

# **LAB 3**

## **DATABASE MANAGEMENT SYSTEM**

**Prof. Dr. Hicham Elmongui**

---

**Asmaa Gamal Abdel-Halem Mabrouk Nagy**  
**Communication & Electronics**

---

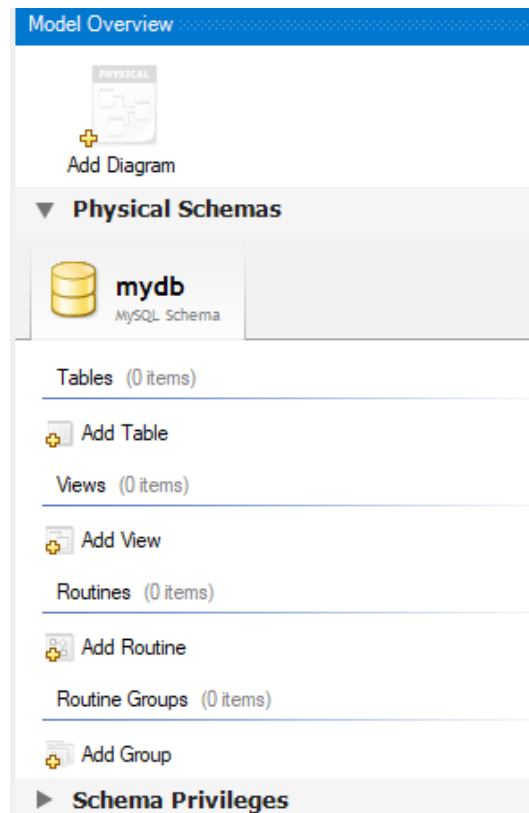
**“Relational Schema” Model using  
MySQL Workbench Tools**



# The Lab Report

- Data Modeling and Schema Generation:

1. Use the data modeler module in the Workbench.



2. Create a new schema; call it **SAMPLEx**.

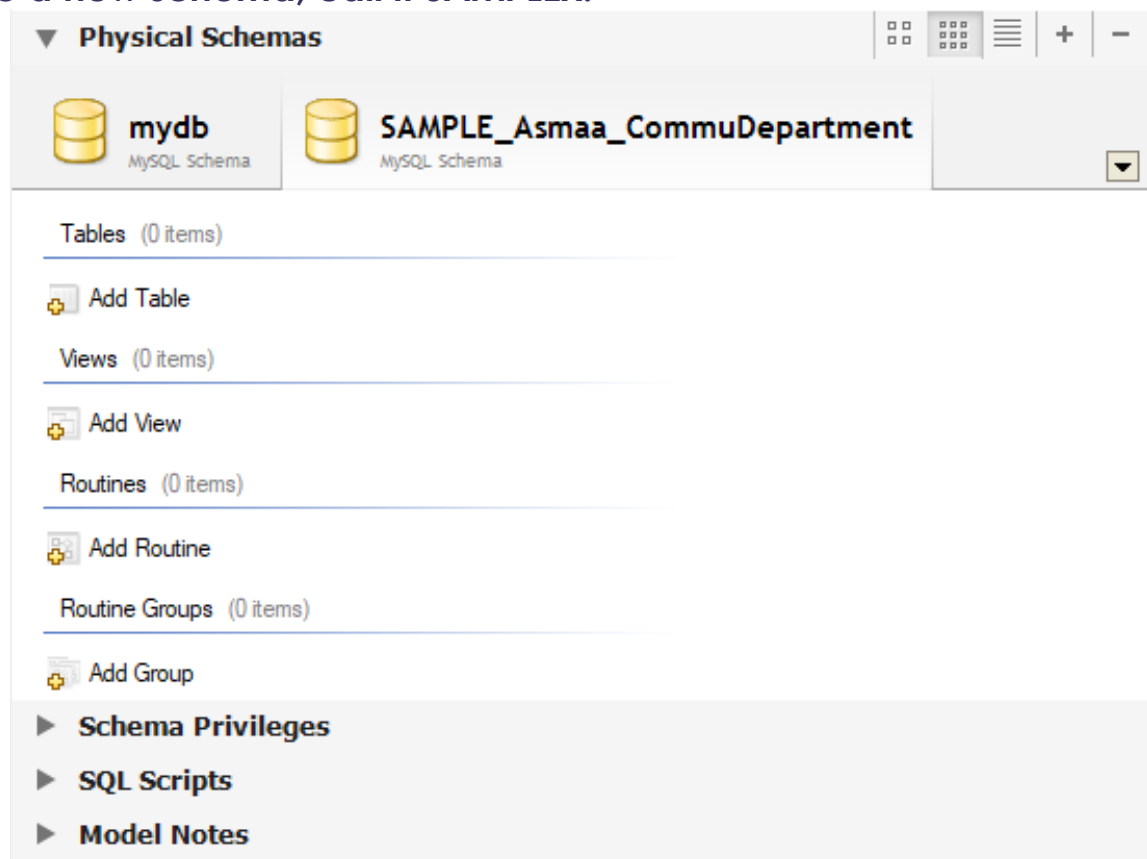




Table Name:  Schema: **SAMPLE\_Asmaa\_CommDepartme**

Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
VARCHAR(45)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
VARCHAR(100)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
VARCHAR(13)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Table Name:  Schema: **SAMPLE\_Asmaa\_CommDepart**

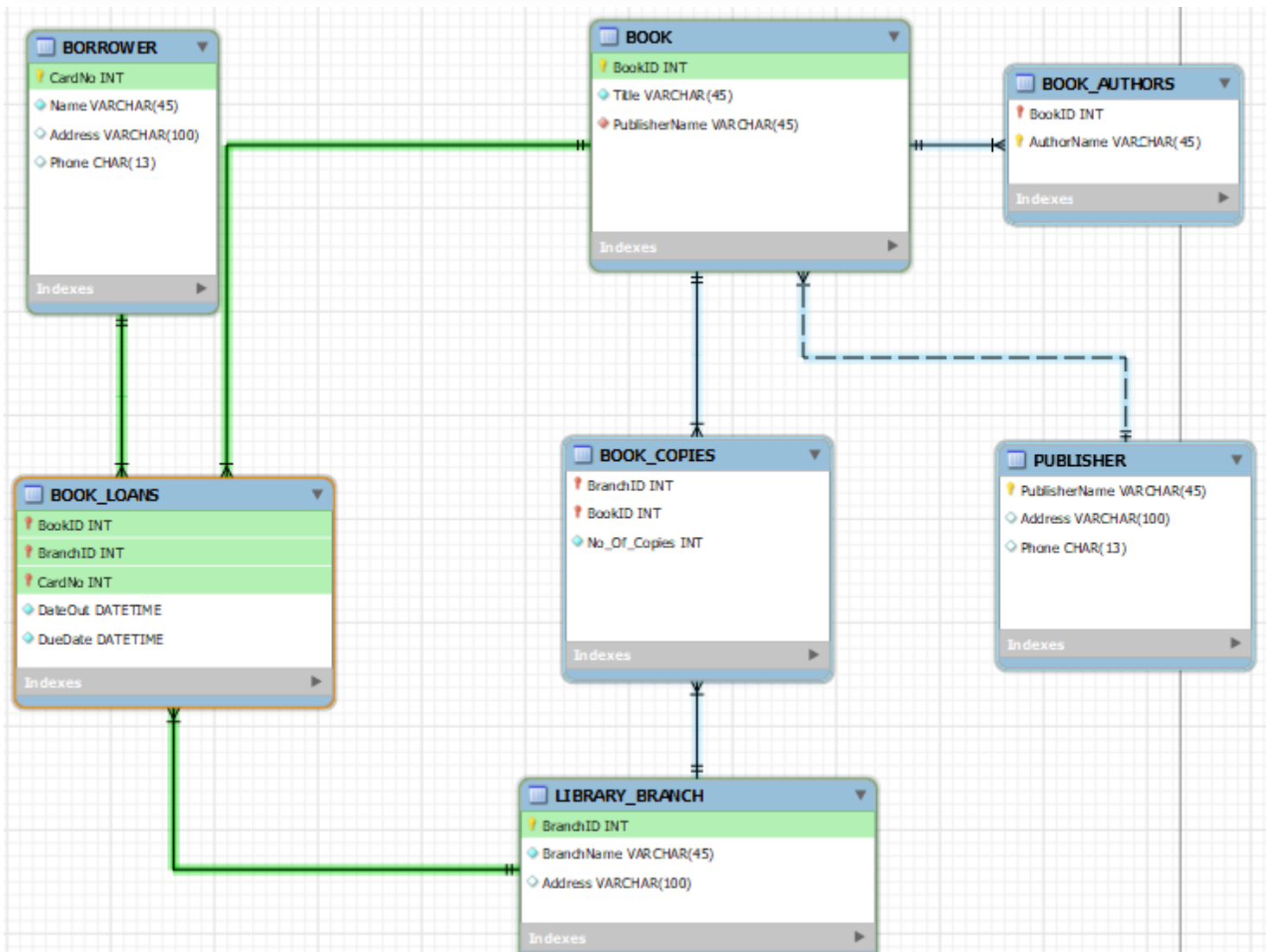
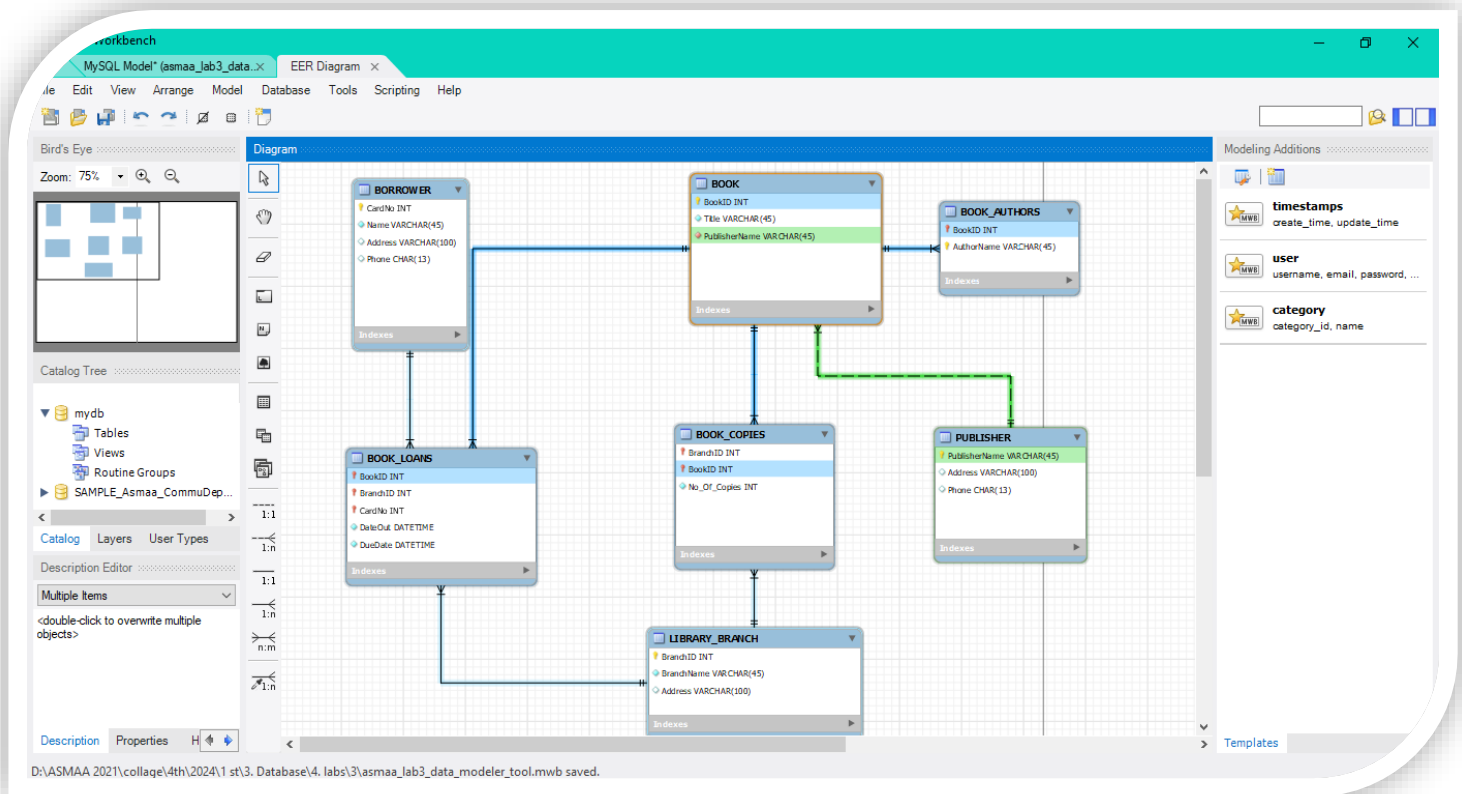
Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
DATETIME	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Table Name:  Schema: **SAMPLE\_Asmaa\_CommDepartment**

Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
BranchID	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
BookID	INT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
No_Of_Copies	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

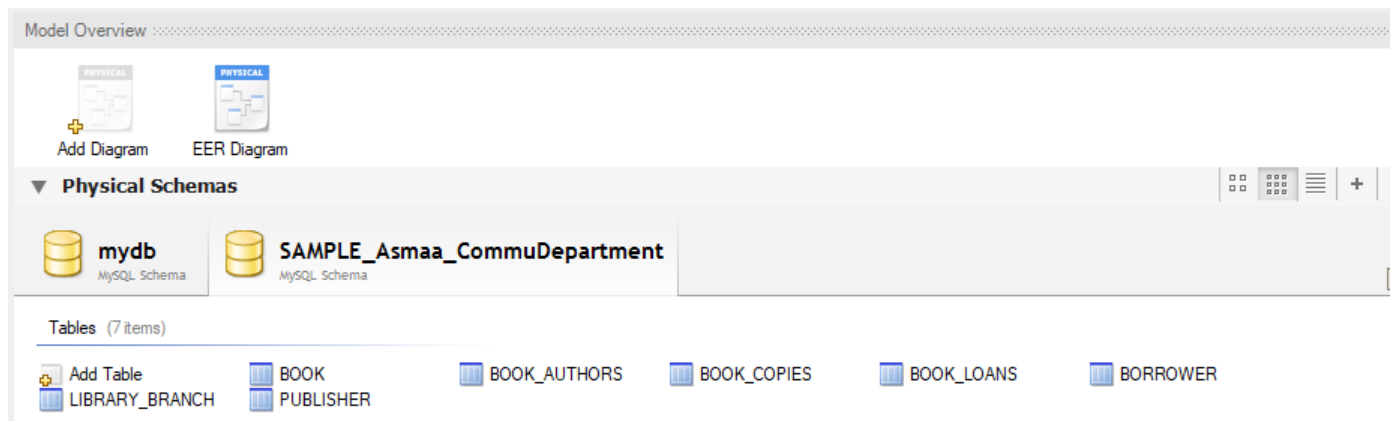


### 3. Total “LIBRARY ERD Model” Solution:

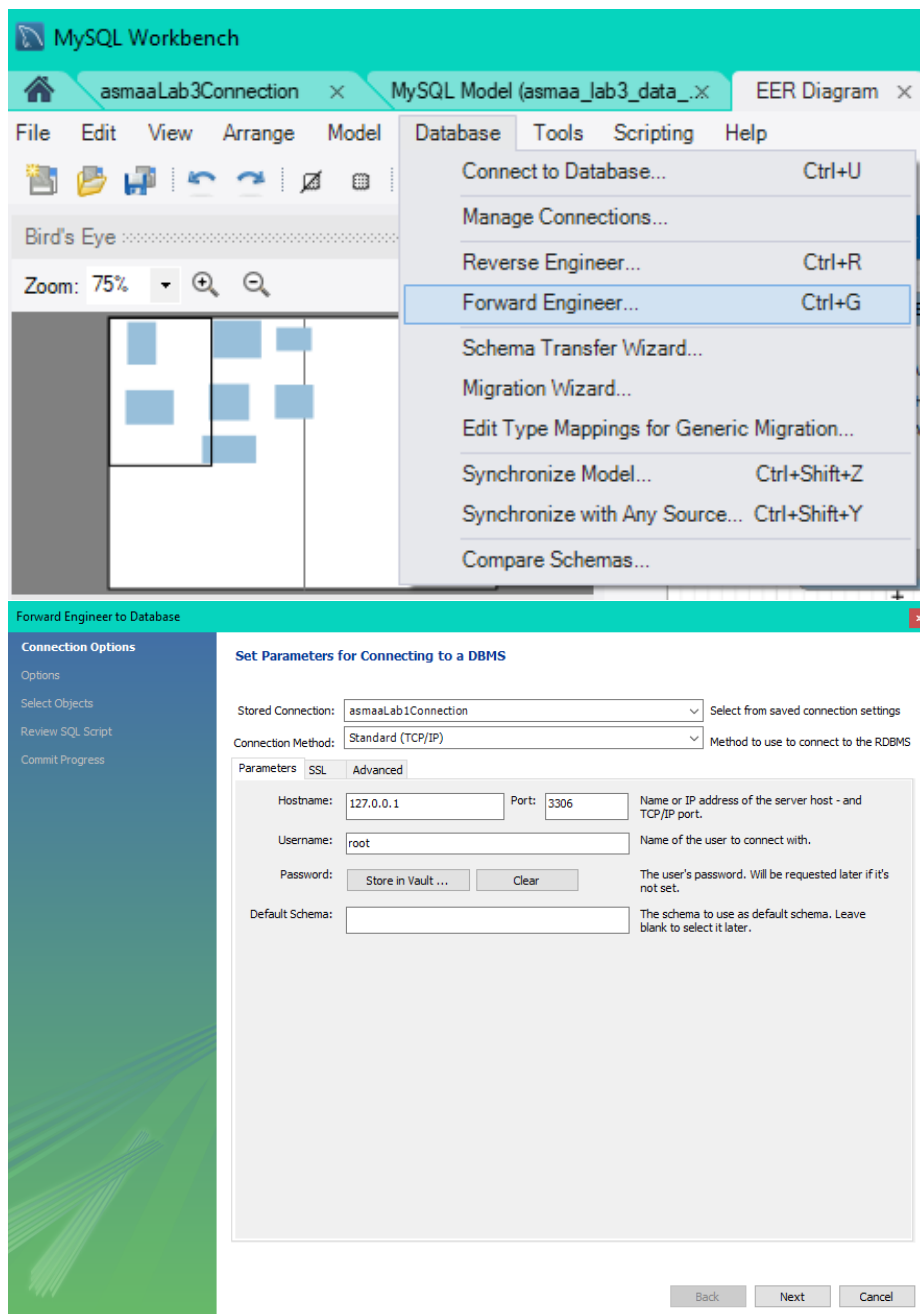


#### 4. Generate the database tables in the schema and the creation DDL scripts.

Here, the 7 tables were generated:



Now, The steps to generate the DDL statements from this ERD:



## Connection Options

## Options

Select Objects

Review SQL Script

Commit Progress

## Set Options for Database to be Created

## Tables

- ☐ Skip creation of FOREIGN KEYS
- ☐ Skip creation of FK Indexes as well
- ☐ Generate separate CREATE INDEX statements
- ☐ Generate INSERT statements for tables
- ☐ Disable FK checks for INSERTs

## Other Objects

- ☐ Don't create view placeholder tables
- ☐ Do not create users. Only create privileges (GRANTS)

## Code Generation

- ☐ DROP objects before each CREATE object
- ☐ Generate DROP SCHEMA
- ☐ Omit schema qualifier in object names
- ☐ Generate USE statements
- ☐ Add SHOW WARNINGS after every DDL statement
- ☒ Include model attached scripts

Back

Next

Cancel

## Connection Options

## Options

## Select Objects

Review SQL Script

Commit Progress

## Select Objects to Forward Engineer

To exclude objects of a specific type from the SQL Export, disable the corresponding checkbox. Press Show Filter and add objects or patterns to the ignore list to exclude them from the export.

☒ Export MySQL Table Objects

7 Total Objects, 7 Selected

Show Filter

☐ Export MySQL View Objects

0 Total Objects, 0 Selected

Show Filter

☐ Export MySQL Routine Objects

0 Total Objects, 0 Selected

Show Filter

☐ Export MySQL Trigger Objects

0 Total Objects, 0 Selected

Show Filter

☐ Export User Objects

0 Total Objects, 0 Selected

Show Filter

Back

Next

Cancel



## Here are the generated DDL scripts:

```
1  -- MySQL Workbench Forward Engineering
2
3  SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
4  SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
5  SET @OLD_SQL_MODE=@@SQL_MODE,
6  SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
7
8  -----
9  -- Schema mydb
10 -----
11 -----
12 -- Schema SAMPLE_Asmaa_CommuneDepartment
13 -----
14
15 -----
16 -- Schema SAMPLE_Asmaa_CommuneDepartment
17 -----
18 CREATE SCHEMA IF NOT EXISTS `SAMPLE_Asmaa_CommuneDepartment` ;
19 USE `SAMPLE_Asmaa_CommuneDepartment` ;
20
21 -----
22 -- Table `SAMPLE_Asmaa_CommuneDepartment`.`PUBLISHER`
23 -----
24 CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_CommuneDepartment`.`PUBLISHER` (
25   `PublisherName` VARCHAR(45) NOT NULL,
26   `Address` VARCHAR(100) NULL,
27   `Phone` VARCHAR(13) NULL,
28   PRIMARY KEY (`PublisherName`),
29
30   ENGINE = InnoDB;
31
32
33 -----
34 -- Table `SAMPLE_Asmaa_CommuneDepartment`.`BOOK`
35 -----
36 CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_CommuneDepartment`.`BOOK` (
37   `BookID` INT NOT NULL AUTO_INCREMENT,
38   `Title` VARCHAR(45) NOT NULL,
39   `PublisherName` VARCHAR(45) NOT NULL,
40   PRIMARY KEY (`BookID`),
41   UNIQUE INDEX `BookID_UNIQUE` (`BookID` ASC) VISIBLE,
42   INDEX `PublisherName_idx` (`PublisherName` ASC) VISIBLE,
43   CONSTRAINT `PublisherName`
44     FOREIGN KEY (`PublisherName`)
45     REFERENCES `SAMPLE_Asmaa_CommuneDepartment`.`PUBLISHER` (`PublisherName`)
46     ON DELETE NO ACTION
47     ON UPDATE NO ACTION)
48   ENGINE = InnoDB;
49
50
51 -----
52 -- Table `SAMPLE_Asmaa_CommuneDepartment`.`BOOK_AUTHORS`
53 -----
54 CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_CommuneDepartment`.`BOOK_AUTHORS` (
```

```

55     `BookID` INT NOT NULL,
56     `AuthorName` VARCHAR(45) NOT NULL,
57     PRIMARY KEY (`AuthorName`, `BookID`),
58     INDEX `fk_BOOK_AUTHORS_BOOK_idx` (`BookID` ASC) VISIBLE,
59     CONSTRAINT `fk_BOOK_AUTHORS_BOOK`
60         FOREIGN KEY (`BookID`)
61         REFERENCES `SAMPLE_Asmas_CommuDepartment`.`BOOK` (`BookID`)
62         ON DELETE NO ACTION
63         ON UPDATE NO ACTION)
64     ENGINE = InnoDB;
65
66
67     -----
68     -- Table `SAMPLE_Asmas_CommuDepartment`.`LIBRARY_BRANCH`
69     -----
70     CREATE TABLE IF NOT EXISTS `SAMPLE_Asmas_CommuDepartment`.`LIBRARY_BRANCH` (
71         `BranchID` INT NOT NULL AUTO_INCREMENT,
72         `BranchName` VARCHAR(45) NOT NULL,
73         `Address` VARCHAR(100) NULL,
74         PRIMARY KEY (`BranchID`),
75         UNIQUE INDEX `BranchID_UNIQUE` (`BranchID` ASC) VISIBLE)
76     ENGINE = InnoDB;
77
78
79     -----
80     -- Table `SAMPLE_Asmas_CommuDepartment`.`BOOK_COPIES`
81     -----
82     CREATE TABLE IF NOT EXISTS `SAMPLE_Asmas_CommuDepartment`.`BOOK_COPIES` (
83         `BranchID` INT NOT NULL,
84         `BookID` INT NOT NULL,
85         `No_Of_Copies` INT ZEROFILL NOT NULL AUTO_INCREMENT,
86         PRIMARY KEY (`BranchID`, `BookID`),
87         CONSTRAINT `BookID`
88             FOREIGN KEY (`BookID`)
89             REFERENCES `SAMPLE_Asmas_CommuDepartment`.`BOOK` (`BookID`)
90             ON DELETE NO ACTION
91             ON UPDATE NO ACTION,
92         CONSTRAINT `BranchID`
93             FOREIGN KEY (`BranchID`)
94             REFERENCES `SAMPLE_Asmas_CommuDepartment`.`LIBRARY_BRANCH` (`BranchID`)
95             ON DELETE NO ACTION
96             ON UPDATE NO ACTION)
97     ENGINE = InnoDB;
98
99
100    -----
101    -- Table `SAMPLE_Asmas_CommuDepartment`.`BORROWER`
102    -----
103    CREATE TABLE IF NOT EXISTS `SAMPLE_Asmas_CommuDepartment`.`BORROWER` (
104        `CardNo` INT NOT NULL AUTO_INCREMENT,
105        `Name` VARCHAR(45) NOT NULL,
106        `Address` VARCHAR(100) NULL,
107        `Phone` VARCHAR(13) NULL,
108        PRIMARY KEY (`CardNo`),

```

```

109     UNIQUE INDEX `CardNo_UNIQUE` (`CardNo` ASC) VISIBLE,
110     UNIQUE INDEX `Phone_UNIQUE` (`Phone` ASC) VISIBLE)
111     ENGINE = InnoDB;
112
113
114     -----
115     -- Table `SAMPLE_Asmas_CommuDepartment`.`BOOK_LOANS`
116     -----
117  CREATE TABLE IF NOT EXISTS `SAMPLE_Asmas_CommuDepartment`.`BOOK_LOANS` (
118     `BookID` INT NOT NULL,
119     `BranchID` INT NOT NULL,
120     `CardNo` INT NOT NULL,
121     `DateOut` DATETIME NOT NULL,
122     `DueDate` DATETIME NOT NULL,
123     INDEX `BookID_idx` (`BookID` ASC) VISIBLE,
124     PRIMARY KEY (`BookID`, `BranchID`, `CardNo`),
125     INDEX `BranchID_idx` (`BranchID` ASC) VISIBLE,
126     INDEX `CardNo_idx` (`CardNo` ASC) VISIBLE,
127     CONSTRAINT `BookID`
128         FOREIGN KEY (`BookID`)
129         REFERENCES `SAMPLE_Asmas_CommuDepartment`.`BOOK` (`BookID`)
130         ON DELETE NO ACTION
131         ON UPDATE NO ACTION,
132     CONSTRAINT `BranchID`
133         FOREIGN KEY (`BranchID`)
134         REFERENCES `SAMPLE_Asmas_CommuDepartment`.`LIBRARY_BRANCH` (`BranchID`)
135         ON DELETE NO ACTION
136         ON UPDATE NO ACTION,
137     CONSTRAINT `CardNo`
138         FOREIGN KEY (`CardNo`)
139         REFERENCES `SAMPLE_Asmas_CommuDepartment`.`BORROWER` (`CardNo`)
140         ON DELETE NO ACTION
141         ON UPDATE NO ACTION)
142     ENGINE = InnoDB;
143
144
145     SET SQL_MODE=@OLD_SQL_MODE;
146     SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
147     SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
148

```

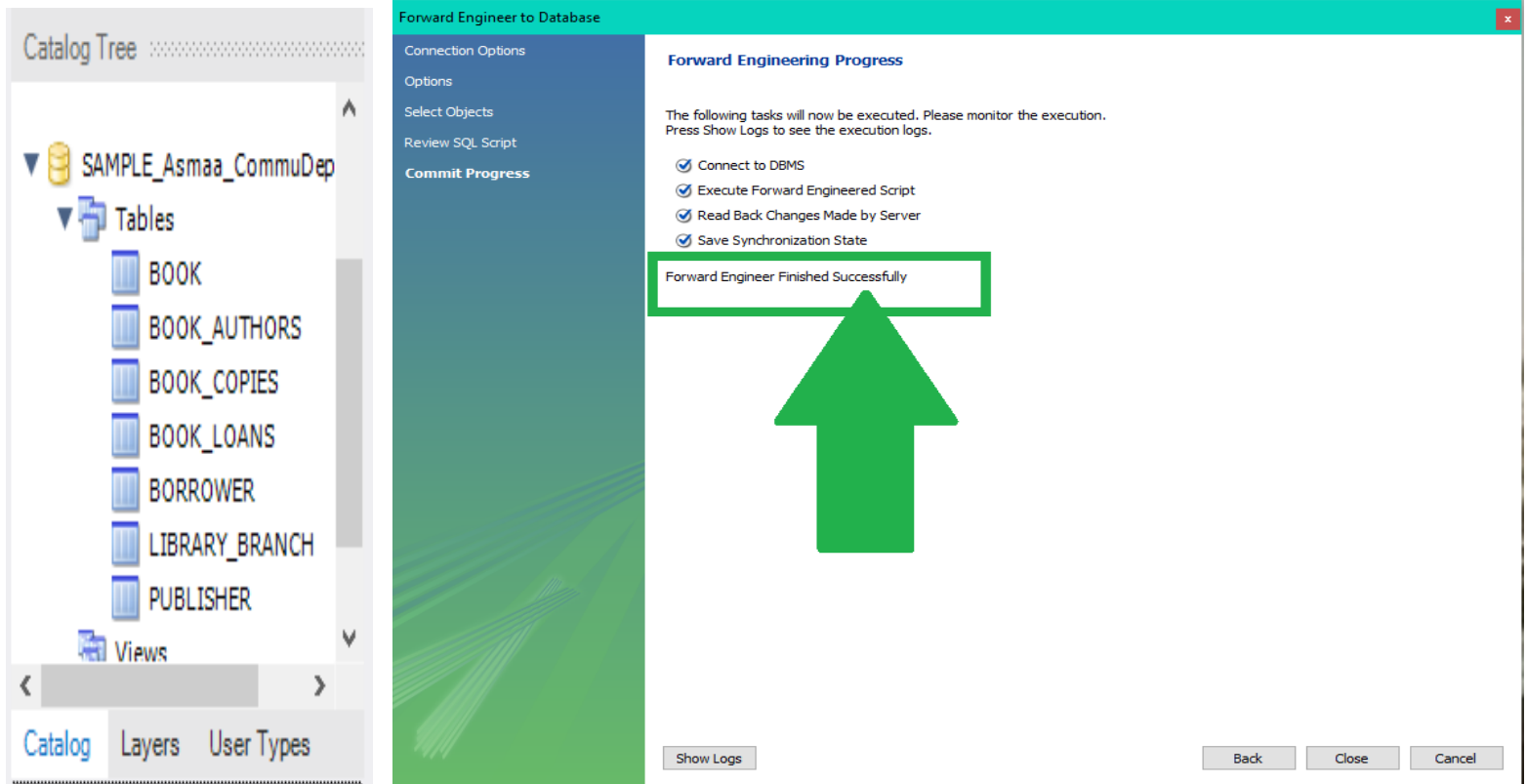
## 5. Write the steps and show the ERD, DDL scripts, created tables.

**ANS )** As shown in all snippets and code above:

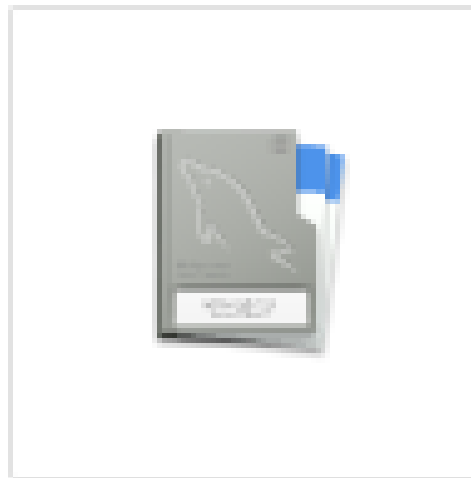
- ✓ Firstly, drew the ERD, each table had its entity.
- ✓ Then, we determined the relations between each two entities.
- ✓ Finally, we generated the equivalent tables and DDL statement to the drawn ERD.

## 6. Connect to the database and verify the creation of tables.

**ANS)** All database Tables are created successfully.



## 7. Save the generated scripts to your hard disk.

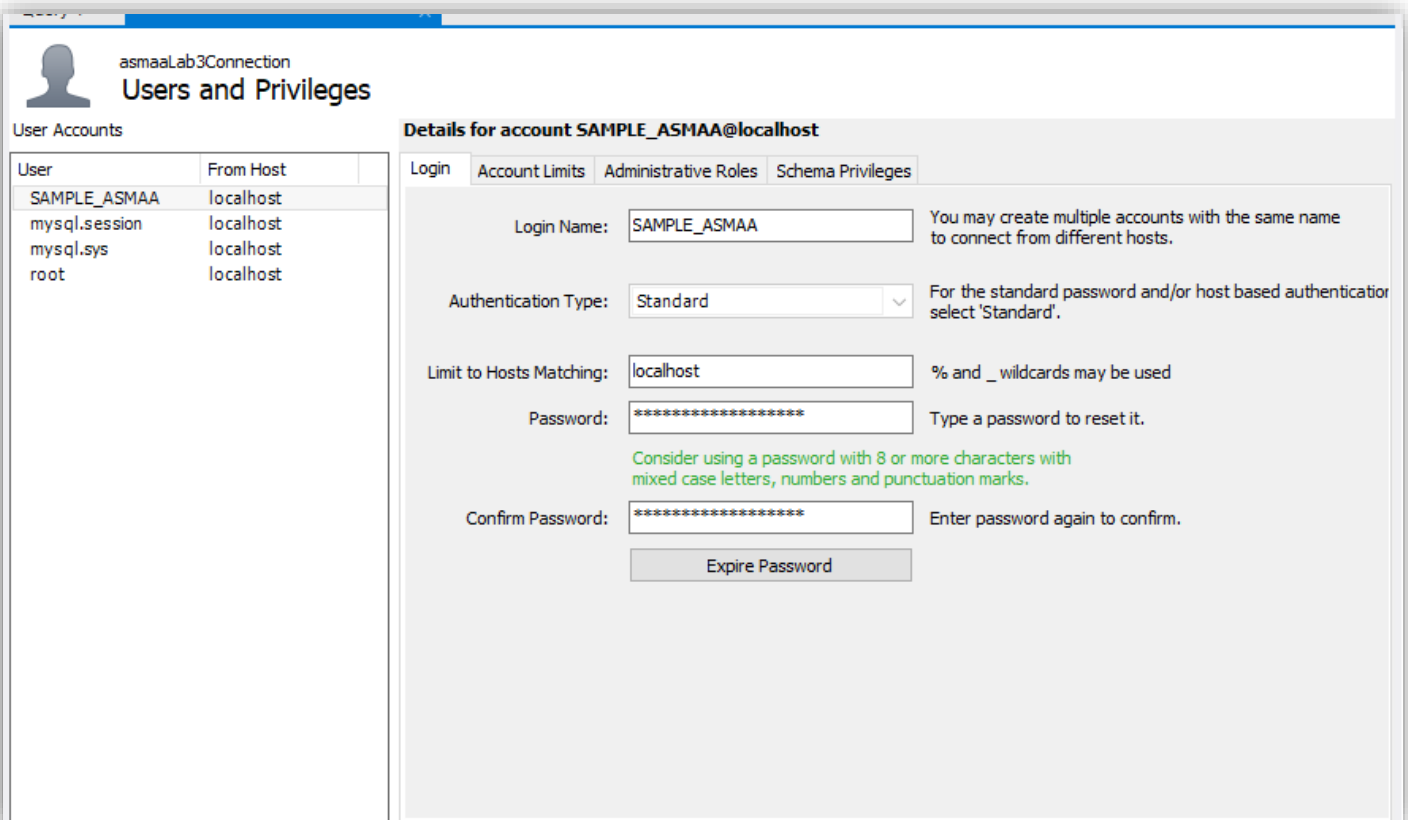
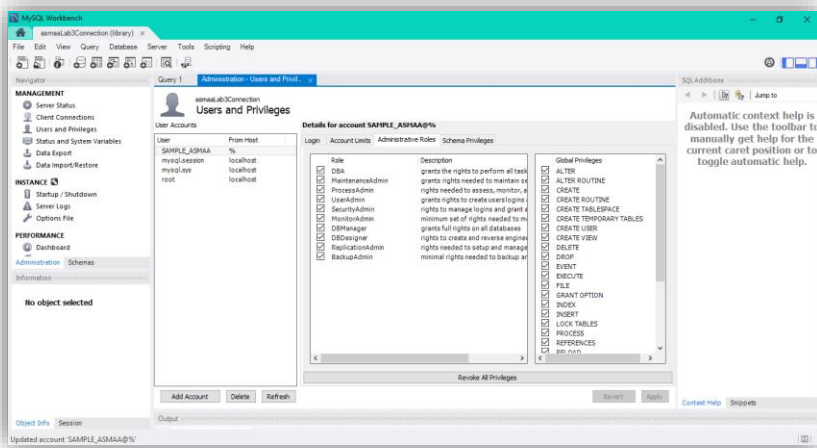
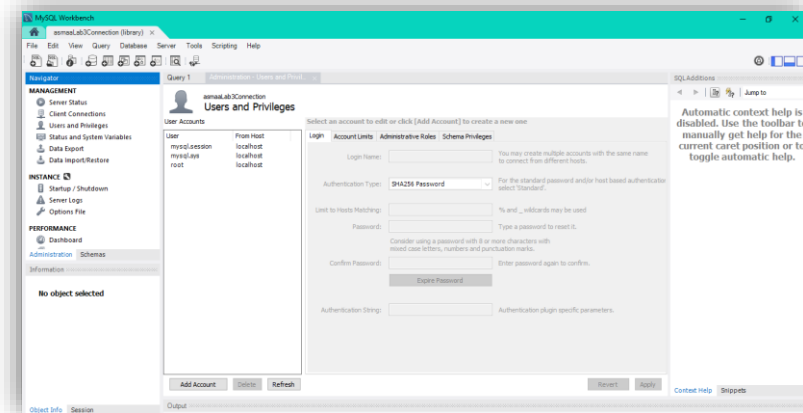


asmaa\_lab3\_DDL  
script



# Database Administration:

8. Using the SQL Developer, create a new user account called SAMPLE and grant it as DBA.



## 9. Write a simple program or use an offline tool, to insert about 200,000 records or more for each table

### Code:

```
1  DELIMITER $$
2  /*
3   Delimiters other than the default ; are typically used when defining functions,
4   stored procedures, and triggers wherein you must define multiple statements.
5   defining a different delimiter like $$ which is used to define the end of the entire procedure,
6   but inside it, individual statements are each terminated by ;.
7  */
8  create procedure insertRandom(in numRows int)
9  begin
10     declare i int ;
11     declare pName varchar(45) ;
12     declare count int default 0;
13     set i=1;
14     start transaction;
15     loop1: while i<= numRows do
16         set pName= CONCAT(left(MD5(RAND()), 39),i);
17         select count(*) into count from publisher where 'PublisherName'=pName;
18         if count > 0 then
19             iterate loop1;
20         end if;
21         insert into publisher values (pName, left(MD5(RAND()),100), left(MD5(RAND()),13) );
22         set i=i+1;
23     end while;
24     commit;
25
26     set i=1;
27     start transaction;
28     while i<= numRows do
29         insert into book( Title, PublisherName)
30         select left(MD5(RAND()), 45), PublisherName
31         FROM publisher limit i,1;
32         set i=i+1;
33     end while;
34     commit;
35
36
37     set i=1;
38     start transaction;
39     while i<= numRows do
40         set pName= CONCAT(left(MD5(RAND()), 39),i);
41         select count(*) into count from book_authors where 'AuthorName'=pName;
42         insert into book_authors(AuthorName,BookID)
43         select pName , BookID from book limit i,1;
44         set i=i+1;
45     end while;
46     commit;
```

```

47
48     set i=1;
49     start transaction;
50     while i<= numRows do
51         insert into library_branch(BranchName,Address)
52         Values(left(MD5(RAND()),45), left(MD5(RAND()),100));
53         set i=i+1;
54     end while;
55     commit;
56
57     set i=1;
58     start transaction;
59     while i<= numRows do
60         insert into borrower (borrowerName,Address,phone)
61         Values(left(MD5(RAND()),45), left(MD5(RAND()),100),left(MD5(RAND()),13));
62         set i=i+1;
63     end while;
64     commit;
65
66     set i=1;
67     start transaction;
68     while i<= numRows do
69         insert into book_copies ( BookID, BranchID, No_Of_Copies)
70         select BookID,BranchID, (10+CEIL(RAND()*(1000-10)))
71         FROM book, library_branch limit i,1;
72         set i=i+1;
73     end while;
74     commit;
75
76     set i=1;
77     start transaction;
78     while i<= numRows do
79         insert into book_loans( BookID, BranchID, CardNo, DateOut, DueDate)
80         select BookID,BranchID, CardNo , curdate(),curdate()
81         FROM book, library_branch, borrower limit i,1;
82         set i=i+1;
83     end while;
84     commit;
85     end$$
86
87     /* Finally, reset the delimiter to the default ; */
88     DELIMITER ;

```

Steps:

library\_branch

publisher

Views

Stored Procedures

insertRandom

Functions

sys

world

Administration

Schemas

Information

66

67

68

69

70

71

72

73

74

start

while

end

comm

end\$

Call stored procedure sample\_asmaa\_commudepartment...

Enter values for parameters of your procedure and click <Execute> to create an SQL editor and run the call:

numRows 200000 [IN] int

Execute Cancel

Random

asmaaLab3StoredProcedure

SQL File 5\*

sample\_asmaa\_c

Don't Limit

1 • SELECT \* FROM sample\_asmaa\_commudepartment.book\_loans;

Result Grid

Filter Rows:

book\_loans1 x

BookID	BranchID	CardNo	DateOut	DueDate
1	1	2	2023-12-10	2023-12-10
2	1	1	2023-12-10	2023-12-10
2	1	2	2023-12-10	2023-12-10
3	1	1	2023-12-10	2023-12-10
4	1	1	2023-12-10	2023-12-10
5	1	1	2023-12-10	2023-12-10
6	1	1	2023-12-10	2023-12-10
7	1	1	2023-12-10	2023-12-10
8	1	1	2023-12-10	2023-12-10
9	1	1	2023-12-10	2023-12-10

1 • SELECT count(\*) FROM sample\_asmaa\_commudepartment.book\_loans;

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

count(*)
20000

Random

asmaaLab3StoredProcedure

SQL File 5\*

sample\_asmaa\_commudepartm...

Info

Tables

Columns

Indexes

Triggers

Views

Stored Procedures

Functions

Grants

Events

Name	Engine	Version	Row Format	Rows	Avg Row Length	Data Length	Max Data Length	Index Length	Data Free	Auto Incre...
book	InnoDB	10	Dynamic	19965	132	2.5 MiB	0.0 bytes	1.5 MiB	4.0 MiB	2000
book_authors	InnoDB	10	Dynamic	18834	140	2.5 MiB	0.0 bytes	1.5 MiB	4.0 MiB	
book_copies	InnoDB	10	Dynamic	19879	79	1.5 MiB	0.0 bytes	304.0 KiB	4.0 MiB	
book_loans	InnoDB	10	Dynamic	20000	79	1.5 MiB	0.0 bytes	1.0 MiB	4.0 MiB	
borrower	InnoDB	10	Dynamic	20002	131	2.5 MiB	0.0 bytes	0.0 bytes	4.0 MiB	2000
library_branch	InnoDB	10	Dynamic	20111	131	2.5 MiB	0.0 bytes	0.0 bytes	4.0 MiB	2000
publisher	InnoDB	10	Dynamic	17274	213	3.5 MiB	0.0 bytes	0.0 bytes	4.0 MiB	

Count: 7

Maintenance >

Inspect Table

Refresh

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 1	23:45:50	/* Delimiters other than the default ; are typically used when defining functions, stored...	0 row(s) affected	0.000 sec
✓ 2	23:46:22	call sample_asmaa_commudepartment.insertRandom(20000)	0 row(s) affected	655.219 sec



Before insertion:

Name	Engine	Version	Row Format	Rows	Avg Row Length	Data Length	Max Data Length	Index Length	Data Free	Auto Incre...
book	InnoDB	10	Dynamic	0	0	16.0 KIB	0.0 bytes	16.0 KIB	0.0 bytes	0
book_authors	InnoDB	10	Dynamic	0	0	16.0 KIB	0.0 bytes	0.0 bytes	0.0 bytes	0
book_copies	InnoDB	10	Dynamic	0	0	16.0 KIB	0.0 bytes	16.0 KIB	0.0 bytes	0
book_loans	InnoDB	10	Dynamic	0	0	16.0 KIB	0.0 bytes	32.0 KIB	0.0 bytes	0
borrower	InnoDB	10	Dynamic	0	0	16.0 KIB	0.0 bytes	0.0 bytes	0.0 bytes	0
library_branch	InnoDB	10	Dynamic	0	0	16.0 KIB	0.0 bytes	0.0 bytes	0.0 bytes	0
publisher	InnoDB	10	Dynamic	0	0	16.0 KIB	0.0 bytes	0.0 bytes	0.0 bytes	0

After Insertion:

asmaaLab3StoredProcedure*											
asmaaLab3StoredProcedure_C...											
sample_asmaa_commudepartm...											
Info	Tables	Columns	Indexes	Triggers	Views	Stored Procedures	Functions	Grants	Events		
Name	Engine	Version	Row Format	Rows	Avg Row Length	Data Length	Max Data Length	Index Length	Data Free	Auto Incre...	
book	InnoDB	10	Dynamic	19965	132	2.5 MiB	0.0 bytes	1.5 MiB	4.0 MiB	2000	
book_authors	InnoDB	10	Dynamic	18834	140	2.5 MiB	0.0 bytes	1.5 MiB	4.0 MiB		
book_copies	InnoDB	10	Dynamic	19879	79	1.5 MiB	0.0 bytes	304.0 KiB	4.0 MiB		
book_loans	InnoDB	10	Dynamic	20000	79	1.5 MiB	0.0 bytes	1.0 MiB	4.0 MiB		
borrower	InnoDB	10	Dynamic	20002	131	2.5 MiB	0.0 bytes	0.0 bytes	4.0 MiB	2000	
library_branch	InnoDB	10	Dynamic	20111	131	2.5 MiB	0.0 bytes	0.0 bytes	4.0 MiB	2000	
publisher	InnoDB	10	Dynamic	17274	213	3.5 MiB	0.0 bytes	0.0 bytes	4.0 MiB		

The screenshot shows a SQL IDE with the following SQL code:

```

8 • alter table book_authors row_format= compressed;
9 • alter table book_copies row_format= compressed;
10 • alter table book_loans row_format= compressed;
11 • alter table borrower row_format= compressed;
12 • alter table library_branch row_format= compressed;
13 • alter table publisher row_format= compressed;
14 • select table_name as "table",
15     ROUND(((data_length+ index_length)/1024/1024),2 ) as "Size(MB)"
16     from information_schema.tables
17     where table_schema="sample_asmaa_commudepartm"
18     order by(data_length+ index_length) DESC;

```

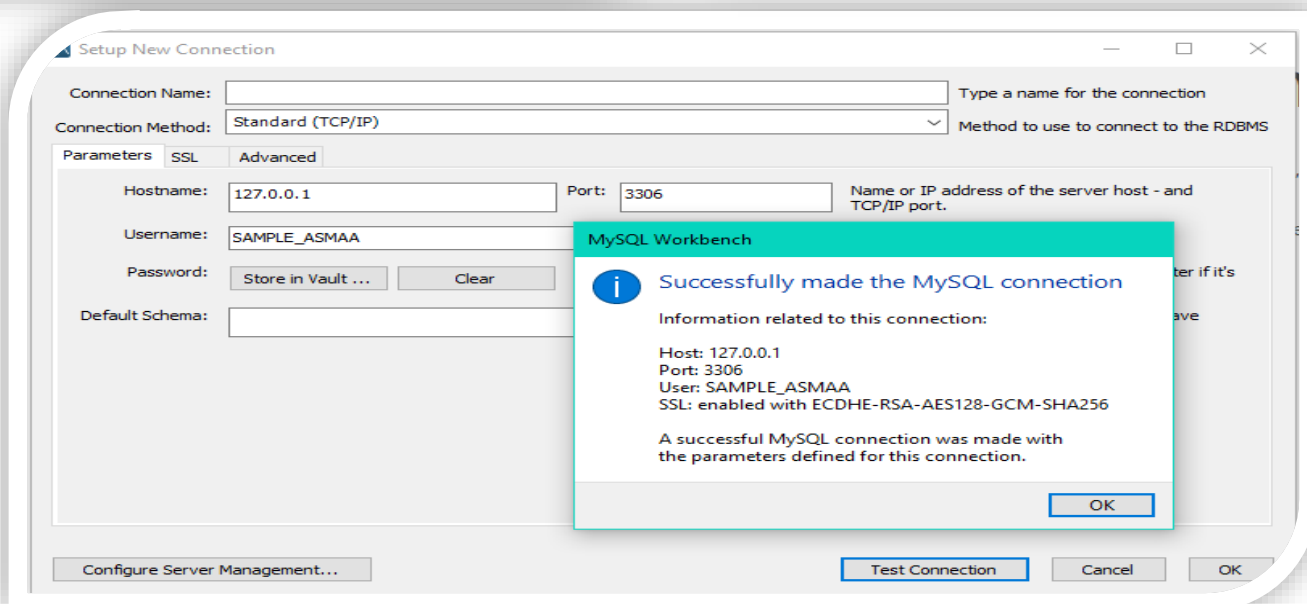
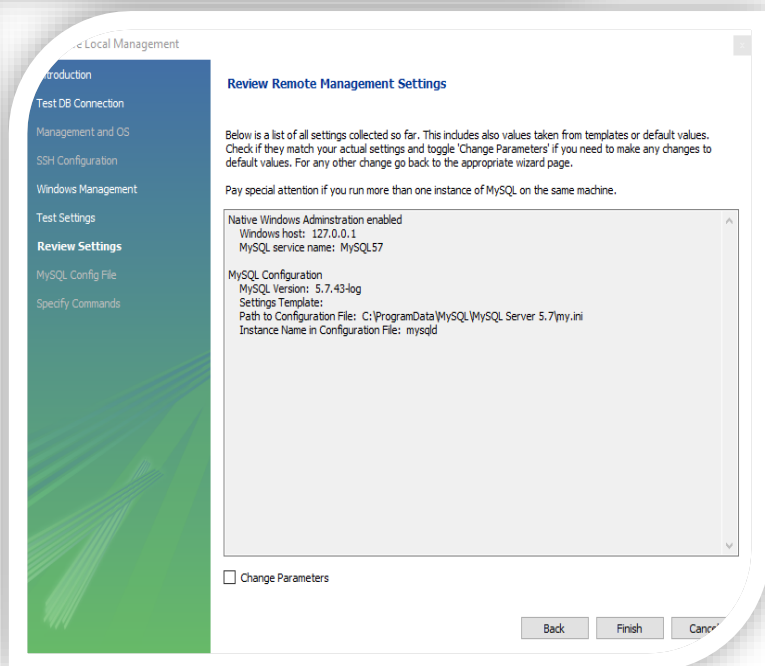
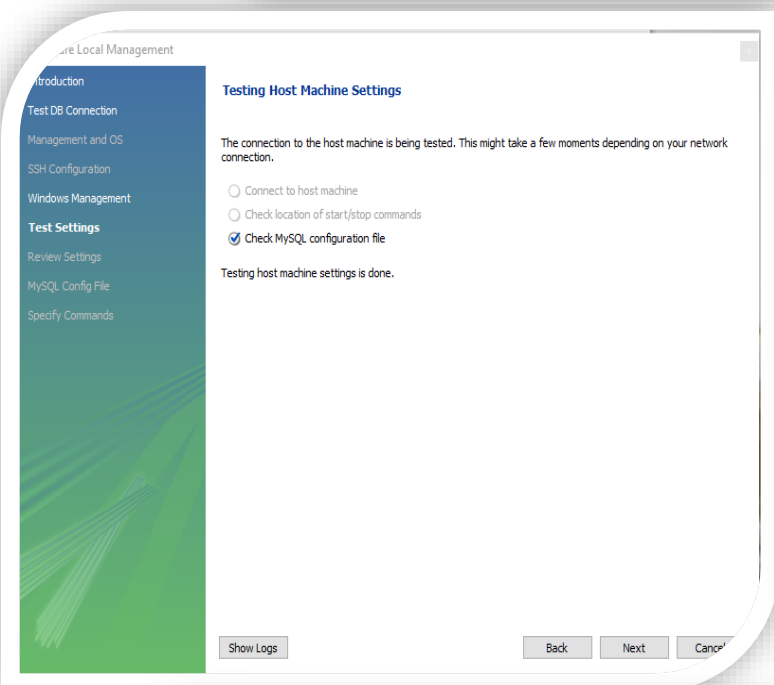
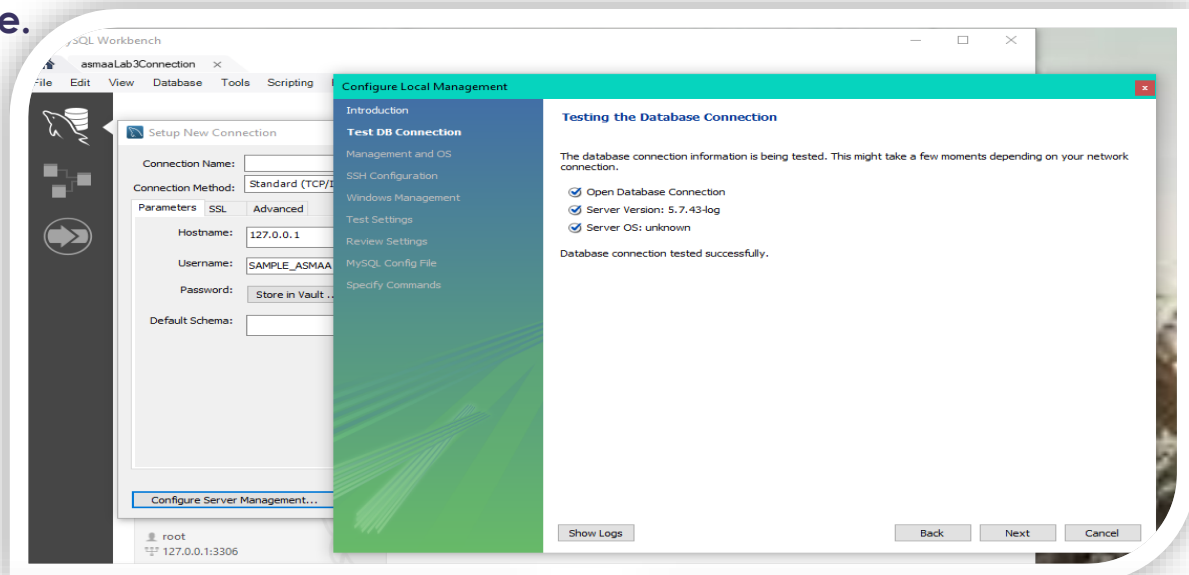
The results grid shows the following data:

table	Size(MB)
book	3.52
book_authors	3.02
publisher	2.76
borrower	2.76
library_branch	2.26
book_loans	1.30
book_copies	0.90

The bottom of the screenshot shows the 'Action Output' tab with the following entries:

#	Time	Action	Message	Duration / Fetch
6	01:04:29	alter table library_branch row_format= c...	0 row(s) affected Records: 0 Duplicate...	0.500 sec
7	01:04:29	alter table publisher row_format= compr...	0 row(s) affected Records: 0 Duplicate...	0.453 sec

12. establish a local network between your database host machine, and another machine that will be a database client. Then, configure the network environment between the two machines and connect from the client machine to the database on the host one.




**ANS)** There are 64 matched row records.

**14. Expire the password of the HR account and drop the SAMPLE one.**

## 1<sup>ST</sup> expiring method:

## 2<sup>nd</sup> expiring method:



asmaalab3Connection

Users and Privileges

User Accounts

User	From Host
HR	%
SAMPLE_ASMAA	%
mysql.session	localhost
mysql.sys	localhost
root	localhost

Add Account
Delete
Refresh

Details for account HR@%

Login
Account Limits
Administrative Roles
Schema Privileges

Login Name: HR

You may cre to connect fi

Authentication Type: Standard

For the stan select 'Stanc

Limit to Hosts Matching: %

% and \_ wil

Password: \*\*\*\*\*

Type a pass

Consider using a password with 8 or more character mixed case letters, numbers and punctuation marks

Confirm Password: \*\*\*\*\*

Enter passw

Expire Password

Force user to change password after next login. The user will be unable to issue any command other than SET PASSWORD.

## Dropping:

28 • `drop user 'SAMPLE' '@'%;`

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output

Output


Output

Output

## BEFORE dropping:

asmaaLab3AdministrationQuesti...

Administration - Users and Privile...

 asmaaLab3Connection

### Users and Privileges

User Accounts

User	From Host
HR	%
SAMPLE	%
SAMPLE_ASMAA	%
mysql.session	localhost
mysql.sys	localhost
root	localhost

Add AccountDeleteRefresh

Details for account SAMPLE@%

Login

Account Limits

Administrative Roles

Schema Privileges

Login Name: SAMPLE

You may cre to connect f

Authentication Type: Standard

For the stan select 'Stand

Limit to Hosts Matching: %

% and \_ wilc

Password: \*\*\*\*\*

Type a pass

Consider using a password with 8 or more character mixed case letters, numbers and punctuation marks

Confirm Password: \*\*\*\*\*

Enter passw

Expire Password

Revert

Apply

## AFTER dropping:

asmaaLab3Connection

Users and Privileges

User Accounts

User	From Host
HR	%
SAMPLE_ASMAA	%
mysql.session	localhost
mysql.sys	localhost
root	localhost

Add AccountDeleteRefresh

Select an account to edit or click [Add Account] to create a new one

Login

Account Limits

Administrative Roles

Schema Privileges

Login Name:

You may cre to connect f

Authentication Type: SHA256 Password

For the stan select 'Stand

Limit to Hosts Matching:

% and \_ wilc

Password:

Type a pass

Consider using a password with 8 or more character mixed case letters, numbers and punctuation marks

Revert

Apply



## • Stored Procedures

15. Consider the following database schema:

**DEPT** (Dnumber , Dname, Founded, Mgr\_ssn, Budget)

**EMPLOYEE** (Ssn, Ename , Bdate, Dno, Salary)

Note that the attribute Founded represents the foundation date of the department.  
Create the database schema containing tables above.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane shows the 'companylab3' database with tables 'dept' and 'employee'. The 'dept' table is selected. The right pane shows the SQL script for creating the tables:

```
1 • create schema companyLab3;
2 • use companyLab3;
3 • create table DEPT(
4     Dnumber int auto_increment primary key,
5     Dname varchar(45),
6     founded date,
7     Mgr_ssn int ,
8     budget int);
9 • create table EMPLOYEE(
10    Ssn int primary key auto_increment,
11    Ename varchar(45),
12    Bdate date,
13    Dno int,
14    salary int);
15 • alter table DEPT add constraint FK_Mgr_ssn FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn);
16 • alter table EMPLOYEE add constraint FK_Dno FOREIGN KEY (Dno) REFERENCES DEPT(Dnumber);
```

16. Create a stored function Count\_Emp (Dnumber NUMBER) that returns the number of employees working for the department Dnumber.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane shows the 'companylab3' database with tables 'dept' and 'employee', and a function 'f() Count\_Emp'. The right pane shows the SQL script for creating the function:

```
1 DELIMITER **
2 • create function Count_Emp (Dnumber int) returns int deterministic
3 Begin
4     declare myCount int;
5     select COUNT(*) into myCount from EMPLOYEE where Dno=Dnumber;
6     return myCount;
7 end **
8 DELIMITER ;
```

Below the script, the 'Output' pane shows the execution results:

#	Time	Action	Message
✓ 1	00:22:24	create function Count_Emp (Dnumber int) returns int deterministic Begin declare myC...	0 row(s) affected

17. Create a stored procedure that ensures that Year(DEPT.Founded) >=1960 for all departments; if a row violates this constraint then set its date to be '01-JAN-1960'

```

1 DELIMITER !!
2 • create procedure set_foundation_year()
3 begin
4 update DEPT set founded='1960/01/01' where year(founded) < 1960;
5 end !!
6 DELIMITER ;

```

18. Create a trigger to ensure that no department has more than 8 employees.

Navigator

SCHMAS

Filter objects

- companylab3
  - Tables
    - dept
    - employee
      - Columns
      - Indexes
      - Foreign Keys
      - Triggers
  - check\_employee\_count
- Views
- Stored Procedures
  - set\_foundation\_year
- Functions
  - f() Count\_Emp

```

1 DELIMITER ##
2 • create trigger check_employee_count before insert on EMPLOYEE for each row
3 begin
4 if Count_Emp(NEW.Dno) >= 8 then
5 -- equal because the trigger should be triggered after inserting the 8 the employee
6 -- SET SQL_SAFE_UPDATES = 0;
7 signal sqlstate '45000';
8 -- SET SQL_SAFE_UPDATES = 1;
9 end if;
10 end ##
11 DELIMITER ;

```

19. Create a trigger to implement "ON UPDATE CASCADE" for the foreign key EMPLOYEE.Dno.

SQL Workbench

asmaaLab3Connection

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHMAS

Filter objects

- Columns
  - Dnumber
  - Dname
  - founded
  - Mgr\_ssn
  - budget
- Indexes
- Foreign Keys
- Triggers
  - onUpdate\_DeptNum\_cascade\_theEmpDno
- employee
  - Columns
    - Ssn
    - Ename
    - Bdate
    - Dno
    - salary
  - Indexes
  - Foreign Keys

Administration Schemas

Information

Trigger: onUpdate\_DeptNum\_cascade\_theEmpDno

Definition:

Event UPDATE

Timing AFTER

Object Info Session

```

1 DELIMITER ^^
2 • create trigger onUpdate_DeptNum_cascade_theEmpDno after update on DEPT for
3 begin
4 update employee
5 set Dno = new.Dnumber
6 where Dno = old.Dnumber ;
7 end ^^
8 DELIMITER ;

```

Output

#	Time	Action	Message	Duration / Fetch
1	17:35:28	create trigger onUpdate_DeptNum_...	0 row(s) affected	0.015 sec

20. Create a trigger to ensure that whenever an employee is given a raise in salary, his department manager's salary must be increased to be at least as much.

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Navigator' pane displays the 'companylab3' schema, including tables like 'dept', 'employee', and 'check\_employee\_count'. The 'Stored Procedures' section is expanded, showing 'update\_reasonable\_salary' selected. Below it, the 'Parameters' section lists 'employeeSSN' and 'newSalary' as input integers.

The main editor shows the SQL code for the stored procedure:

```

1 DELIMITER $$
2 create procedure update_reasonable_salary(in employeeSSN int,in newSalary int )
3 begin
4     declare managerSSN int;
5
6     -- join to know the manager of the disered employee:
7     select Mgr_ssn into managerSSN from
8         DEPT natural join employee where Ssn= employeeSSN;
9
10    -- if the manger salary is less than the empolyee salary ,
11    -- update the manager salary
12    update employee
13        set salary = newSalary
14        where salary < newSalary and Ssn=managerSSN;
15
16    -- then update the employee salary
17    update employee
18        set salary = newSalary
19        where Ssn=employeeSSN;
20 end $$
21 DELIMITER ;

```

The 'Output' pane at the bottom shows the execution result: 'create procedure update\_reasonable\_salary(in employeeSSN int,in newSalary int) 0 row(s) affected'.

21. Test the created functions/procedures/triggers by suitable SQL statements to ensure their correctness.

ANS) testing steps :

1. create a new department "no.1".
2. Insert 8 employees in dept "no.1".
3. When inserting more employees to dept no. 1, the trigger will refuse it.

The screenshot shows two SQL queries executed in the editor:

```

1 SELECT * FROM companylab3.employee;
2 SELECT COUNT(*) FROM companylab3.employee;

```

The results are displayed in the 'Result Grid' pane. The first query shows 8 rows of employee data. The second query shows a single row with the count of 8.

Ssn	Ename	Bdate	Dno	salary
3	Gamal	1920-01-01	1	500
4	Abdel-Halem	1970-01-01	1	750
5	Mabrouk	1920-01-01	1	500
6	Mariam	1970-01-01	1	750
7	Khouloud	1920-01-01	1	500
8	Asmaa Mohamed	1970-01-01	1	750
NULL	NULL	NULL	NULL	NULL

The 'COUNT(\*)' query result is shown as a small table:

COUNT(*)
8

MySQL Workbench

asmaaLab3Connection

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- Foreign Keys
- Triggers
  - onUpdate\_DeptNum\_cascade\_theEmpDno
- employee
  - Columns
    - Ssn
    - Ename
    - Bdate
    - Dno
    - salary
  - Indexes
  - Foreign Keys
  - Triggers
- Views
  - check\_employee\_count
- Stored Procedures
  - set\_foundation\_year
  - update\_reasonable\_salary
- Functions

Administration Schemas

Information

Trigger: check\_employee\_count

Definition:

Event INSERT

Timing BEFORE

Object Info Session

asmaaLab3StoredProcedures\_1 x asmaaLab3StoredProcedures\_c...

Don't Limit

```

1  -- INSERT 1 department:
2  • insert into dept (Dnumber,Dname,founded,budget) values (1, "programming", '1940-01-01',1000);
3  -- insert 8 employees:
4  • insert into employee (Ename,Bdate,Dno,salary) values("Nagy","1920-01-01",1,500);
5  • insert into employee (Ename,Bdate,Dno,salary) values("asmaa", '1970-01-01',1,750);
6  • insert into employee (Ename,Bdate,Dno,salary) values("Gamal","1920-01-01",1,500);
7  • insert into employee (Ename,Bdate,Dno,salary)
8  values
9  ("Abdel-Hale", '1970-01-01',1,750),
10 ("Mabrouk","1920-01-01",1,500),
11 ("Mariam", '1970-01-01',1,750),
12 ("Khouloud","1920-01-01",1,500),
13 ("Asmaa Mohamed", '1970-01-01',1,750);
14 -- insert 9th employee: (The CURDATE function returns the current date)
15 ✖ insert into employee (Ename,Bdate,Dno,salary) values(check_employee_count"9th_employees",curdate(),1,
16 -- DROP TRIGGER check_employee_count:
  
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
6	20:13:09	insert into employee (Ename,Bdate,Dno,salary) values("...	1 row(s) affected	0.000 sec
7	20:14:12	insert into employee (Ename,Bdate,Dno,salary) values("...	1 row(s) affected	0.000 sec
8	20:15:21	insert into employee (Ename,Bdate,Dno,salary) values("...	1 row(s) affected	0.016 sec
9	20:17:09	insert into employee (Ename,Bdate,Dno,salary) values (...)	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
10	20:20:26	insert into employee (Ename,Bdate,Dno,salary) values("...	1 row(s) affected	0.000 sec
11	20:21:21	insert into employee (Ename,Bdate,Dno,salary) values("...	Error Code: 1644. Unhandled user-defined exception co...	0.000 sec

4. Insert the foreign key in the DEPT table:

```

27  -----
28  -- TEST STEP 4:
29  -----
30  • update dept set Mgr_ssn =2 where Dnumber=2 ;
31  • SELECT * FROM companylab3.dept;
32  -----
  
```

Result Grid

	Dnumber	Dname	founded	Mgr_ssn	budget
▶	2	programming	1940-01-01	2	1000
•	NULL	NULL	NULL	NULL	NULL



5. Update Dnumber in the DEPT table to no. 2 , and this new value will be cascaded to the EMPLOYEE table:

```
20 • insert into dept (Mgr_ssn) values (2);
21
22 -- SET FOREIGN_KEY_CHECKS=0; -- to disable them
23 • update DEPT set Dnumber=2 where Dnumber=1;
24 -- SET FOREIGN_KEY_CHECKS=1; -- to re-enable them
25 • select * from employee;
26 -- DROP TRIGGER onUpdate_DeptNum_Cascade_theEmpDno;
27
```

Ssn	Ename	Bdate	Dno	salary
1	Nagy	1920-01-01	2	500
2	asmaa	1970-01-01	2	750
3	Gamal	1920-01-01	2	500
4	Abdel-Halem	1970-01-01	2	750
5	Mabrouk	1920-01-01	2	500
6	Mariam	1970-01-01	2	750
7	Khoulood	1920-01-01	2	500
8	Asmaa Mohamed	1970-01-01	2	750

#	Time	Action	Message	Duration / Fetch
12	22:54:47	update DEPT set Dnumber=2 w...	1 row(s) affected Rows matched...	0.000 sec
13	22:55:00	SET FOREIGN_KEY_CHECKS=1	0 row(s) affected	0.000 sec
14	22:55:41	select * from employee	8 row(s) returned	0.000 sec / 0.000 sec

6. Call set\_foundation\_year to update Founded date from 1940-01-01 to 1960-01-01.

```
41 -----
42 -- TEST STEP 6:
43 -----
44 -- SET SQL_SAFE_UPDATES = 0;
45 • CALL set_foundation_year();
46 -- SET SQL_SAFE_UPDATES = 1;
47 • select * from dept;
```

Dnumber	Dname	founded	Mgr_ssn	budget
2	programming	1960-01-01	2	1000

#	Time	Action	Message	Duration / Fetch
1	23:30:50	CALL set_foundation_year()	1 row(s) affected	0.000 sec
2	23:30:54	SET SQL_SAFE_UPDATES = 1	0 row(s) affected	0.000 sec
3	23:31:00	select * from dept	1 row(s) returned	0.000 sec / 0.000 sec

7. update the salary of an employee to 800 exceeding the boss salary (Mgr\_ssn = 2), this will also update the manager salary automatically to 800:

The screenshot shows a SQL IDE window with a script editor and a results pane. The script editor contains the following SQL code:

```
38 -- SET FOREIGN_KEY_CHECKS=1; -- to re-enable them
39 • select * from employee;
40 -- DROP TRIGGER onUpdate_DeptNum_Cascade_theEmpDno;
41 -----
42 -- TEST STEP 6:
43 -----
44 • CALL update_reasonable_salary(5,800);
45 • select * from employee;
46
```

The results pane displays a table with 6 columns: Ssn, Ename, Bdate, Dno, and salary. The table contains 8 rows of data, including a row with NULL values. The results are as follows:

	Ssn	Ename	Bdate	Dno	salary
▶	1	Nagy	1920-01-01	2	500
	2	asmaa	1970-01-01	2	800
	3	Gamal	1920-01-01	2	500
	4	Abdel-Halem	1970-01-01	2	750
	5	Mabrouk	1920-01-01	2	800
	6	Mariam	1970-01-01	2	750
	7	Khouloud	1920-01-01	2	500
	8	Asmaa Mohamed	1970-01-01	2	750
*	NULL	NULL	NULL	NULL	NULL

The results pane also shows an "Output" section with a table of action results:

#	Time	Action	Message	Duration / Fetch
6	23:19:11	CALL update_reasonable_salary...	0 row(s) affected	0.000 sec
7	23:19:16	select * from employee	8 row(s) returned	0.000 sec / 0.000