

DATABASE MANAGEMENT SYSTEM

AND SQL BY BOARD IFINITY

Submitted in partial fulfilment of the requirements for the award of degree
of

B. Tech in

Computer Science and Engineering

Submitted to

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB



LOVELY
PROFESSIONAL
UNIVERSITY

From 05/06/23 to 10/07/23

SUBMITTED BY

Name of student: SHAIK JULFEEN AHMADH

Registration number: 12110554

Signature of the student:

A rectangular box containing a handwritten signature in black ink. The signature appears to be "Sh. Julfeen" written in a cursive, slightly stylized script.

S. No	Title	Page
1	Declaration by student	3
2	Training Certificate from Organization	4
3	Acknowledgement	5
4	Introduction of the Internship and Projects Undertaken	6 & 7
5	Board Infinity Dashboard	33
6	Conclusion	35
7	References	36

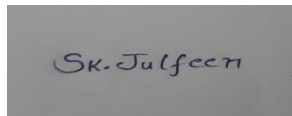
STUDENT DECLARATION

To whomsoever it may concern,

I, **SHAIK JULFEEN AHMADH**, bearing Registration number **12110554**, hereby declare that the work done by me on learning “**DATABASE MANAGEMENT SYSTEM & SQL**” from **June, 2023** to **July, 2023**, is a record of original work for the partial fulfilment of the requirements for the award of the degree **B. Tech, CSE**.

Name of the Student: SHAIK JULFEEN AHMADH (12110554)

Signature of Student :

A rectangular box containing a handwritten signature in cursive script that reads "SK. Julfeen".

Dated:01/08/23

SUMMER TRAINING CERTIFICATE

CERTIFICATE OF COMPLETION

THIS CERTIFICATE IS AWARDED TO

Shaik Julfeen Ahmadh

for successfully completing Microlearning Program in
Database Management System & Sql

12 July, 2023

ISSUED DATE



CEO, Board Infinity
Sumesh Nair

BI22LPBI345426088

CERTIFICATE NO.

BOARD

ACKNOWLEDGEMENT

I would like to express my gratitude towards my University as well as Board Infinity for providing me the golden opportunity to do this wonderful summer training internship DATABASE MANAGEMENT SYSTEM & SQL, which also helped me in doing a lot of homework and learning. As a result, I came toknow about so many things. So, I am really thankful to the whole team.

A heartfelt thank you to the dedicated instructors and mentors whose expertise and unwavering support were instrumental in my growth during the training. Their commitment to delivering quality education and addressing my queries has been exceptional.

Lastly, I want to express my gratitude to my family and friends for their encouragement and belief in my abilities. Their support motivated me to strive for excellence, and I look forward to applying the knowledge acquired in future endeavors.

INTRODUCTION OF THE INTERNSHIP

My internship experience with Board Infinity in Database Management Systems (DBMS) and SQL has been a valuable journey in expanding my knowledge and skills in this critical area of information technology.

Database Management Systems and SQL are foundational tools in the realm of data management. DBMS serves as a vital software system for effectively storing, retrieving, and managing data, while SQL is the language used to interact with these databases. This internship has allowed me to delve deep into the world of DBMS and SQL, providing me with a robust understanding of how to design, implement, and query databases.

The internship program was structured into four comprehensive weeks, during which I not only absorbed theoretical knowledge but also applied it practically through hands-on projects. These projects served as a bridge between theory and real-world application, enhancing my problem-solving skills and reinforcing the concepts I learned.

Moreover, the online nature of the course provided incredible flexibility. Daily classes were conducted from 8:00 PM to 9:30 PM, with recorded lectures available for reference. This flexibility allowed me to manage my learning schedule effectively. The course content, including video lectures and the availability of mentors to address queries, made the learning experience seamless and interactive.

In the following sections of this report, I will delve into the specific aspects of DBMS and SQL that I explored during this internship and highlight the projects I completed, showcasing how this experience has enriched my understanding of data management and analysis.

INTRODUCTION OF THE PROJECT UNDERTAKEN

➤ Objectives of the work undertaken :

Throughout the internship, I was given 2 projects to complete which includes #Creation of a database and draw ER diagram and #Perform few queries on HR dataset .

PHARMACY DATABASE :

The pharmacy database creation project aimed to develop a robust and efficient database system tailored for pharmaceutical management. The objectives included designing a structured database architecture, ensuring data integrity and security, enabling swift data retrieval, generating comprehensive reports for decision-making, implementing a user-friendly interface, and ensuring scalability for future requirements. Comprehensive documentation and training materials were also provided to facilitate easy usage. This project empowered effective pharmaceutical data management and streamlined daily operations within the pharmacy environment

Queries on HR dataset : Our mentor provided a database and we performed few queries on them which helped in understating few topics of Data analysis.

➤ **Scope of the work :**

During this internship, I delved into the realm of Database Management Systems (DBMS) and SQL, gaining valuable insights into these crucial technologies. My scope of work included comprehending the fundamentals of DBMS and SQL, learning to design and manage complex databases, and honing my skills in writing efficient SQL queries for data retrieval and manipulation. While I did not delve into server-side frameworks, I developed a strong foundation in understanding the backend operations of database-driven systems. This knowledge equips me to navigate the intricacies of real-world database environments effectively, enhancing my capabilities in data management and analysis.

➤ **Importance and Applicability of DBMS & SQL:**

Database Management Systems (DBMS) and SQL are integral components of contemporary data management across diverse fields. DBMS provides the foundation for organized data storage and retrieval, while SQL serves as the universal language for interacting with databases. Their ubiquity spans industries such as web development, finance, healthcare, and logistics, facilitating efficient data-driven decision-making, secure information storage, and seamless user experiences. Whether optimizing e-commerce, enhancing healthcare records, or enabling robust logistics management, DBMS and SQL empower organizations to harness the full potential of their data for operational excellence and informed choices.

➤ **Role and Profile:**

These assignments were meant to be completed alone, so I did them on my own utilising what I had already learnt. I worked on every aspect of the project, including the page's styling and any JavaScript functionality for the components.

BRIEF DESCRIPTION OF THE

PROJECTS DONE

1 .Pharmacy Database:

SQL CODE:

```
CREATE DATABASE Pharmacy;
use pharmacy;
CREATE TABLE `company` (
  `NAME` varchar(50) NOT NULL,
  `ADDRESS` varchar(50) NOT NULL,
  `PHONE` varchar(20) NOT NULL
);
INSERT INTO `company` (`NAME`, `ADDRESS`, `PHONE`)
VALUES
('Cipla', 'Mumbai', '12903'),
('Sun Pharma', 'Mysore', '01289078443'),
('Med_City', 'Nellore', '010114367832');

CREATE TABLE `drugs` (
  NAME varchar(50) NOT NULL,
  TYPE varchar(20) NOT NULL,
  BARCODE varchar(20) NOT NULL,
  DOSE varchar(10) NOT NULL,
  CODE varchar(10) NOT NULL,
  COST_PRICE double NOT NULL,
  SELLING_PRICE double NOT NULL,
  EXPIRY varchar(20) NOT NULL,
  COMPANY_NAME varchar(50) NOT NULL,
  PRODUCTION_DATE date NOT NULL,
  EXPIRATION_DATE date NOT NULL,
  PLACE varchar(20) NOT NULL,
  QUANTITY int(11) NOT NULL
);
INSERT INTO drugs (`NAME`, `TYPE`, `BARCODE`, `DOSE`,
```

```

`CODE`, `COST_PRICE`, `SELLING_PRICE`, `EXPIRY`,
`COMPANY_NAME`, `PRODUCTION_DATE`,
`EXPIRATION_DATE`, `PLACE`, `QUANTITY`)
VALUES
('Novalo', 'Bills', 'fsdgjfhjorodsf', 'normal', '3d00', 2, 3, 'Available
for use', 'Med_City', '2017-03-03', '2019-03-03', 'N-Right', 40),
('novafol', 'Bills', 'ftrkl432432md', 'normal', '2xaa', 33, 40,
'Available for use', 'Med_City', '2016-01-01', '2017-01-01', 'N-Left',
27);

```

```

CREATE TABLE `expiry` (
`PRODUCT_NAME` varchar(50) NOT NULL,
`PRODUCT_CODE` varchar(20) NOT NULL,
`DATE_OF_EXPIRY` varchar(10) NOT NULL,
`QUANTITY_REMAIN` int(11) NOT NULL
);

```

```

CREATE TABLE `history_sales` (
`USER_NAME` varchar(20) NOT NULL,
`BARCODE` varchar(20) NOT NULL,
`NAME` varchar(50) NOT NULL,
`TYPE` varchar(10) NOT NULL,
`DOSE` varchar(10) NOT NULL,
`QUANTITY` int(11) NOT NULL,
`PRICE` double NOT NULL,
`AMOUNT` double NOT NULL,
`DATE` varchar(15) NOT NULL,
`TIME` varchar(20) NOT NULL
);

```

```

INSERT INTO `history_sales` (`USER_NAME`, `BARCODE`,
`NAME`, `TYPE`, `DOSE`, `QUANTITY`,
`PRICE`, `AMOUNT`, `DATE`, `TIME`) VALUES
('admin', 'sgnfsjkfnstdjfkfb', 'Breofin', 'Bills', 'Free used', 2, 6, 12,
'12-02-2017', '05:02:06'),
('admin', 'sgnfsjkfnstdjfkfb', 'Breofin', 'Bills', 'Free used', 2, 6, 12,
'12-02-2017', '05:02:26'),
('admin', 'sgnfsjkfnstdjfkfb', 'Breofin', 'Bills', 'Free used', 4, 6, 24,
'12-02-2017', '05:02:40'),

```

('admin', 'nbhdl4978549', 'Morfin', 'Injection', '1 (Day)', 2, 14, 28, '13-02-2017', '01:38:00'),
('admin', 'nbhdl4978549', 'Morfin', 'Injection', '1 (Day)', 2, 14, 28, '13-02-2017', '01:38:10'),
('admin', 'nbhdl4978549', 'Morfin', 'Injection', '1 (Day)', 7, 14, 98, '13-02-2017', '01:38:28'),
('admin', 'nbhdl4978549', 'Morfin', 'Injection', '1 (Day)', 1, 14, 14, '13-02-2017', '01:38:46'),
('mark', 'sgnfsjksfnsdjfkb', 'Breofin', 'Bills', 'Free used', 2, 6, 12, '13-02-2017', '01:59:34'),
('mark', 'sgnfsjksfnsdjfkb', 'Breofin', 'Bills', 'Free used', 5, 6, 30, '13-02-2017', '01:59:43'),
('admin', 'sgnfsjksfnsdjfkb', 'Breofin', 'Bills', 'Free used', 1, 6, 6, '13-02-2017', '02:12:33'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 2, 14, 28, '17-02-2017', '09:55:43'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 2, 14, 28, '17-02-2017', '09:55:58'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 5, 14, 70, '17-02-2017', '09:56:11'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 2, 17, 34, '17-02-2017', '10:04:58'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 2, 17, 34, '17-02-2017', '10:05:15'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 5, 17, 85, '17-02-2017', '10:05:26'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 4, 20, 80, '18-02-2017', '11:16:08'),
('admin', 'fsdjkbdfjkffds', 'Declofien', 'Injection', 'Free', 4, 20, 80, '18-02-2017', '11:16:28'),
('admin', 'AnyBarcodedaf', 'AnyName', 'Drink', '2 Days', 4, 14, 56, '18-02-2017', '11:17:06'),
('admin', 'AnyBarcodedaf', 'AnyName', 'Drink', '2 Days', 4, 14, 56, '18-02-2017', '11:17:15'),
('admin', 'AnyBarcodedaf', 'AnyName', 'Drink', '2 Days', 7, 14, 98, '18-02-2017', '11:17:24'),
('mark', 'AnyBarcodedaf', 'AnyName', 'Drink', '2 Days', 6, 14, 84, '18-02-2017', '11:18:29'),
('mark', 'AnyBarcodedaf', 'AnyName', 'Drink', '2 Days', 2, 14, 28, '18-02-2017', '11:18:41'),

('mark', 'AnyBarcodedaf', 'AnyName', 'Drink', '2 Days', 2, 14, 28,
 '18-02-2017', '11:18:45'),
 ('admin', 'ftrkl432432md', 'novafol', 'Bills', 'normal', 2, 40, 80, '14-
 04-2017', '04:50:32'),
 ('admin', 'ftrkl432432md', 'novafol', 'Bills', 'normal', 2, 40, 80, '14-
 04-2017', '04:50:53'),
 ('admin', 'ftrkl432432md', 'novafol', 'Bills', 'normal', 6, 40, 240, '14-
 04-2017', '04:51:01'),
 ('admin', 'ftrkl432432md', 'novafol', 'Bills', 'normal', 1, 40, 40, '03-
 05-2017', '03:33:30'),
 ('admin', 'ftrkl432432md', 'novafol', 'Bills', 'normal', 1, 40, 40, '03-
 05-2017', '03:33:36'),

('admin', 'ftrkl432432md', 'novafol', 'Bills', 'normal', 1, 40, 40, '03-
 05-2017', '03:33:41');

```

CREATE TABLE `inbox` (
  `MESSAGE_FROM` varchar(20) NOT NULL,
  `MESSAGE_TO` varchar(20) NOT NULL,
  `MESSAGE_TEXT` varchar(200) NOT NULL
);
  
```

```

INSERT INTO `inbox` (`MESSAGE_FROM`, `MESSAGE_TO`,
  `MESSAGE_TEXT`) VALUES
  
```

('admin', 'mark', 'Welcome mark'),
 ('mark', 'admin', 'Welcome sir'),
 ('admin', 'mark', 'Hay , mark '),
 ('mark', 'admin', 'Hay '),
 ('mark', 'admin', 'Welcome'),
 ('admin', 'mark', 'Hello , mark'),
 ('admin', 'mark', 'mark , Please go and update\nthe drug roof '),
 ('admin', 'Tony', 'Welcome'),
 ('admin', 'mark', 'This is your salary on the disk , \n3000 , close in
 12 ; good luck'),
 ('admin', 'mark', 'good job meet me in five'),
 ('mark', 'admin', 'Ok i will '),
 ('admin', 'mark', 'gdfgfdgfdgfdg'),
 ('admin', 'mark', 'haaaaaaaaaaaaaai'),
 ('admin', 'mark', 'What is wrong ? '),
 ('mark', 'admin', 'I am okay thanks ');

```

CREATE TABLE `login` (
  `NAME` varchar(50) NOT NULL,
  `TYPE` varchar(20) NOT NULL,
  `DATE` varchar(20) NOT NULL,
  `TIME` varchar(20) NOT NULL
);
INSERT INTO `login` (`NAME`, `TYPE`, `DATE`, `TIME`)
VALUES
('admin', 'Admin', '17-02-2017', '10:30:24'),
('admin', 'Admin', '17-02-2017', '10:32:48'),
('mark', 'Employee', '17-02-2017', '10:32:56'),
('admin', 'Admin', '17-02-2017', '10:33:10'),
('mark', 'Employee', '17-02-2017', '10:33:37'),
('admin', 'Admin', '17-02-2017', '10:36:21'),
('admin', 'Admin', '17-02-2017', '10:36:53'),
('admin', 'Admin', '17-02-2017', '10:49:27'),
('admin', 'Admin', '17-02-2017', '11:02:23'),
('admin', 'Admin', '17-02-2017', '01:40:08'),
('admin', 'Admin', '18-02-2017', '10:50:29'),
('admin', 'Admin', '18-02-2017', '10:51:50'),
('admin', 'Admin', '18-02-2017', '10:53:33'),
('admin', 'Admin', '18-02-2017', '10:58:41'),
('admin', 'Admin', '18-02-2017', '11:15:39'),
('mark', 'Employee', '18-02-2017', '11:18:19'),
('admin', 'Admin', '18-02-2017', '11:23:25'),
('mark', 'Employee', '18-02-2017', '11:24:19'),
('admin', 'Admin', '04-04-2017', '06:32:57'),
('mark', 'Employee', '04-04-2017', '06:39:00'),
('admin', 'Admin', '13-04-2017', '02:57:26'),
('admin', 'Admin', '13-04-2017', '03:06:11'),
('admin', 'Admin', '13-04-2017', '03:08:31'),
('admin', 'Admin', '13-04-2017', '03:09:40'),
('admin', 'Admin', '13-04-2017', '03:13:24'),
('admin', 'Admin', '13-04-2017', '05:04:26'),
('admin', 'Admin', '13-04-2017', '05:07:20'),
('admin', 'Admin', '13-04-2017', '05:10:11'),
('admin', 'Admin', '13-04-2017', '05:21:53'),

```

('admin', 'Admin', '14-04-2017', '05:11:57'),
('admin', 'Admin', '14-04-2017', '05:13:44'),
('admin', 'Admin', '14-04-2017', '05:17:42'),
('admin', 'Admin', '14-04-2017', '05:19:38'),
('admin', 'Admin', '14-04-2017', '05:22:00'),
('admin', 'Admin', '14-04-2017', '05:28:37'),
('admin', 'Admin', '14-04-2017', '05:30:48'),
('admin', 'Admin', '14-04-2017', '05:35:00'),
('admin', 'Admin', '14-04-2017', '05:39:54'),
('admin', 'Admin', '14-04-2017', '05:41:53'),
('admin', 'Admin', '14-04-2017', '05:44:29'),
('admin', 'Admin', '14-04-2017', '05:47:08'),
('admin', 'Admin', '14-04-2017', '05:48:24'),
('admin', 'Admin', '14-04-2017', '05:49:36'),
('admin', 'Admin', '14-04-2017', '05:51:28'),
('admin', 'Admin', '14-04-2017', '05:53:15'),
('admin', 'Admin', '14-04-2017', '06:22:53'),
('admin', 'Admin', '14-04-2017', '06:30:59'),
('admin', 'Admin', '14-04-2017', '02:32:24'),
('admin', 'Admin', '14-04-2017', '02:40:18'),
('admin', 'Admin', '14-04-2017', '02:43:43'),
('admin', 'Admin', '14-04-2017', '02:46:41'),
('admin', 'Admin', '14-04-2017', '02:48:26'),
('admin', 'Admin', '14-04-2017', '02:49:19'),
('mark', 'Employee', '14-04-2017', '02:52:01'),
('admin', 'Admin', '14-04-2017', '02:58:36'),
('admin', 'Admin', '14-04-2017', '03:14:22'),
('admin', 'Admin', '14-04-2017', '03:17:23'),
('admin', 'Admin', '14-04-2017', '03:19:28'),
('admin', 'Admin', '14-04-2017', '03:27:34'),
('admin', 'Admin', '14-04-2017', '04:49:24'),
('mark', 'Employee', '14-04-2017', '04:55:06'),
('mark', 'Employee', '14-04-2017', '05:01:50'),
('mark', 'Employee', '14-04-2017', '05:03:59'),
('mark', 'Employee', '14-04-2017', '05:14:50'),
('mark', 'Employee', '14-04-2017', '05:17:01'),
('mark', 'Employee', '14-04-2017', '05:17:50'),

('mark', 'Employee', '14-04-2017', '05:21:19'),
('mark', 'Employee', '14-04-2017', '05:23:30'),
('mark', 'Employee', '14-04-2017', '05:26:03'),
('mark', 'Employee', '14-04-2017', '05:28:53'),
('mark', 'Employee', '14-04-2017', '05:32:36'),
('mark', 'Employee', '14-04-2017', '05:35:04'),
('mark', 'Employee', '14-04-2017', '05:37:17'),
('admin', 'Admin', '14-04-2017', '07:19:33'),
('mark', 'Employee', '15-04-2017', '02:03:12'),
('admin', 'Admin', '15-04-2017', '02:47:28'),
('admin', 'Admin', '15-04-2017', '02:56:16'),
('admin', 'Admin', '15-04-2017', '03:06:20'),
('admin', 'Admin', '15-04-2017', '03:36:58'),
('admin', 'Admin', '15-04-2017', '03:42:44'),
('admin', 'Admin', '03-05-2017', '01:23:14'),
('admin', 'Admin', '03-05-2017', '01:51:20'),
('admin', 'Admin', '03-05-2017', '01:52:35'),
('admin', 'Admin', '03-05-2017', '03:31:40'),
('admin', 'Admin', '03-05-2017', '03:47:32'),
('admin', 'Admin', '05-05-2017', '01:24:00'),
('admin', 'Admin', '05-05-2017', '03:06:19'),
('admin', 'Admin', '05-05-2017', '07:54:02'),
('mark', 'Employee', '05-05-2017', '07:55:52'),
('admin', 'Admin', '05-05-2017', '08:01:50'),
('admin', 'Admin', '05-05-2017', '08:02:44'),
('admin', 'Admin', '05-05-2017', '08:05:37'),
('admin', 'Admin', '05-05-2017', '08:07:37'),
('mark', 'Employee', '05-05-2017', '08:09:23'),
('admin', 'Admin', '05-05-2017', '08:14:18'),
('mark', 'Employee', '05-05-2017', '08:15:44'),
('admin', 'Admin', '05-05-2017', '08:16:15'),
('admin', 'Admin', '06-05-2017', '09:51:33'),
('mark', 'Employee', '06-05-2017', '09:52:46'),
('mark', 'Employee', '06-05-2017', '09:54:33'),
('admin', 'Admin', '07-05-2017', '04:44:39'),
('admin', 'Admin', '07-05-2017', '10:02:15'),
('admin', 'Admin', '07-05-2017', '10:12:11'),

('admin', 'Admin', '27-05-2017', '03:53:36'),
('admin', 'Admin', '27-05-2017', '03:54:05'),
('admin', 'Admin', '27-05-2017', '04:05:04'),
('mark', 'Employee', '27-05-2017', '04:06:02'),
('admin', 'Admin', '30-05-2017', '03:13:41'),
('admin', 'Admin', '31-05-2017', '10:57:35'),
('admin', 'Admin', '31-05-2017', '11:00:02'),
('admin', 'Admin', '31-05-2017', '11:06:32'),
('admin', 'Admin', '07-06-2017', '08:38:00'),
('admin', 'Admin', '07-06-2017', '08:40:43'),
('admin', 'Admin', '07-06-2017', '08:41:28'),
('mark', 'Employee', '19-09-2017', '06:10:07'),
('admin', 'Admin', '07-10-2017', '04:39:50'),
('admin', 'Admin', '07-10-2017', '04:40:39'),
('admin', 'Admin', '21-11-2017', '09:06:10'),
('admin', 'Admin', '21-11-2017', '09:15:39'),
('mark', 'Employee', '14-12-2017', '02:56:45'),
('admin', 'Admin', '24-03-2018', '07:20:36'),
('admin', 'Admin', '24-03-2018', '08:47:14'),
('admin', 'Admin', '24-03-2018', '08:51:01'),
('mark', 'Employee', '24-03-2018', '08:52:17'),
('admin', 'Admin', '24-03-2018', '08:52:50');

```
CREATE TABLE `message_history` (  
  `MESSAGE_FROM` varchar(20) NOT NULL,  
  `MESSAGE_TO` varchar(20) NOT NULL,  
  `MESSAGE_TEXT` varchar(200) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE `purchase` (  
  `BARCODE` varchar(20) NOT NULL,  
  `NAME` varchar(50) NOT NULL,  
  `TYPE` varchar(20) NOT NULL,  
  `COMPANY_NAME` varchar(20) NOT NULL,  
  `QUANTITY` int(11) NOT NULL,  
  `PRICE` double NOT NULL,  
  `AMOUNT` double NOT NULL
```



```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
INSERT INTO `purchase` (`BARCODE`, `NAME`, `TYPE`,  
`COMPANY_NAME`, `QUANTITY`, `PRICE`, `AMOUNT`)  
VALUES
```

```
('fsdgjfhjorodsf', 'Novalo', 'Bills', 'Med_City', 40, 2, 80);
```

```
CREATE TABLE `sales` (  
  `BARCODE` varchar(20) NOT NULL,  
  `NAME` varchar(50) NOT NULL,  
  `TYPE` varchar(10) NOT NULL,  
  `DOSE` varchar(10) NOT NULL,  
  `QUANTITY` int(11) NOT NULL,  
  `PRICE` double NOT NULL,  
  `AMOUNT` double NOT NULL,  
  `DATE` varchar(15) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE `users` (  
  `ID` int(11) NOT NULL,  
  `NAME` varchar(50) NOT NULL,  
  `DOB` varchar(20) NOT NULL,  
  `ADDRESS` varchar(100) NOT NULL,  
  `PHONE` varchar(20) NOT NULL,  
  `SALARY` double NOT NULL,  
  `PASSWORD` varchar(20) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
INSERT INTO `users` (`ID`, `NAME`, `DOB`, `ADDRESS`,  
`PHONE`, `SALARY`, `PASSWORD`) VALUES  
(1, 'admin', '23-12-1995', 'Someplace India', '9800000000', 50000,  
'admin'),  
(2, 'mark', '3-2-1972', 'Bangalore India', '01290789432', 2000,  
'mark'),  
(3, 'clark', '3-2-1971', 'Nowhere Earth-616', '01147893423', 4000,  
'rootaccess');
```

```
(4, 'Tont Stark', '7-8-1977', ' 10880 Malibu Point, Malibu,  
California', '011804368743', 3000,  
'rootaccess');
```

```
ALTER TABLE `company`  
  ADD PRIMARY KEY (`NAME`);  
ALTER TABLE `drugs`  
  ADD PRIMARY KEY (`BARCODE`);  
ALTER TABLE `purchase`  
  ADD PRIMARY KEY (`BARCODE`);  
ALTER TABLE `users`  
  ADD PRIMARY KEY (`ID`);  
ALTER TABLE purchase  
  add constraint fkr3  
  foreign key (company_name)  
  references company(name)  
  on update cascade  
  on delete cascade;
```

2.Queries on HR Dataset:

Name: SHAIK JULFEEN AHMADH

Reg.No: 12110554

The HR database is a sample database that was originally created by Microsoft and used as the basis for their tutorials in a variety of database products for decades.

The HR sample database has seven tables:

1. The **employees** table stores the data of employees.
2. The **jobs** table stores the job data including job title and salary range.
3. The **departments** table stores department data.
4. The **job history** table stores the job history of employees.
5. The **locations** table stores the location of the departments of the company.

6. The **countries** table stores the data of countries where the company is doing business.

7. The **regions** table stores the data of regions such as Asia, Europe, America, and the Middle East and Africa. The countries are grouped into regions.

Data Set Link - <https://www.kaggle.com/datasets/sirajahmad/hr-schema-mysql>

Tasks and answers

1) Write a query to find the addresses (location_id, street_address, city, state_province, country_name) of all the departments

ANS:

SELECT

l.location_id , l.street_address,

l.city , l.state_province , c.country_name

FROM departments d

JOIN locations l ON d.location_id = l.location_id

JOIN countries c ON l.country_id = c.country_id;

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
/* 1) Write a query to find the addresses (location_id, street_address, city, state_province, country_name)
of all the departments */
SELECT l.location_id, l.street_address, l.city, l.state_province, c.country_name
FROM departments d
JOIN locations l ON d.location_id = l.location_id
JOIN countries c ON l.country_id = c.country_id;
```

The Results Grid shows the following data:

location_id	street_address	city	state_province	country_name
1400	2014 Jabberwocky Rd	Southlake	Texas	United States of America
1500	2011 Interiors Blvd	South San Francisco	California	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America
1700	2004 Charade Rd	Seattle	Washington	United States of America

The Output pane shows the following message:

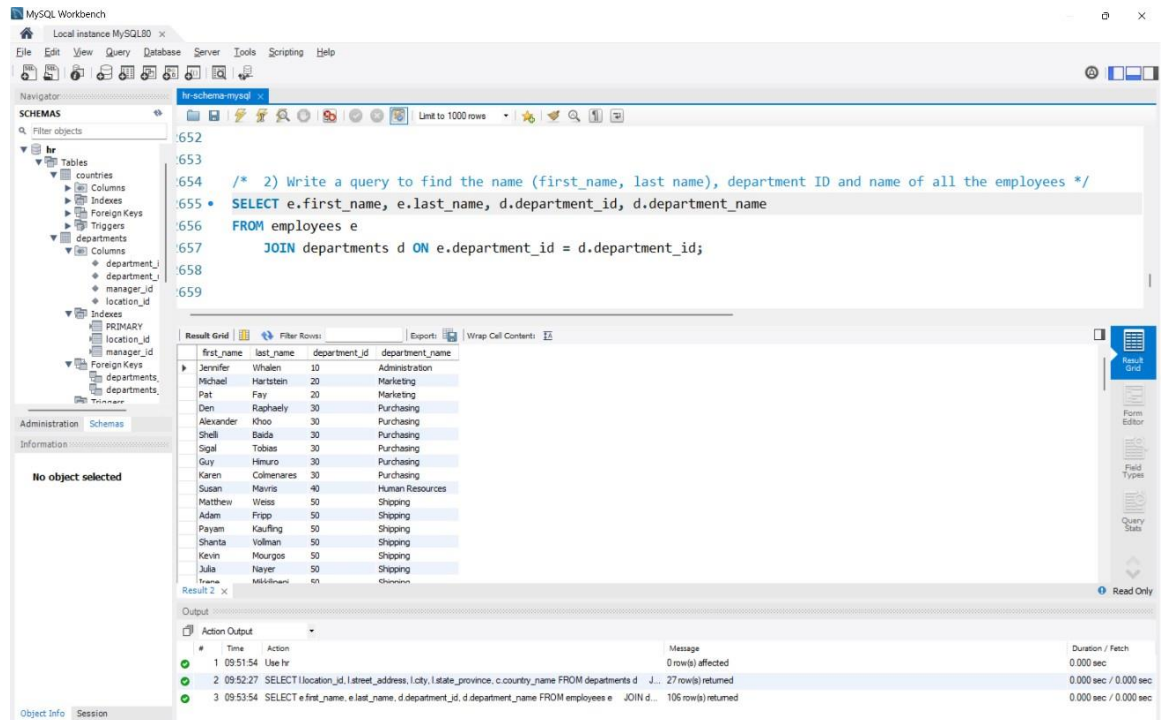
```
1 09:51:54 Use hr
2 09:52:27 SELECT l.location_id, l.street_address, l.city, l.state_province, c.country_name FROM departments d JOIN locations l ON d.location_id = l.location_id JOIN countries c ON l.country_id = c.country_id; 27 row(s) returned
```

2) Write a query to find the name (first_name, last name), department ID and name of all the employees

ANS:

SELECT e.first_name, e.last_name, d.department_id, d.department_name FROM employees e

JOIN departments d ON e.department_id = d.department_id;



3) Write a query to find the name (first_name, last_name), job, department ID and name of the employees who works in London

ANS:

SELECT

```

e.first_name, e.last_name,
j.job_title, e.department_id,
d.department_name

```

FROM employees e

```

JOIN jobs j ON e.job_id = j.job_id

```

```

JOIN departments d ON e.department_id = d.department_id JOIN

```

```

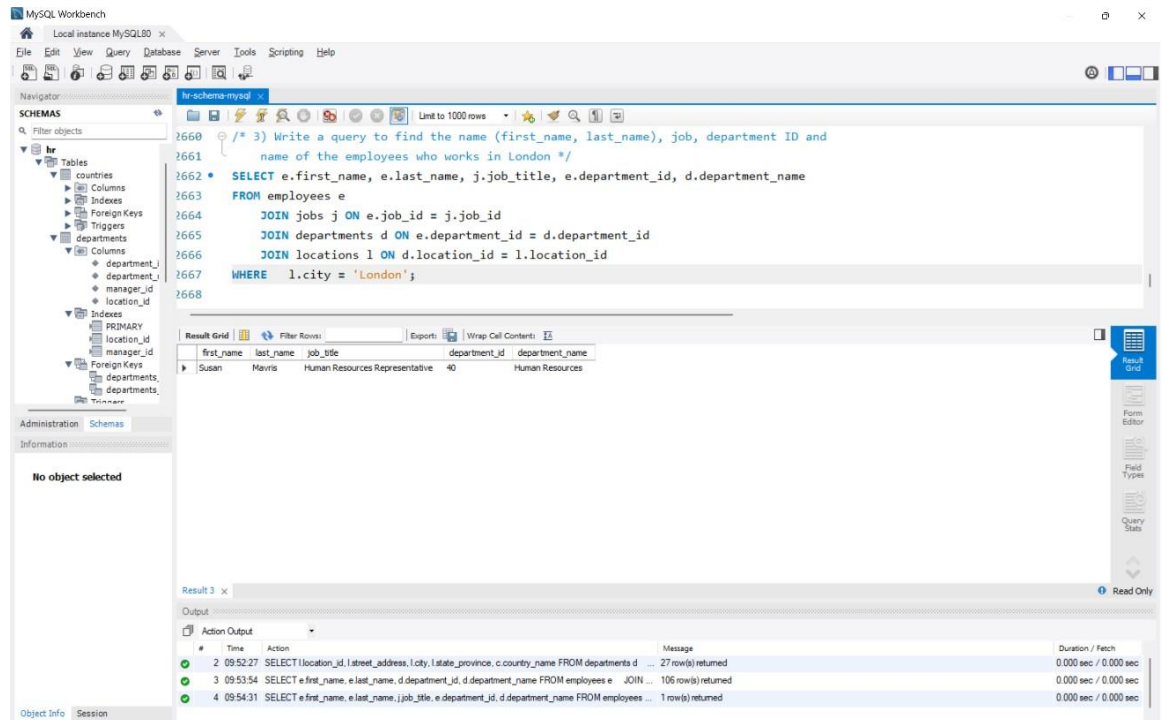
locations l ON d.location_id = l.location_id

```

```

WHERE l.city = 'London';

```



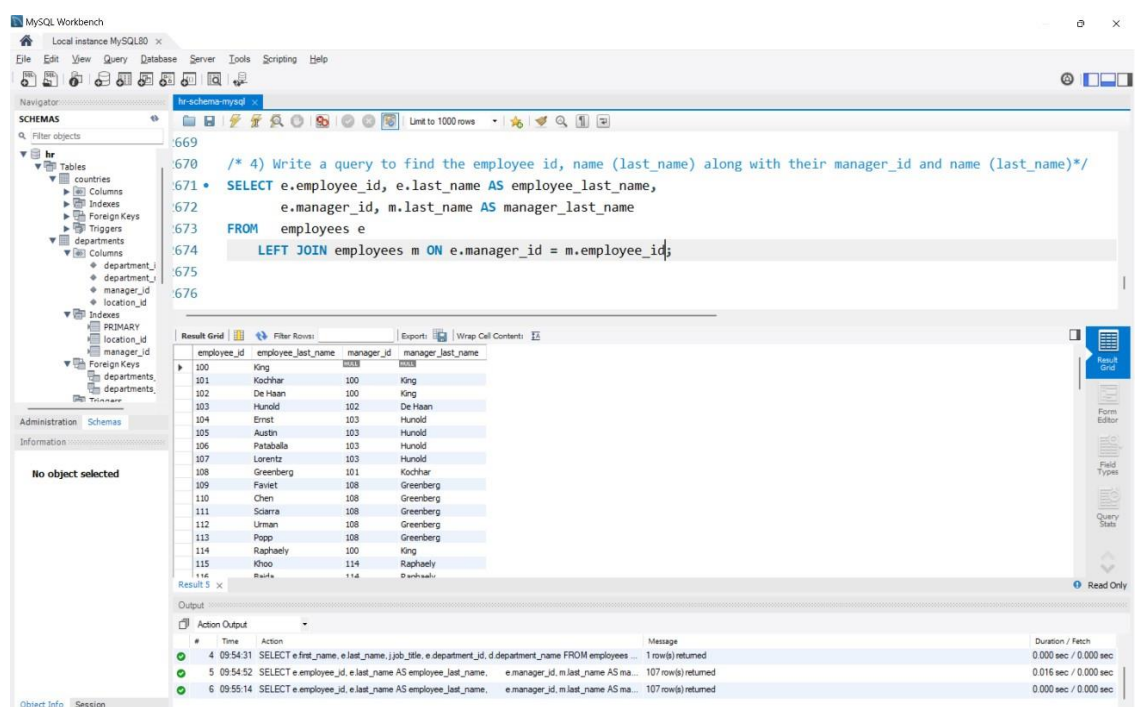
4) Write a query to find the employee id, name (last_name) along with their manager_id and name (last_name)

ANS:

```
SELECT e.employee_id, e.last_name AS employee_last_name, e.manager_id,
m.last_name AS manager_last_name
```

```
FROM employees e
```

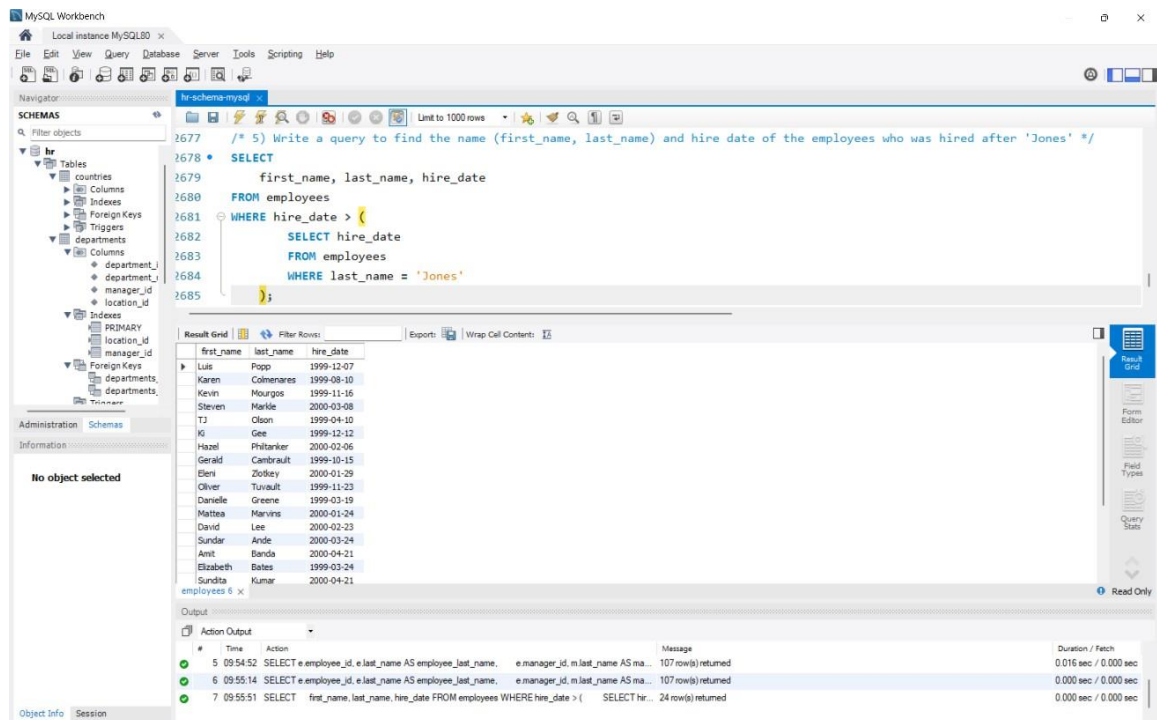
```
LEFT JOIN employees m ON e.manager_id = m.employee_id;
```



5) Write a query to find the name (first_name, last_name) and hire date of the employees who was hired after 'Jones'

ANS:

```
SELECT first_name, last_name, hire_date
FROM employees
WHERE hire_date > (
    SELECT hire_date
    FROM employees
    WHERE last_name = 'Jones'
);
```

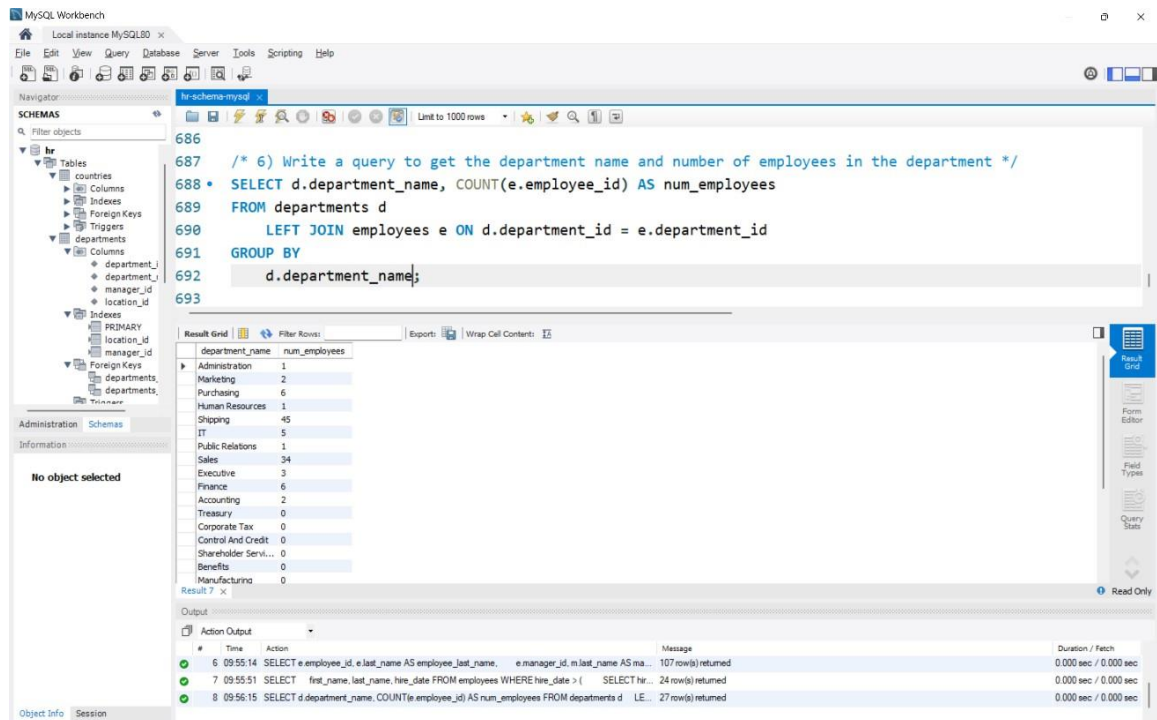


6) Write a query to get the department name and number of employees in the department

ANS:

```
SELECT
    d.department_name, COUNT(e.employee_id) AS num_employees
FROM departments d
JOIN employees e ON d.department_id = e.department_id
```

```
LEFT JOIN employees e ON d.department_id = e.department_id GROUP BY
d.department_name;
```



7) Write a query to display department name, name (first_name, last_name), hire date, salary of the manager for all managers whose experience is more than 15 years

ANS:

```

SELECT

    d.department_name, e.first_name,
    e.last_name, e.hire_date, e.salary

FROM employees e

JOIN departments d ON e.department_id = d.department_id WHERE

    e.employee_id IN (

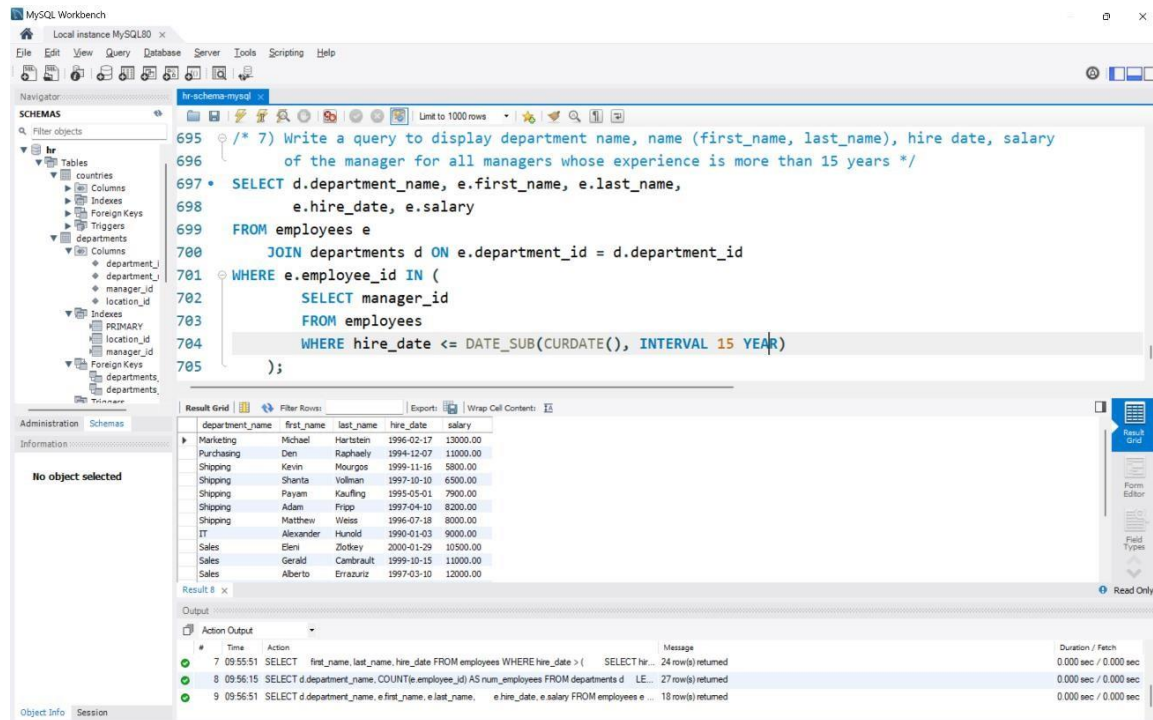
        SELECT manager_id

        FROM employees

        WHERE hire_date <= DATE_SUB(CURDATE(), INTERVAL 15 YEAR)

    );

```

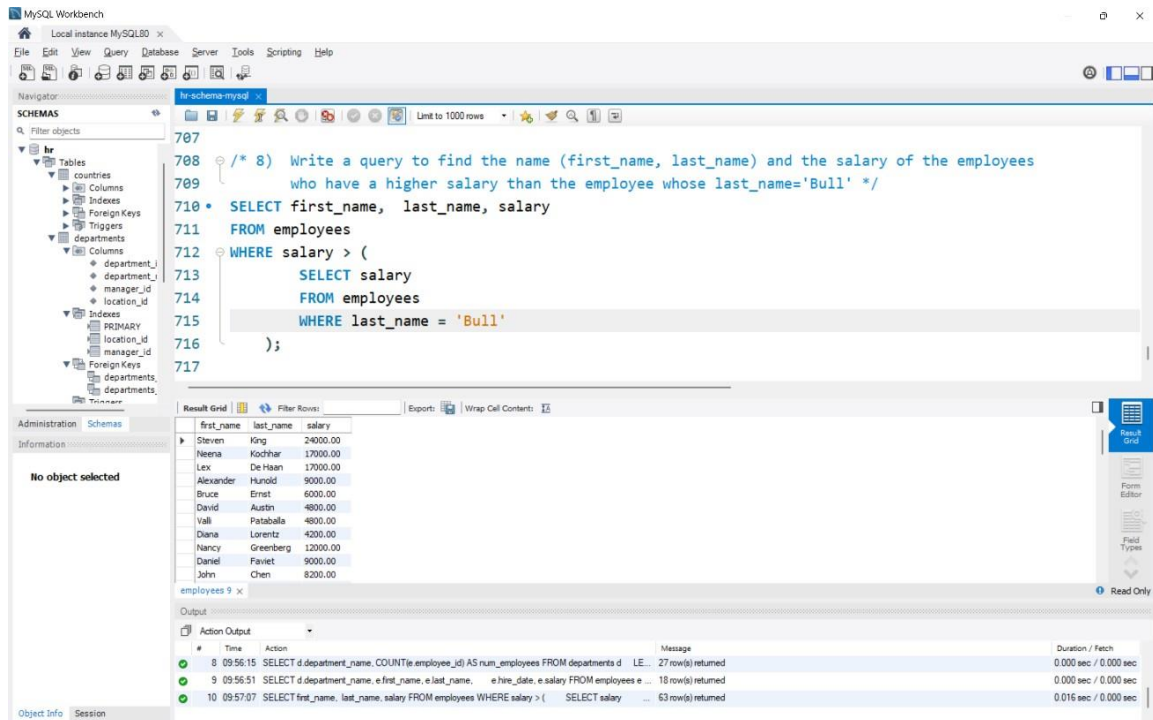
8) Write a query to find the name (first_name, last_name) and the salary of the employees who have a higher salary than the employee whose last_name='Bull'

ANS:

```

SELECT first_name, last_name, salary
FROM employees
WHERE salary > (
    SELECT salary
    FROM employees
    WHERE last_name = 'Bull'
);

```



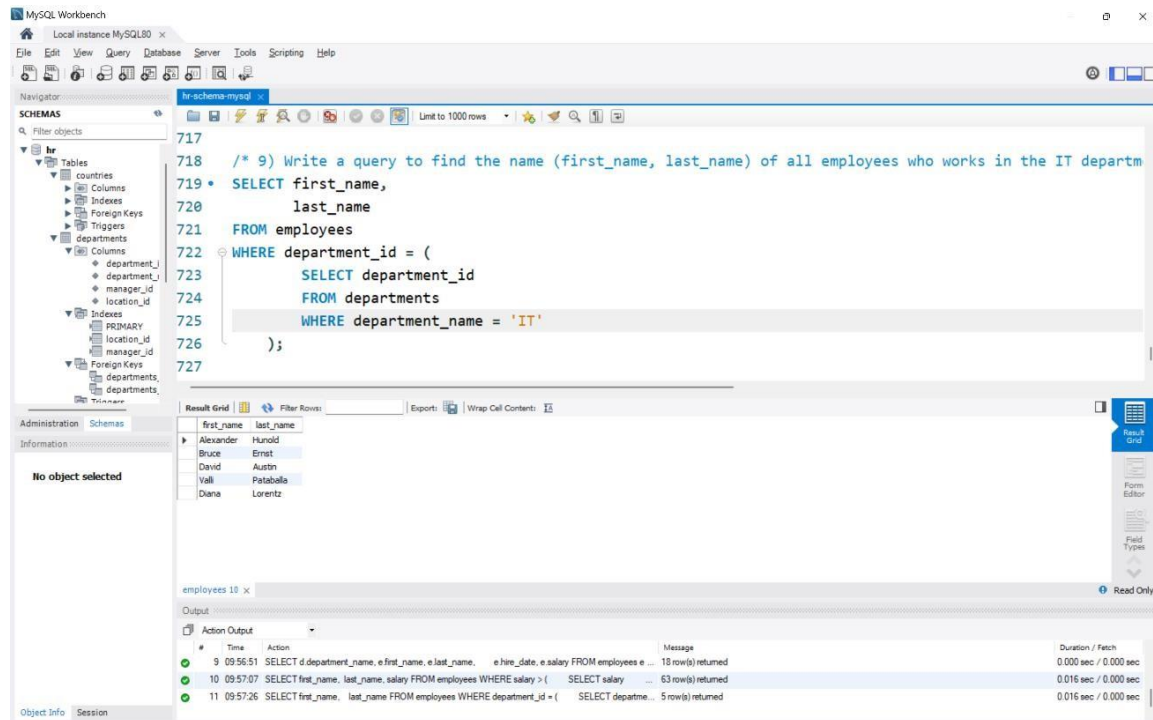
9) Write a query to find the name (first_name, last_name) of allemployees who works in the IT department

ANS:

```

SELECT first_name, last_name
FROM employees
WHERE department_id = (
    SELECT department_id
    FROM departments
    WHERE department_name = 'IT'
);

```



10) Write a query to find the name (first_name, last_name) of the employees who have a manager and worked in a USA based department

ANS:

```
SELECT e.first_name, e.last_name
```

```
FROM employees e
```

```
JOIN departments d ON e.department_id = d.department_id WHERE
```

```
e.manager_id IS NOT NULL
```

```
AND d.location_id IN (
```

```
SELECT location_id
```

```
FROM locations WHERE
```

```
country_id =
```

```
(
```

```
SELECT country_id
```

```
FROM countries
```

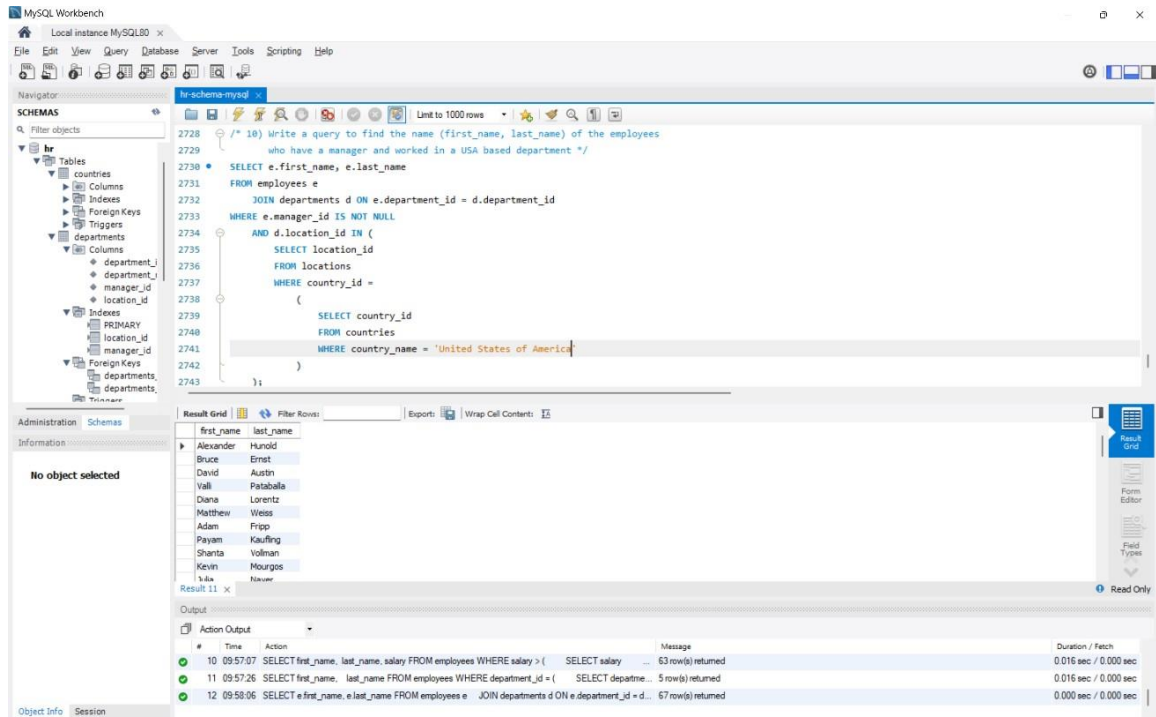
```

WHERE country_name = 'United States of America'

)

);

```



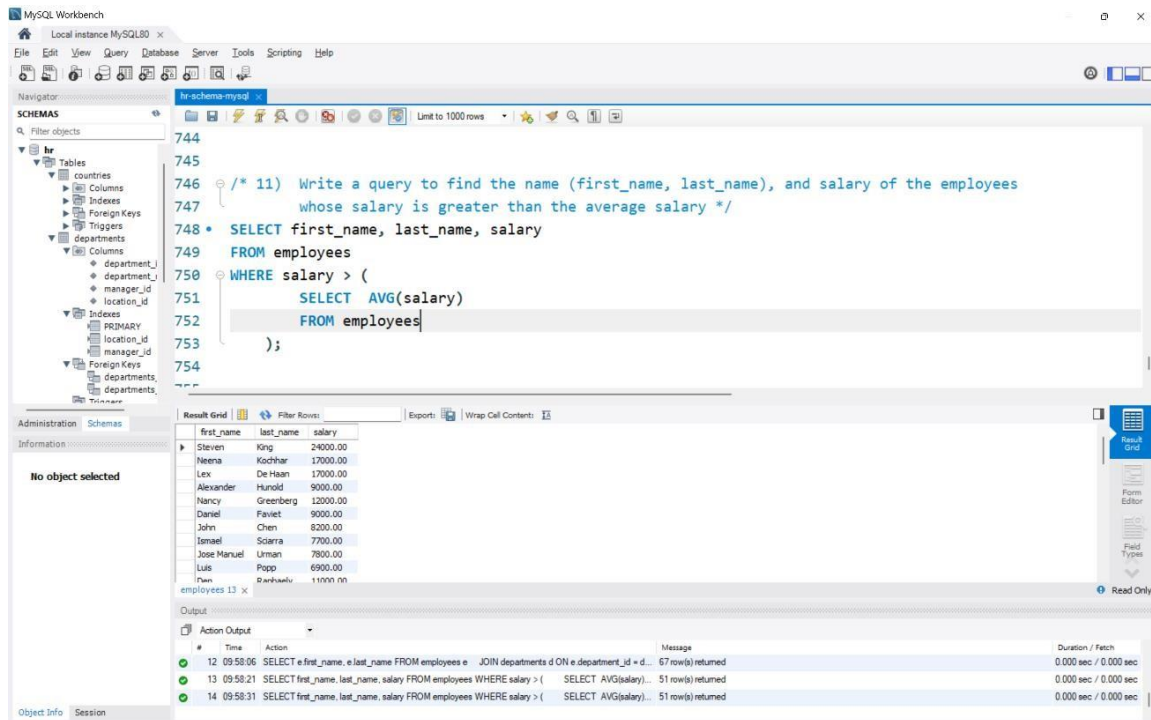
11) Write a query to find the name (first_name, last_name), and salary of the employees whose salary is greater than the average salary

ANS:

```

SELECT first_name, last_name, salary FROM
employees
WHERE salary > ( SELECT
    AVG(salary) FROM
    employees
);

```



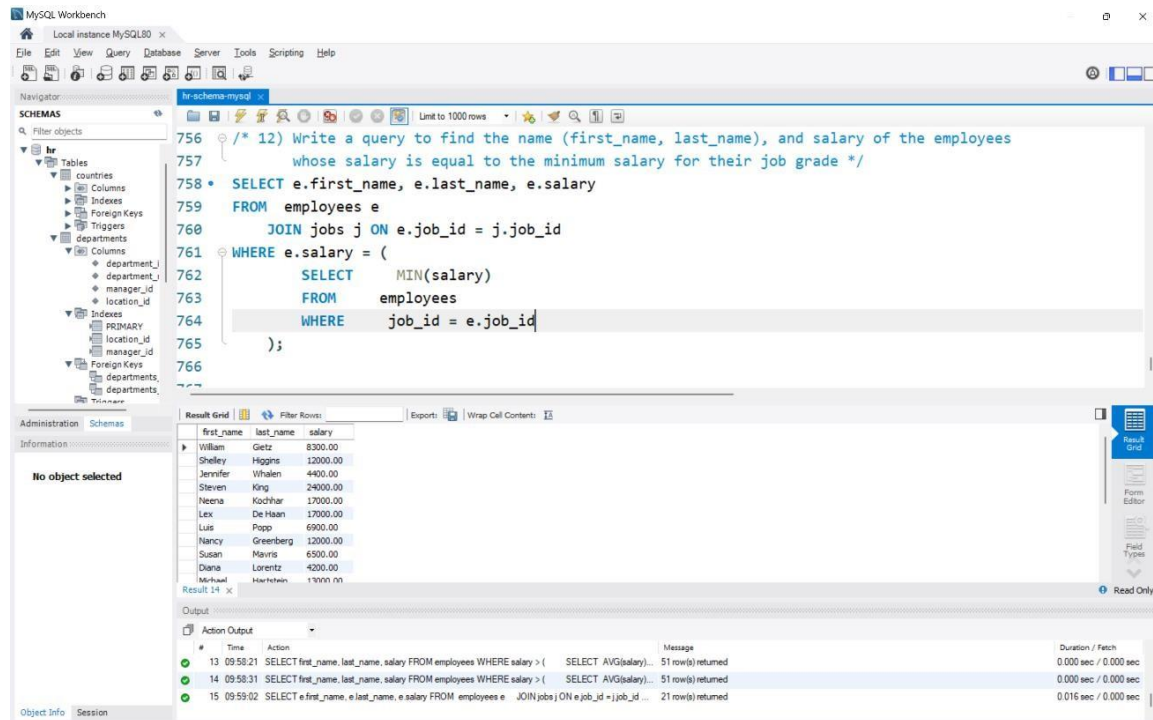
12) Write a query to find the name (first_name, last_name), and salary of the employees whose salary is equal to the minimum salary for their job grade

ANS:

```

SELECT e.first_name, e.last_name, e.salary
FROM employees e
JOIN jobs j ON e.job_id = j.job_id WHERE
e.salary = (
    SELECT MIN(salary) FROM
    employees WHERE
    job_id =
    e.job_id
);

```



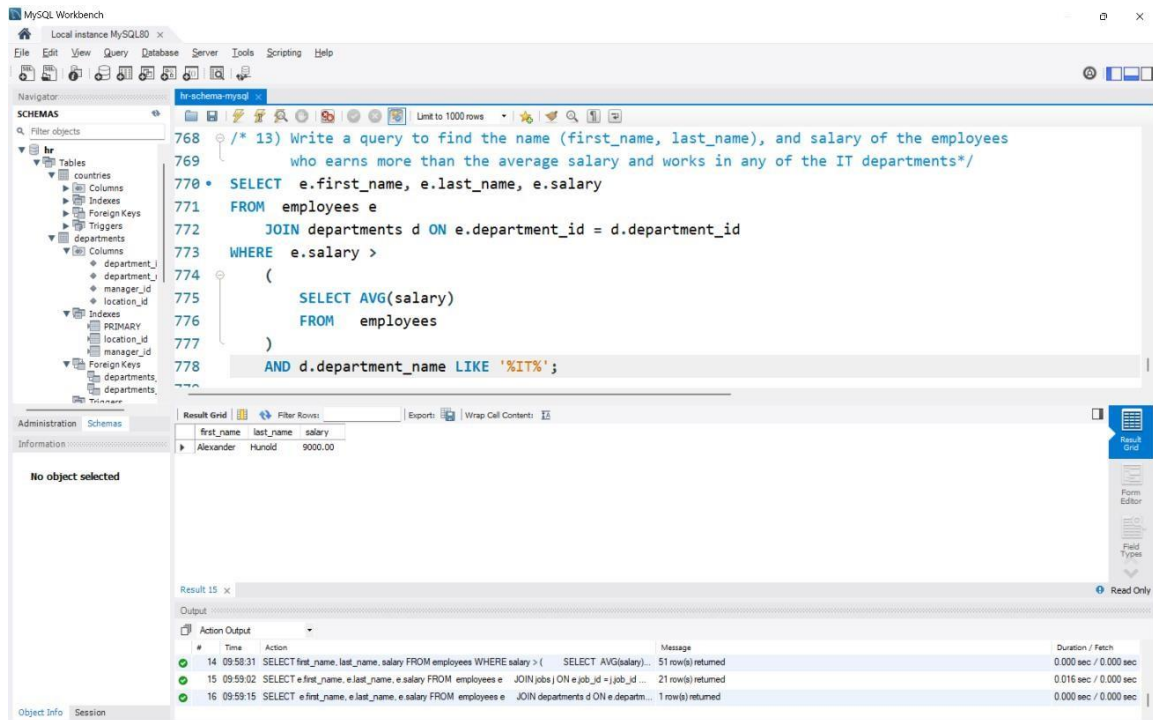
13) Write a query to find the name (first_name, last_name), and salary of the employees who earns more than the average salary and works in any of the IT departments

ANS:

```

SELECT e.first_name, e.last_name, e.salary
FROM employees e
JOIN departments d ON e.department_id = d.department_id WHERE
e.salary >
(
    SELECT AVG(salary)
    FROM employees
)
AND d.department_name LIKE '%IT%';

```



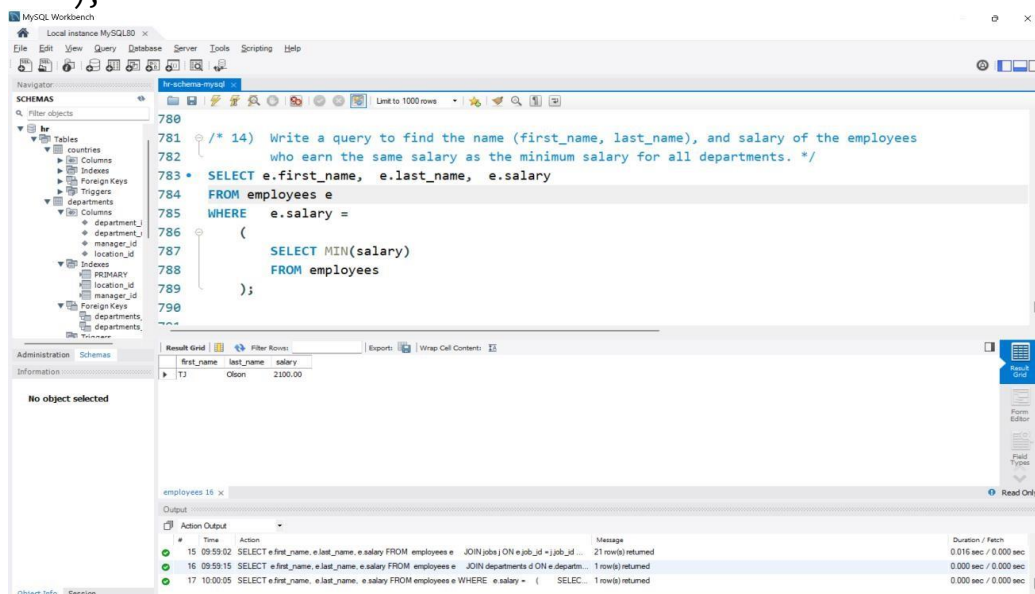
14) Write a query to find the name (first_name, last_name), and salary of the employees who earn the same salary as the minimum salary for all departments.

ANS:

```

SELECT e.first_name, e.last_name, e.salary
FROM employees e
WHERE e.salary = (
    SELECT MIN(salary)
    FROM employees
);

```



15) Write a query to find the name (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = 'SH_CLERK'). Sort the results of the salary of the lowest to highest

ANS:

```
SELECT e.first_name, e.last_name, e.salary
FROM employees e
WHERE e.salary >
(
    SELECT MAX(salary)
    FROM employees
    WHERE job_id = 'SH_CLERK'
)
ORDER BY e.salary ASC;
```

The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL query:

```
/* 15) Write a query to find the name (first_name, last_name) and salary of the employees who earn a salary that is higher than the salary of all the Shipping Clerk (JOB_ID = 'SH_CLERK'). Sort the results of the salary of the lowest to highest */
SELECT e.first_name, e.last_name, e.salary
FROM employees e
WHERE e.salary >
(
    SELECT MAX(salary)
    FROM employees
    WHERE job_id = 'SH_CLERK'
)
ORDER BY e.salary ASC;
```

The Results Grid shows the following data:

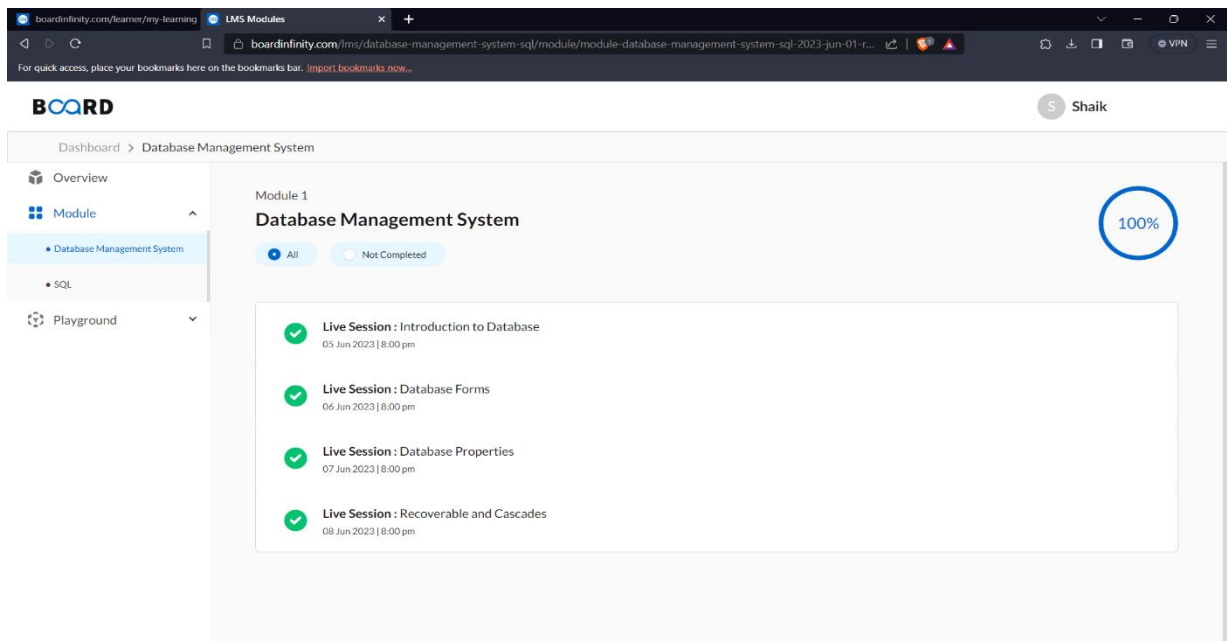
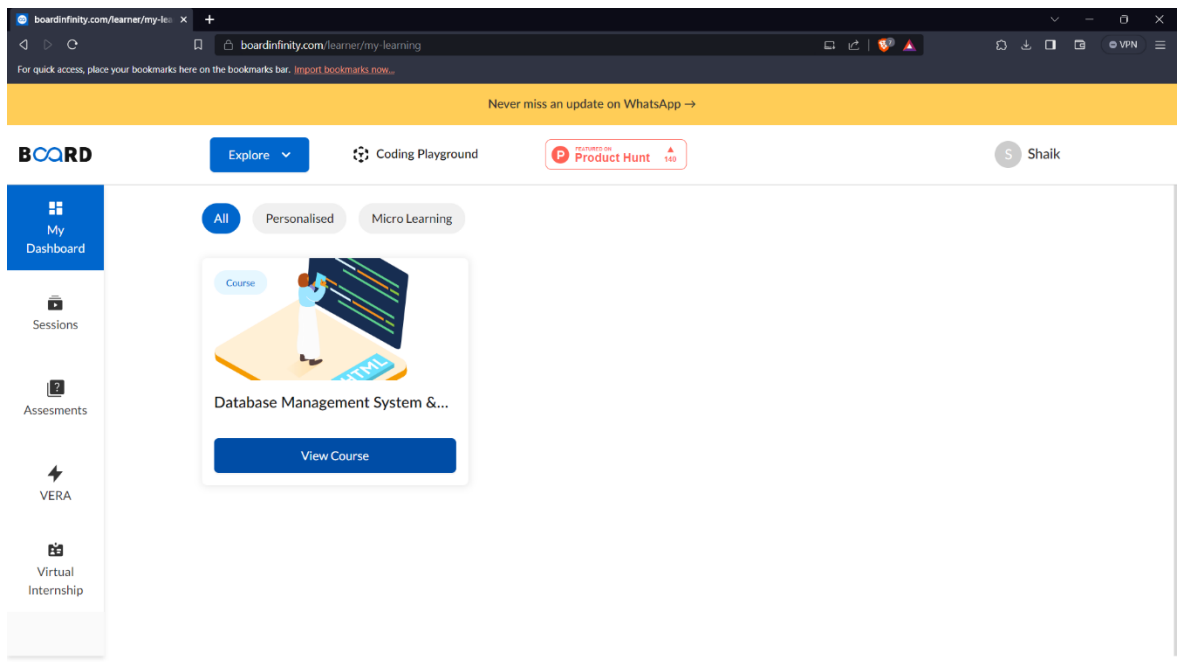
first_name	last_name	salary
Jennifer	Whalen	4400.00
David	Austin	4800.00
Valli	Potaballa	4800.00
Kevin	Hourges	5800.00
Bruce	Ernst	6000.00
Pat	Fay	6000.00
Sundita	Kumar	6100.00
Amit	Banda	6200.00
Charles	Johnson	6200.00
Sundar	Aide	6400.00
Ghanshi	Vollenwe	6400.00

The Action Output pane shows the execution of the query:

#	Time	Action	Message	Duration / Fetch
16	09:59:15	SELECT e.first_name, e.last_name, e.salary FROM employees e JOIN departments d ON e.departm...	1 row(s) returned	0.000 sec / 0.000 sec
17	10:00:05	SELECT e.first_name, e.last_name, e.salary FROM employees e WHERE e.salary = (SELECT ...	1 row(s) returned	0.000 sec / 0.000 sec
18	10:00:48	SELECT e.first_name, e.last_name, e.salary FROM employees e WHERE e.salary > (SELECT ...	61 row(s) returned	0.000 sec / 0.000 sec

Learning: I learned how to use queries in Sql . Modify databases using SQL . Understood about basic introduction about data analysis.

BOARD INFINITY DASHBOARD IMAGE



boardinfinity.com/learner/my-learning LMS Modules

boardinfinity.com/lms/database-management-system-sql/module/module-database-management-system-sql-2023-jun-01-r... import bookmarks now...

For quick access, place your bookmarks here on the bookmarks bar. [import bookmarks now...](#)

BOARD Shaik

Dashboard > SQL

Overview

Module

- Database Management System
- SQL

Playground

Module 2

SQL

92.3%

All Assessments Live Session Not Completed

- Live Session : Introduction to SQL
09 Jun 2023 | 8:00 pm
- Live Session : Basics of Excel
10 Jun 2023 | 8:00 pm
- Live Session : Filtering Data using SQL
12 Jun 2023 | 8:00 pm
- Live Session : Functions in Database
13 Jun 2023 | 8:00 pm
- Live Session : Functions in Database
14 Jun 2023 | 8:00 pm
- Live Session : Joins and it's Types

boardinfinity.com/learner/my-learning LMS Modules

boardinfinity.com/lms/database-management-system-sql/module/module-database-management-system-sql-2023-jun-01-r... import bookmarks now...

For quick access, place your bookmarks here on the bookmarks bar. [import bookmarks now...](#)

BOARD Shaik

Dashboard > SQL

Overview

Module

- Database Management System
- SQL

Playground

Module 2

SQL

92.3%

All Assessments Live Session Not Completed

- Live Session : Overview of SQL
24 Jun 2023 | 8:00 pm
- Project : Database Management & SQL Project
24 Jun 2023 | 10:00 am

< 1 2 >

CONCLUSION

In conclusion, a Database Management System (DBMS) stands as a cornerstone in modern information technology, facilitating the storage, organization, retrieval, and manipulation of data with efficiency, security, and reliability. SQL (Structured Query Language) plays an integral role within this framework, serving as a standardized language for interacting with databases.

The significance of a well-implemented DBMS lies in its ability to streamline data management processes, enhance data integrity, and provide a scalable foundation for applications across various industries. Through the utilization of SQL, users can effectively communicate with databases, performing operations like data insertion, retrieval, update, and deletion, as well as complex tasks like data transformation and aggregation.

Furthermore, the evolution of DBMS technology has given rise to various models, such as relational, NoSQL, and NewSQL databases, each catering to specific needs and scenarios. SQL, though commonly associated with relational databases, has adapted to these changes, enabling users to work with diverse database systems.

However, as with any technology, challenges persist. DBMS design requires careful consideration of factors like data modeling, normalization, and indexing to ensure optimal performance. Security concerns, including data breaches and unauthorized access, necessitate robust authentication, authorization, and encryption measures.

In conclusion, the symbiotic relationship between a Database Management System and SQL showcases their enduring importance in data-driven applications. As technology continues to advance, these tools will likely adapt and expand to meet the ever-growing demands of data storage, retrieval, and manipulation, ultimately shaping the future of information management. Whether it's traditional relational databases or newer database paradigms, the foundational concepts of efficient data organization and manipulation through SQL will remain indispensable in the realm of data management.

Throughout our exploration of Database Management Systems (DBMS) and SQL (Structured Query Language), I have gained a comprehensive understanding of the crucial role they play in modern data management. I have learned that a well-designed DBMS serves as the backbone of efficient data storage, retrieval, and manipulation, enabling us to maintain data integrity, ensure security, and streamline operations. Our study of SQL has equipped us with the ability to communicate effectively with databases, allowing us to perform a range of tasks, from basic data retrieval to complex transformations and aggregations. Moreover, we've come to appreciate the adaptability of both DBMS and SQL, as they have evolved to meet the diverse needs of various industries and accommodate different database models. As I conclude our learning journey, we recognize that the principles I have absorbed will continue to be invaluable in shaping the landscape of data-driven applications and technology.

REFERENCES

I referred SQL Queries and Structures from W3Schools for better understanding.

<https://www.w3schools.com/sql/>

for solving the challenges I faced I referred Oracle.

<https://www.oracle.com/>

for video solutions I referred to YouTube.

<https://www.youtube.com/>