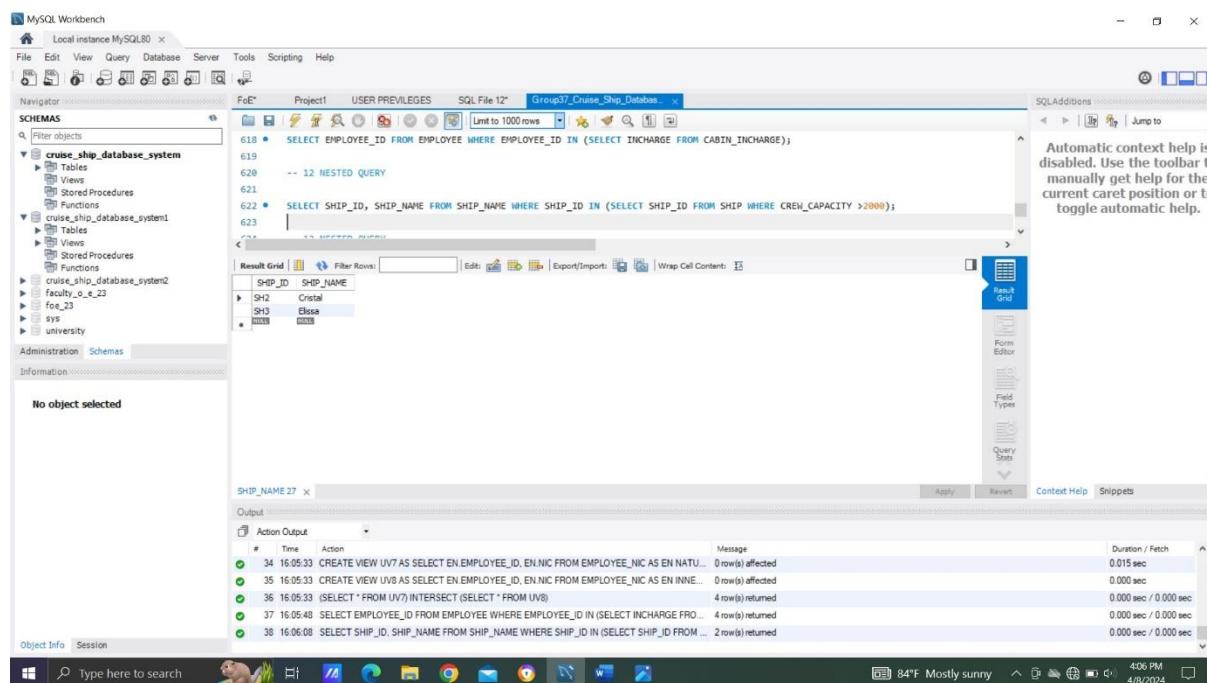
The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** The current schema is `cruise_ship_database_system`.
- Tables:** The `EMPLOYEE` table is selected.
- SQL Editor:** The query window displays nested queries:


```

615
616    -- 11 NESTED QUERY
617
618 •   SELECT EMPLOYEE_ID FROM EMPLOYEE WHERE EMPLOYEE_ID IN (SELECT INCHARGE FROM CABIN_INCHARGE);
619
620    -- 12 NESTED QUERY
      
```
- Result Grid:** The results show employee IDs SH1EMP1 through SH1EMP2.
- Output Window:** Shows the execution history for the current session, including the creation of views UV7 and UV8, and the execution of nested queries.
- System Tray:** Displays the date and time as 4/8/2024, 4:06 PM.

Figure 5-18



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

- Schemas:** The current schema is `cruise_ship_database_system`.
- Tables:** The `SHIP` table is selected.
- SQL Editor:** The query window displays nested queries:


```

618 •   SELECT EMPLOYEE_ID FROM EMPLOYEE WHERE EMPLOYEE_ID IN (SELECT INCHARGE FROM CABIN_INCHARGE);
619
620    -- 12 NESTED QUERY
621
622 •   SELECT SHIP_ID, SHIP_NAME FROM SHIP_NAME WHERE SHIP_ID IN (SELECT SHIP_ID FROM SHIP WHERE CREW_CAPACITY >2000);
623
      
```
- Result Grid:** The results show ship IDs SH2 and SH3.
- Output Window:** Shows the execution history for the current session, including the creation of views UV7 and UV8, and the execution of nested queries.
- System Tray:** Displays the date and time as 4/8/2024, 4:06 PM.

Figure 5-19

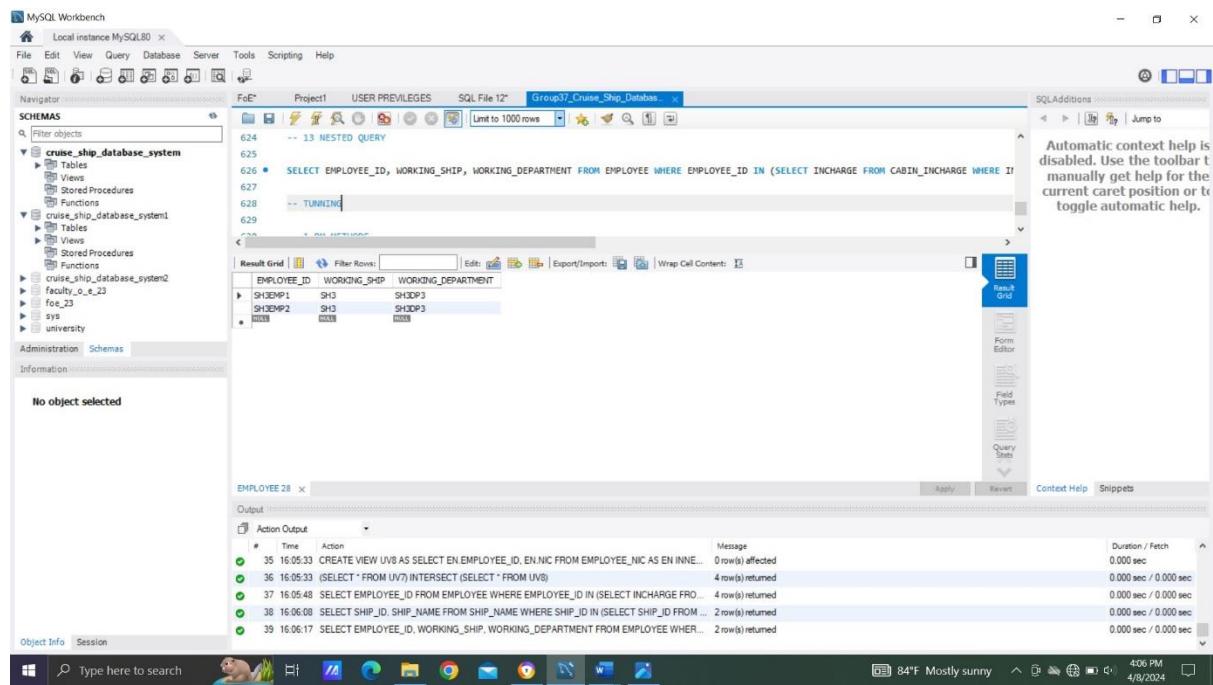


Figure 5-20

6. Database Tuning

Query 1

```

MySQL Workbench - Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help
Navigator Project1 USER PRIVILEGES SQL File 12" Limit to 1000 rows
Schemas
cruise_ship_database
cruise_ship_data
Information
No object selected

Result Grid | Filter Rows | Export: | Wrap Cell Contents: |
id select_type table partitions type possible_keys key key_len ref rows fbs
1 PRIMARY D ALL ref PRIMARY,`idx_department_ship_id` `idx_department_ship_id` 82 const 2 100
2 PRIMARY E ALL ref `idx_employee_working_department` `idx_employee_working_department` 123 const cruise_ship_database_system1.D.DEPARTMEN... 1 100
3 UNION S ALL const PRIMARY 50 const 2 100
4 UNION E ALL ref EMPL_INDEX1 `EMPL_INDEX1` 83 const 2 100
5 UNION RESULT <union1,2> ALL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL

```

Output

#	Time	Action	Message	Duration / Fetch
19	1721:19	EXPLAIN SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E WHERE E.WORKING_SHIP = 'SH3';	4 rows(s) returned.	0.000 sec / 0.000 sec
20	1721:27	SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E WHERE E.WORKING_SHIP = 'SH3';	4 rows(s) returned.	0.000 sec / 0.000 sec
21	1721:37	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID WHERE S.SHIP_ID = 'SH3');	5 rows(s) returned.	0.016 sec / 0.000 sec
22	1721:59	explain (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID WHERE S.SHIP_ID = 'SH3');	5 rows(s) returned.	0.000 sec / 0.000 sec
23	1729:04	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID WHERE S.SHIP_ID = 'SH3');	5 rows(s) returned.	0.000 sec / 0.000 sec

Figure 6-1

The above figure shows the query result before tuning and the below figure shows the result after the tuning. Here basically we use the query optimization method for the tuning. You can see in Figure 6-1 the number of row access is hire than in Figure 6-2.

```

MySQL Workbench - Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help
Navigator Project1 USER PRIVILEGES SQL File 12" Limit to 1000 rows
Schemas
cruise_ship_database
cruise_ship_data
Information
No object selected

Result Grid | Filter Rows | Export: | Wrap Cell Contents: |
id select_type table partitions type possible_keys key key_len ref rows fbs
1 PRIMARY D ALL ref PRIMARY,`idx_department_ship_id` `idx_department_ship_id` 82 const 2 100
2 PRIMARY E ALL ref `idx_employee_working_department` `idx_employee_working_department` 123 const cruise_ship_database_system1.D.DEPARTMEN... 1 100
3 UNION S ALL const PRIMARY 50 const 2 100
4 UNION E ALL ref EMPL_INDEX1 `EMPL_INDEX1` 83 const 2 100
5 UNION RESULT <union1,3> ALL NULL NULL NULL NULL NULL NULL NULL NULL NULL NULL

```

Output

#	Time	Action	Message	Duration / Fetch
23	1729:04	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID WHERE S.SHIP_ID = 'SH3');	5 rows(s) returned.	0.000 sec / 0.000 sec

Figure 6-2

Query 2

```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
Navigator Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Databases ...
689 • EXPLAIN SELECT * FROM UV2;
690
691 -- 7 INDEX TUNED
692
693 -- CREATE VIEW UV3 AS SELECT * FROM SHIP_NAME AS SN LEFT JOIN CABIN AS C ON SN.SHIP_ID = C.SHIP WHERE SN.SHIP_ID != 'SH1';
694 • EXPLAIN SELECT * FROM UV3;
695 • CREATE INDEX CABIN_TYPE_INDEX ON CABIN(CABIN_TYPE);
696 • DROP INDEX CABIN_TYPE_INDEX ON CABIN;
697

Result Grid | Filter Rows: [ ] Export: [ ] Wrap Cell Content: [ ]
id select_type table partitions type possible_keys key key_len ref rows filtered Extra
1 SIMPLE sn ALL range PRIMARY PRIMARY 50 const 5 100.00 Using where
1 SIMPLE c ALL FK_CABIN_BELONG_SHIP FK_CABIN_BELONG_SHIP 82 cruise_ship_database_system1.cn.SHIP_ID 1 100.00 Using where; Using join buffer (hash join)

Result 22 x
Action Output
# Time Action Message Duration / Fetch
29 17:31:43 EXPLAIN SELECT PC.PASSENGER_ID FROM PASSENGER_PASPORT AS PP INNER JOIN PASSENGER_C_... 3 rows) returned 0.000 sec / 0.000 sec
30 17:31:56 EXPLAIN (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMEN... 5 rows) returned 0.000 sec / 0.000 sec
31 17:32:17 EXPLAIN (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMEN... 5 rows) returned 0.000 sec / 0.000 sec
32 17:32:50 EXPLAIN SELECT * FROM UV1 2 rows) returned 0.000 sec / 0.000 sec
33 17:33:03 EXPLAIN SELECT * FROM UV2 2 rows) returned 0.000 sec / 0.000 sec
34 17:33:20 EXPLAIN SELECT * FROM UV3 2 rows) returned 0.000 sec / 0.000 sec

Object Info Session

```

Figure 6-3

Here for the tuning we used the indexing method. The above figure shows the result before indexing and the below figure shows the result after the indexing. In Figure 6-4 the number of row accessed is lower than Figure 6-3.

```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
Navigator Project1 USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Databases ...
689 • EXPLAIN SELECT * FROM UV2;
690
691 -- 7 INDEX TUNED
692
693 -- CREATE VIEW UV3 AS SELECT * FROM SHIP_NAME AS SN LEFT JOIN CABIN AS C ON SN.SHIP_ID = C.SHIP WHERE SN.SHIP_ID != 'SH1';
694 • EXPLAIN SELECT * FROM UV3;
695 • CREATE INDEX CABIN_TYPE_INDEX ON CABIN(CABIN_TYPE);
696 • DROP INDEX CABIN_TYPE_INDEX ON CABIN;
697

Result Grid | Filter Rows: [ ] Export: [ ] Wrap Cell Content: [ ]
id select_type table partitions type possible_keys key key_len ref rows filtered Extra
1 SIMPLE sn ALL range PRIMARY PRIMARY 50 const 5 100.00 Using where
1 SIMPLE c ALL FK_CABIN_BELONG_SHIP FK_CABIN_BELONG_SHIP 82 cruise_ship_database_system1.cn.SHIP_ID 1 100.00 Using where

Result 23 x
Action Output
# Time Action Message Duration / Fetch
31 17:32:17 EXPLAIN (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMEN... 5 rows) returned 0.000 sec / 0.000 sec
32 17:32:50 EXPLAIN SELECT * FROM UV1 2 rows) returned 0.000 sec / 0.000 sec
33 17:33:03 EXPLAIN SELECT * FROM UV2 2 rows) returned 0.000 sec / 0.000 sec
34 17:33:20 EXPLAIN SELECT * FROM UV3 2 rows) returned 0.000 sec / 0.000 sec
35 17:33:31 CREATE INDEX CABIN_TYPE_INDEX ON CABIN(CABIN_TYPE) 0 rows) affected Records: 0 Duplicates: 0 Warnings: 0 0.054 sec
36 17:33:38 EXPLAIN SELECT * FROM UV3 2 rows) returned 0.000 sec / 0.000 sec

Object Info Session

```

Figure 6-4

Query 3

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

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard database management tools.
- Schemas:** Local instance MySQL 8.0, showing the cruise_ship_database schema.
- SQL Editor:** Group37_Cruise_Ship_Databases, displaying the query:


```
649 • EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)
INTERSECT
652 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE E.WORKING_SHIP = 'SH1')
653
```
- Result Grid:** Shows the execution plan with columns: id, select_type, table, partitions, type, possible_keys, key, key_len, ref, rows, filtered, Extra.
- Output:** Action Output section showing a timeline of events from 17:20:27 to 17:30:06, including index drops and EXPLAIN statements.
- System Bar:** Shows the date as 4/8/2024, time as 5:30 PM, and weather as 84°F Mostly sunny.

Figure 6-5

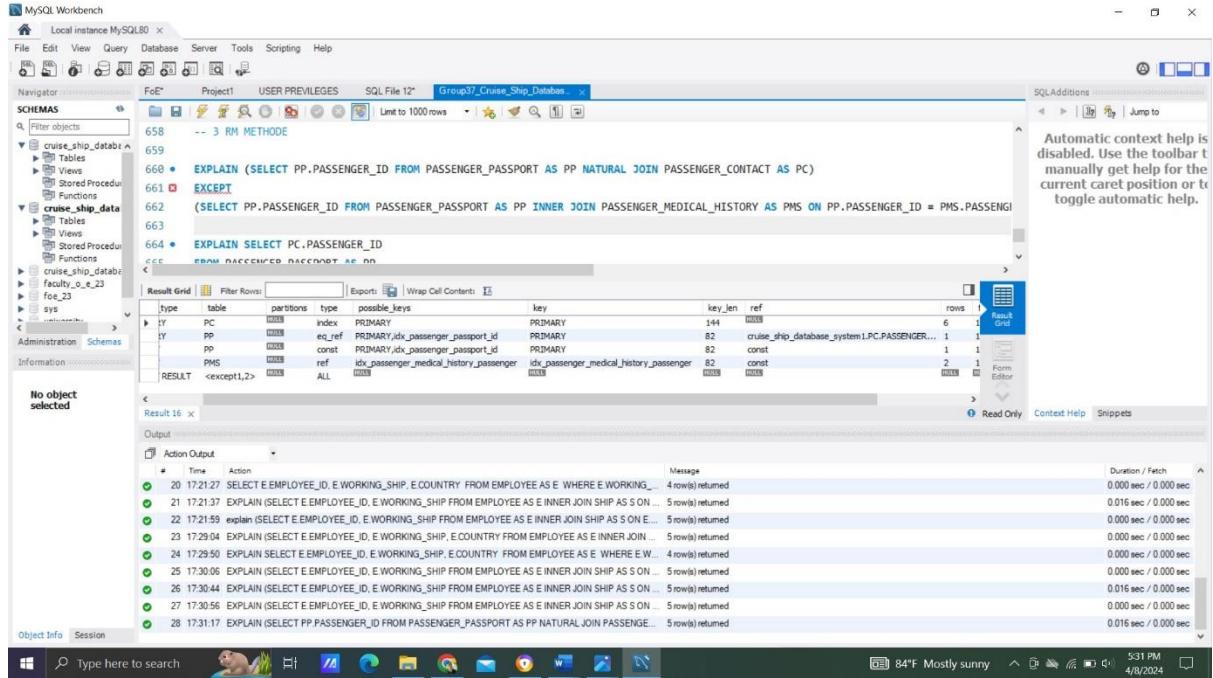
The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard database management tools.
- Schemas:** Local instance MySQL 8.0, showing the cruise_ship_database schema.
- SQL Editor:** Group37_Cruise_Ship_Databases, displaying the query:


```
652 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE E.WOI^
653
654 • EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID WHERE S.SHIP_ID
INTERSECT
656 (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE E.WOI^
657
658 -- 3 RM METHODE
EEO
```
- Result Grid:** Shows the execution plan with columns: id, select_type, table, partitions, type, possible_keys, key, key_len, ref, rows, filtered, Extra.
- Output:** Action Output section showing a timeline of events from 17:21:19 to 17:30:56, including EXPLAIN statements and index statistics.
- System Bar:** Shows the date as 4/8/2024, time as 5:31 PM, and weather as 84°F Mostly sunny.

Figure 6-6

Query 4



MySQL Workbench - Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS cruise_ship_database

Tables Views Stored Procedures Functions

cruise_ship_data

Tables Views Stored Procedures Functions

cruise_ship_database

Tables Views Stored Procedures Functions

faculty_o_e_23

sys

Administration Schemas

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Contents: Result Grid Form Editor

SQL File 12: GroupB7_Cruise_Ship_Databases

Limit to 1000 rows

658 -- 3 RM METHODE

659

660 • EXPLAIN (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP NATURAL JOIN PASSENGER_CONTACT AS PC)

661 EXCEPT

662 (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP INNER JOIN PASSENGER_MEDICAL_HISTORY AS PMS ON PP.PASSENGER_ID = PMS.PASSENGER_ID)

663

664 • EXPLAIN SELECT PC.PASSENGER_ID

665 FROM PASSENGER_PASSPORT AS PP

666

667

668 WHERE PP.PASSENGER_ID <> 'SHIPS1' OR PMS.PASSENGER IS NULL;

669

670 -- 4 RM METHODE

671

Result 16 x

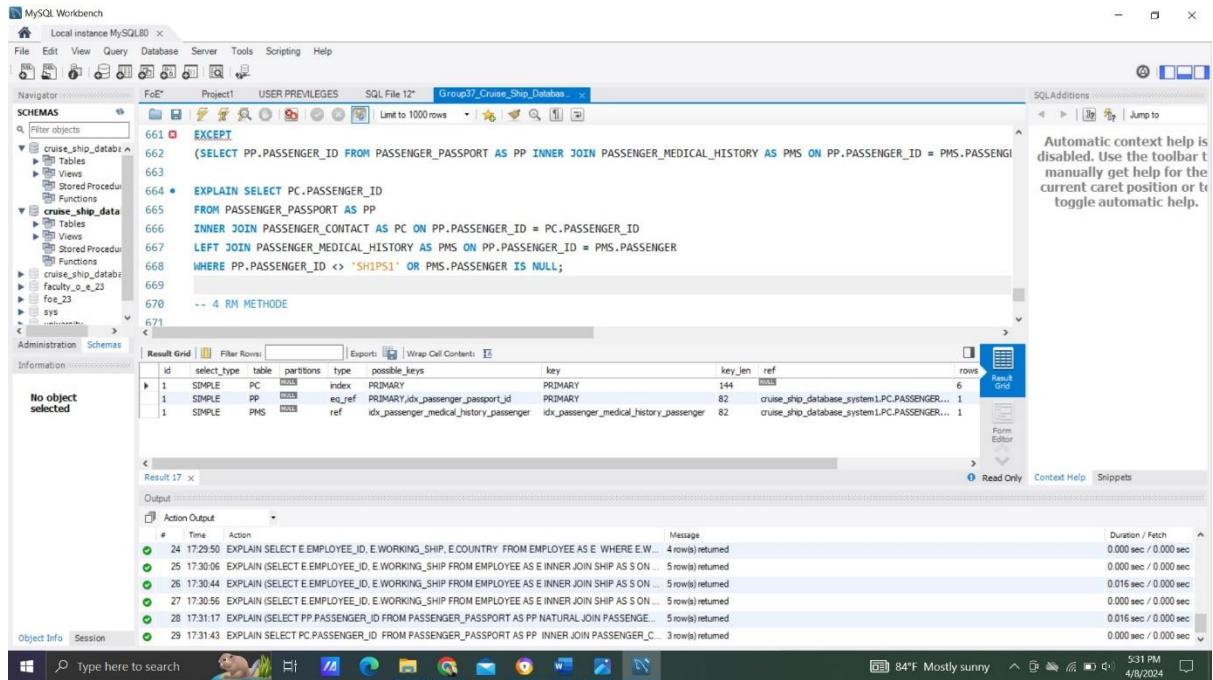
Action Output

#	Time	Action	Message	Duration / Fetch
20	17:21:27	SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E WHERE E.WORKING_SHIP = S.SHIP_ID	4 row(s) returned	0.000 sec / 0.000 sec
21	17:21:37	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.016 sec / 0.000 sec
22	17:21:59	explain (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.000 sec / 0.000 sec
23	17:29:04	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.000 sec / 0.000 sec
24	17:29:50	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E WHERE E.WORKING_SHIP = S.SHIP_ID)	4 row(s) returned	0.000 sec / 0.000 sec
25	17:30:06	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.000 sec / 0.000 sec
26	17:30:44	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.016 sec / 0.000 sec
27	17:30:56	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.000 sec / 0.000 sec
28	17:31:17	EXPLAIN (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP NATURAL JOIN PASSENGER_CONTACT)	5 row(s) returned	0.016 sec / 0.000 sec
29	17:31:43	EXPLAIN SELECT PC.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP INNER JOIN PASSENGER_CONTACT AS PC	3 row(s) returned	0.000 sec / 0.000 sec

Output

84°F Mostly sunny 5:31 PM 4/8/2024

Figure 6-7



MySQL Workbench - Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator Schemas

SCHEMAS cruise_ship_database

Tables Views Stored Procedures Functions

cruise_ship_data

Tables Views Stored Procedures Functions

cruise_ship_database

Tables Views Stored Procedures Functions

faculty_o_e_23

sys

Administration Schemas

No object selected

Result Grid | Filter Rows: Export: Wrap Cell Contents: Result Grid Form Editor

SQL File 12: GroupB7_Cruise_Ship_Databases

Limit to 1000 rows

661 EXCEPT

662 (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP INNER JOIN PASSENGER_MEDICAL_HISTORY AS PMS ON PP.PASSENGER_ID = PMS.PASSENGER_ID)

663

664 • EXPLAIN SELECT PC.PASSENGER_ID

665 FROM PASSENGER_PASSPORT AS PP

666

667

668 WHERE PP.PASSENGER_ID <> 'SHIPS1' OR PMS.PASSENGER IS NULL;

669

670 -- 4 RM METHODE

671

Result 17 x

Action Output

#	Time	Action	Message	Duration / Fetch
24	17:29:50	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP, E.COUNTRY FROM EMPLOYEE AS E WHERE E.WORKING_SHIP = S.SHIP_ID)	4 row(s) returned	0.000 sec / 0.000 sec
25	17:30:06	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.000 sec / 0.000 sec
26	17:30:44	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.016 sec / 0.000 sec
27	17:30:56	EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON E.WORKING_SHIP = S.SHIP_ID)	5 row(s) returned	0.000 sec / 0.000 sec
28	17:31:17	EXPLAIN (SELECT PP.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP NATURAL JOIN PASSENGER_CONTACT)	5 row(s) returned	0.016 sec / 0.000 sec
29	17:31:43	EXPLAIN SELECT PC.PASSENGER_ID FROM PASSENGER_PASSPORT AS PP INNER JOIN PASSENGER_CONTACT AS PC	3 row(s) returned	0.000 sec / 0.000 sec

Output

84°F Mostly sunny 5:31 PM 4/8/2024

Figure 6-8

Query 5

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

- Toolbar:** Standard MySQL Workbench tools like File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** Shows the schema tree for the 'cruise_ship_database' and 'cruise_ship_data' schemas.
- SQL Editor:** Contains the following SQL code:


```

670 -- 4 RM METHODE
671
672 • EXPLAIN (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMENT AS D ON E.EMPLOYEE_ID = D.DEPARTMENT_HEAD
673   INTERSECT
674   (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE CI
675     )
676   EXPLAIN (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMENT AS D ON E.EMPLOYEE_ID = D.DEPARTMENT_HEAD
677   INTERSECT
678   (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE);
      
```
- Result Grid:** Displays the execution plan for the EXPLAIN statements. The first EXPLAIN statement shows:

ID	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	PRIMARY	E	NULL	ALL	NULL	NULL	NULL	NULL	5	100.00	Using where
1	PRIMARY	D	NULL	ref	FK_DEPARTMENT_HEAD	FK_DEPARTMENT_HEAD	83	cruise_ship_database_system1.E.EMPLOYEE_ID	1	100.00	Using where
2	INTERSECT	CI	NULL	index	idx_incharge	idx_incharge	82	NULL	6	16.67	Using where
2	INTERSECT	E	NULL	eq_ref	PRIMARY	PRIMARY	50	cruise_ship_database_system1.CI.INCHARGE	1	100.00	Using where
3	INTERSECT RESULT	<intersect1,2>	NULL	ALL	NULL	NULL	NULL	NULL	NULL	NULL	Using where
- Output Panel:** Shows the execution log with actions, time, and duration.
- Status Bar:** Displays system information like temperature (84°F), weather (Mostly sunny), date (4/8/2024), and time (5:32 PM).

Figure 6-9

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

- Toolbar:** Standard MySQL Workbench tools like File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** Shows the schema tree for the 'cruise_ship_database' and 'cruise_ship_data' schemas.
- SQL Editor:** Contains the following SQL code:


```

674 (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE WHERE CI
675
676 • EXPLAIN (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN DEPARTMENT AS D ON E.EMPLOYEE_ID = D.DEPARTMENT_HEAD
677   INTERSECT
678   (SELECT E.FIRST_NAME, E.LAST_NAME, E.CITY FROM EMPLOYEE AS E LEFT JOIN CABIN_INCHARGE AS CI ON E.EMPLOYEE_ID = CI.INCHARGE);
      
```
- Result Grid:** Displays the execution plan for the EXPLAIN statements. The first EXPLAIN statement shows:

ID	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	PRIMARY	E	NULL	ref	EMPL_INDEX1	EMPL_INDEX1	83	const	2	100.00	
1	PRIMARY	D	NULL	ref	FK_DEPARTMENT_HEAD	FK_DEPARTMENT_HEAD	83	cruise_ship_database_system1.E.EMPLOYEE_ID	1	100.00	Using where
2	INTERSECT	CI	NULL	ALL	NULL	NULL	NULL	NULL	5	100.00	
2	INTERSECT	CI	NULL	ref	idx_incharge	idx_incharge	82	cruise_ship_database_system1.E.EMPLOYEE_ID	1	100.00	Using where
3	INTERSECT RESULT	<intersect1,2>	NULL	ALL	NULL	NULL	NULL	NULL	NULL	NULL	Using where
- Output Panel:** Shows the execution log with actions, time, and duration.
- Status Bar:** Displays system information like temperature (84°F), weather (Mostly sunny), date (4/8/2024), and time (5:32 PM).

Figure 6-10

Query 6

This query has already tuned.

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

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** Schemas, cruise_ship_database, Tables, Views, Stored Procedures, Functions, cruise_ship_data, Tables, Views, Stored Procedures, Functions.
- SQL Editor:** SQL File 12*, Group37_Cruise_Ship_Database. The code is:

```
680 -- 5 ALREADY TUNED
681
682 -- CREATE VIEW UV1 AS SELECT S.SHIP_ID, SN.SHIP_NAME FROM SHIP_NAME AS SN NATURAL JOIN SHIP AS S;
683 • EXPLAIN SELECT * FROM UV1;
684
685 -- 6 TUNED
```
- Result Grid:** Shows the execution plan for the EXPLAIN query. It includes columns: id, select_type, table, partitions, type, possible_keys, key, key_len, ref, rows, filtered, Extra. One row is shown: 1 SIMPLE s index PRIMARY PRIMARY 50 NULL 4 100.00 Using index.
- Action Output:** Shows the execution history with columns: #, Time, Action, Message, Duration / Fetch. Rows include: 27 17:30:56 EXPLAIN (SELECT E.EMPLOYEE_ID, E.WORKING_SHIP FROM EMPLOYEE AS E INNER JOIN SHIP AS S ON ... 5 rows) returned, 0.000 sec / 0.000 sec; 28 17:31:17 EXPLAIN (SELECT PP.PASSENGER_ID FROM PASSENGER_PASPORT AS PP NATURAL JOIN PASSENGER_C_... 5 rows) returned, 0.016 sec / 0.000 sec; etc.
- System Bar:** Read Only, Context Help, Snippets, 84°F Mostly sunny, 5:32 PM, 4/8/2024.

Figure 6-11

Query 7

This query also already tuned.

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

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** Schemas, cruise_ship_database, Tables, Views, Stored Procedures, Functions, cruise_ship_data, Tables, Views, Stored Procedures, Functions.
- SQL Editor:** SQL File 12*, Group37_Cruise_Ship_Database. The code is:

```
686
687 -- CREATE VIEW UV2 AS SELECT EN.EMPLOYEE_ID, EN.NIC, D.DEPARTMENT_NAME FROM EMPLOYEE_NIC AS EN INNER JOIN
688 -- DEPARTMENT AS D ON EN.EMPLOYEE_ID = D.DEPARTMENT_HEAD;
689 • EXPLAIN SELECT * FROM UV2;
690
691 -- 7 INDEX TUNED
```
- Result Grid:** Shows the execution plan for the EXPLAIN query. It includes columns: id, select_type, table, partitions, type, possible_keys, key, key_len, ref, rows, filtered, Extra. One row is shown: 1 SIMPLE d ALL FK_DEPARTMENT_HEAD NULL NULL 5 100.00 Using where.
- Action Output:** Shows the execution history with columns: #, Time, Action, Message, Duration / Fetch. Rows include: 28 17:31:17 EXPLAIN (SELECT PP.PASSENGER_ID FROM PASSENGER_PASPORT AS PP NATURAL JOIN PASSENGER_C_... 5 rows) returned, 0.016 sec / 0.000 sec; 29 17:31:43 EXPLAIN (SELECT PC.PASSENGER_ID FROM PASSENGER_PASPORT AS PP INNER JOIN PASSENGER_C_... 3 rows) returned, 0.000 sec / 0.000 sec; etc.
- System Bar:** Read Only, Context Help, Snippets, 84°F Mostly sunny, 5:33 PM, 4/8/2024.

Figure 6-12

Query 8

Tuned query.

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:** Schemas, Tables, Views, Stored Procedures, Functions, cruise_ship_database, faculty_o_e_23, foe_23, sys.
- SQL Editor:** SQL File 12*, Group37_Cruise_Ship_Database, containing the following code:

```
697
698 -- 8 TUNNED
699
700 -- CREATE VIEW UV4 AS SELECT PC.PASSENGER_ID, PC.CONTACT_NUMBER, PP.PASSPORT FROM PASSENGER_CONTACT AS PC RIGHT JOIN
701 -- PASSENGER_PASSPORT AS PP ON PC.PASSENGER_ID = PP.PASSENGER_ID;
702 • EXPLAIN SELECT * FROM UV4;
703
704 -- 9 TUNNED
```
- Result Grid:** Shows the execution plan for the EXPLAIN command. The output table has columns: id, select_type, table, partitions, type, possible_keys, key, key_len, ref, rows, filtered, Extra. One row is shown:

1	SIMPLE	pp	ALL	ref	PRIMARY	PRIMARY	82	cruise_ship_database_system1.pp.PASSENGER_ID	1	100.00	Using index
---	--------	----	-----	-----	---------	---------	----	--	---	--------	-------------
- Action Output:** Shows the history of 38 MySQL sessions with their time, action, message, and duration/fetch time.
- System Bar:** Shows the date (4/8/2024), time (5:34 PM), and weather (84°F Mostly sunny).

Figure 6-13

Query 9

Tuned query.

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:** Schemas, Tables, Views, Stored Procedures, Functions, cruise_ship_database, faculty_o_e_23, foe_23, sys.
- SQL Editor:** SQL File 12*, Group37_Cruise_Ship_Database, containing the following code:

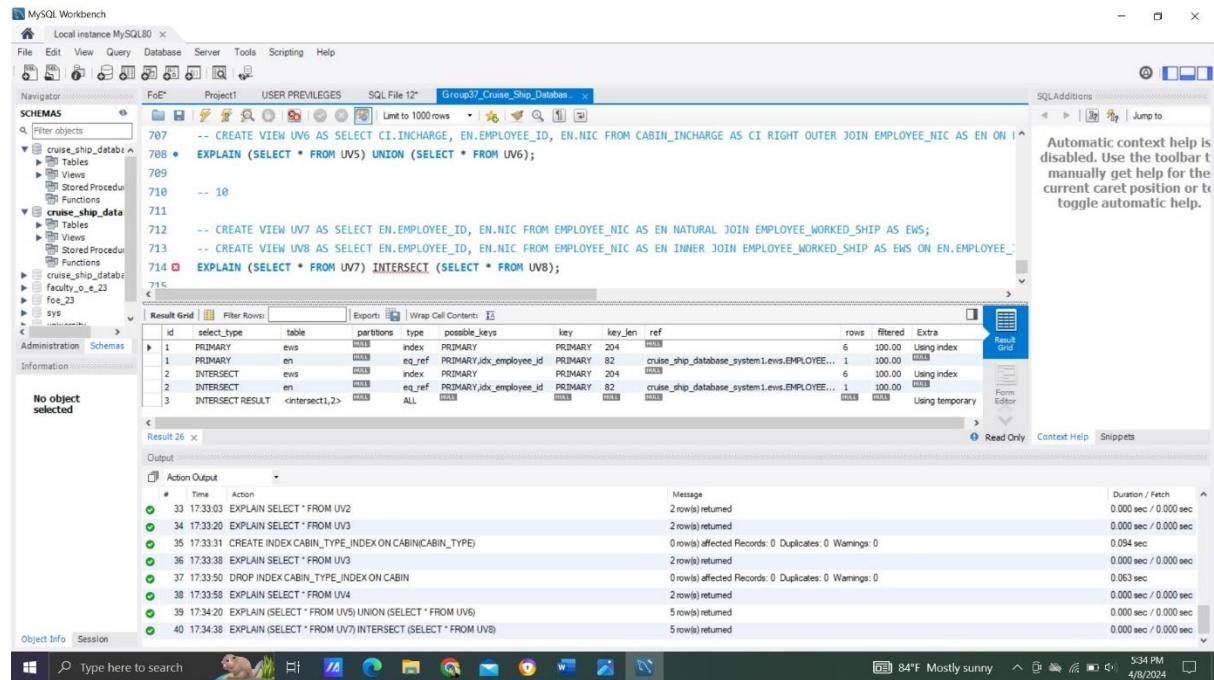
```
703
704 -- 9 TUNNED
705
706 -- CREATE VIEW UV5 AS SELECT EN.EMPLOYEE_ID, EN.NIC, CI.INCHARGE FROM EMPLOYEE AS EN LEFT JOIN DEPARTMENT AS DEP ON EN.DEPARTMENT_ID = DEP.ID
707 -- CREATE VIEW UV6 AS SELECT CI.INCHARGE, EN.EMPLOYEE_ID, EN.NIC FROM CABIN_INCHARGE AS CI RIGHT OUTER JOIN EMPLOYEE_NIC AS EN ON CI.EMPLOYEE_ID = EN.ID
708 • EXPLAIN (SELECT * FROM UV5) UNION (SELECT * FROM UV6);
709
710 -- 10
```
- Result Grid:** Shows the execution plan for the EXPLAIN command. The output table has columns: id, select_type, table, partitions, type, possible_keys, key, key_len, ref, rows, filtered, Extra. Three rows are shown:

1	PRIMARY	en	ALL	ref	idx_incharge	idx_incharge	82	cruise_ship_database_system1.en.EMPLOYEE_ID	1	100.00	Using index
2	PRIMARY	d	ALL	ref	idx_incharge	idx_incharge	82	cruise_ship_database_system1.en.EMPLOYEE_ID	1	100.00	Using index
3	UNION RESULT	union1,2	ALL	ref	idx_incharge	idx_incharge	82	cruise_ship_database_system1.en.EMPLOYEE_ID	1	100.00	Using temporary
- Action Output:** Shows the history of 39 MySQL sessions with their time, action, message, and duration/fetch time.
- System Bar:** Shows the date (4/8/2024), time (5:34 PM), and weather (84°F Mostly sunny).

Figure 6-14

Query 10

Tuned query.



The screenshot shows the MySQL Workbench interface with a SQL editor window containing a tuned query. The code includes several EXPLAIN statements and UNION operations. The results grid shows the execution plan with columns for id, select_type, table, partitions, type, possible_keys, key, key_len, ref, rows, filtered, and Extra. The action output panel at the bottom shows the timeline of events, including EXPLAIN statements, index creation, and the final query execution. The status bar at the bottom right indicates the system is at 84°F, mostly sunny, with a timestamp of 5:34 PM on 4/8/2024.

```
File Edit View Query Database Server Tools Scripting Help
Navigator Project! USER PRIVILEGES SQL File 12* Group37_Cruise_Ship_Database ...
Schemas
cruise_ship_database
Tables Views Stored Procedures Functions
cruise_ship_data
Faculty_o_e_23 foe_23 sys
Administration Schemas Information
No object selected
Result Grid Filter Rows! Export: Wrap Cell Contents
id select_type table partitions type possible_keys key key_len ref rows filtered Extra
1 PRIMARY evs 1 index PRIMARY PRIMARY 204
1 PRIMARY en 1 eq_ref PRIMARY,`idx_employee_id` PRIMARY 82 cruise_ship_database_system1.evs.EMPLOYEE...
2 INTERSECT evs 1 index PRIMARY PRIMARY 204
2 INTERSECT en 1 eq_ref PRIMARY,`idx_employee_id` PRIMARY 82 cruise_ship_database_system1.evs.EMPLOYEE...
3 INTERSECT RESULT <intersect1,2> 1 ALL
Result Grid Form Editor
Read Only Context Help Snippets
Output
Action Output
# Time Action Message Duration / Fetch
33 17:33:03 EXPLAIN SELECT * FROM UV2 2 rows(s) returned 0.000 sec / 0.000 sec
34 17:33:20 EXPLAIN SELECT * FROM UV3 2 rows(s) returned 0.000 sec / 0.000 sec
35 17:33:31 CREATE INDEX CABIN_TYPE_INDEX ON CABIN(CABIN_TYPE) 0 rows(s) affected Records: 0 Duplicates: 0 Warnings: 0 0.054 sec
36 17:33:38 EXPLAIN SELECT * FROM UVQ 2 rows(s) returned 0.000 sec / 0.000 sec
37 17:33:50 DROP INDEX CABIN_TYPE_INDEX ON CABIN 0 rows(s) affected Records: 0 Duplicates: 0 Warnings: 0 0.063 sec
38 17:33:58 EXPLAIN SELECT * FROM UV4 2 rows(s) returned 0.000 sec / 0.000 sec
39 17:34:20 EXPLAIN (SELECT * FROM UV5) UNION (SELECT * FROM UV6) 5 rows(s) returned 0.000 sec / 0.000 sec
40 17:34:38 EXPLAIN (SELECT * FROM UV7) INTERSECT (SELECT * FROM UV8) 5 rows(s) returned 0.000 sec / 0.000 sec
84°F Mostly sunny 5:34 PM 4/8/2024
```

Figure 6-15

In conclusion by comparing the number of row accessed in tunned queries is comparatively lower than the queries before tunning. For the tunning we can use various methods like indexing, query optimization and so on. In our case the most of the complex queries we wrote are almost tunned. But there were some queries which are not tunned. We used indexing method and query optimization method for the tunning.