LAB 3
DATABASE MANAGEMENT SYSTEM

Prof. Dr. Hicham Elmongui

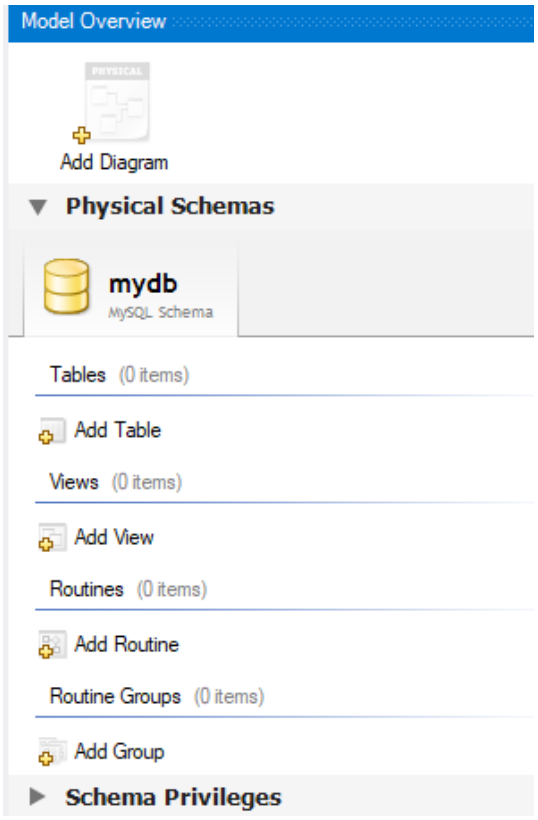
Asmaa Gamal Abdel-Halem Mabrouk Nagy
Communication & Electronics

"Relational Schema" Model using MySQL Workbench Tools

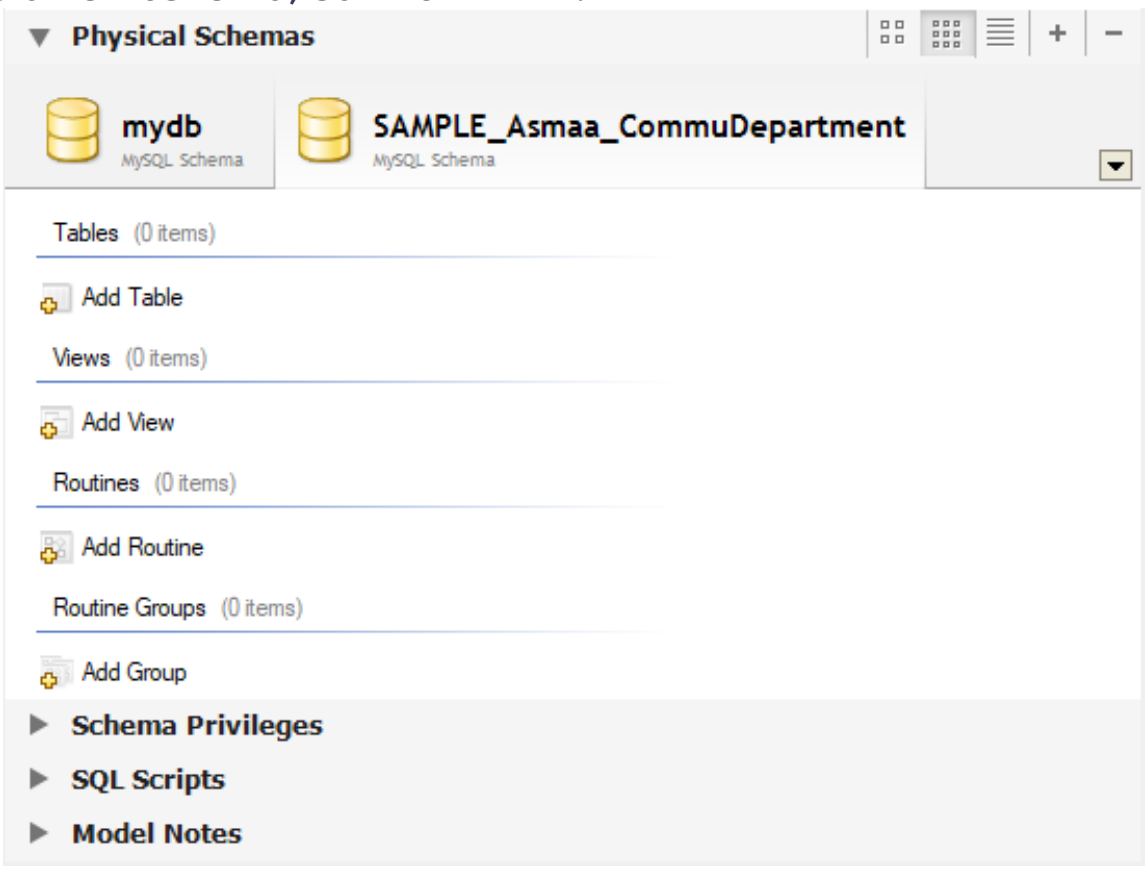
The Lab Report

- ## Data Modeling and Schema Generation:

1. **Use the data modeler module in the Workbench.**



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



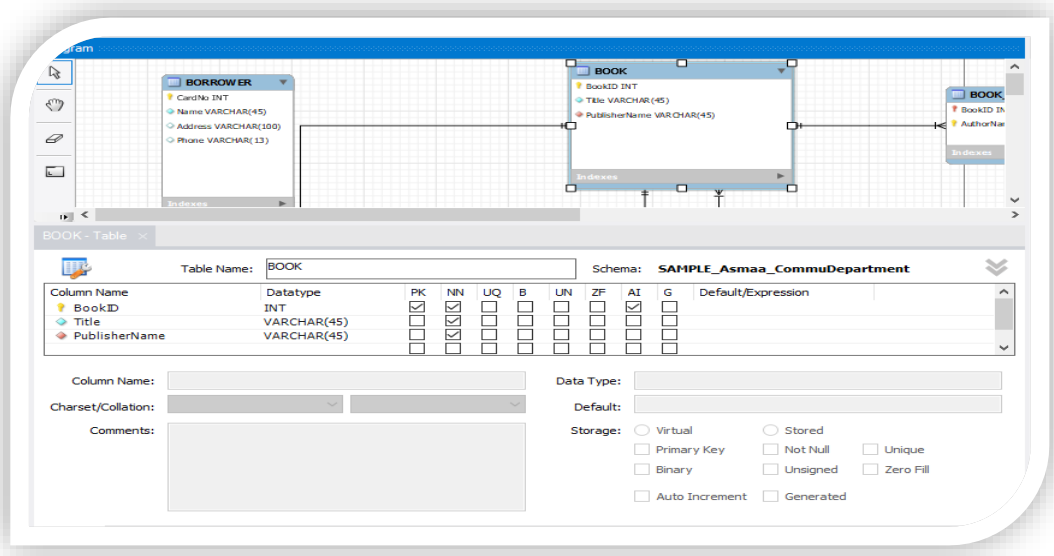
3. **Draw the ERD as shown in the schema figure section.**

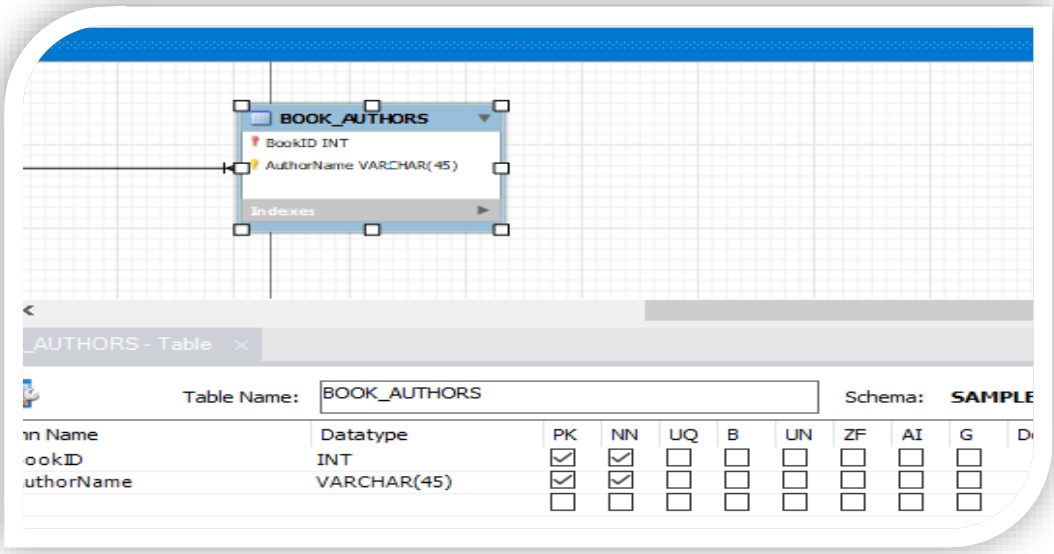
Ans :

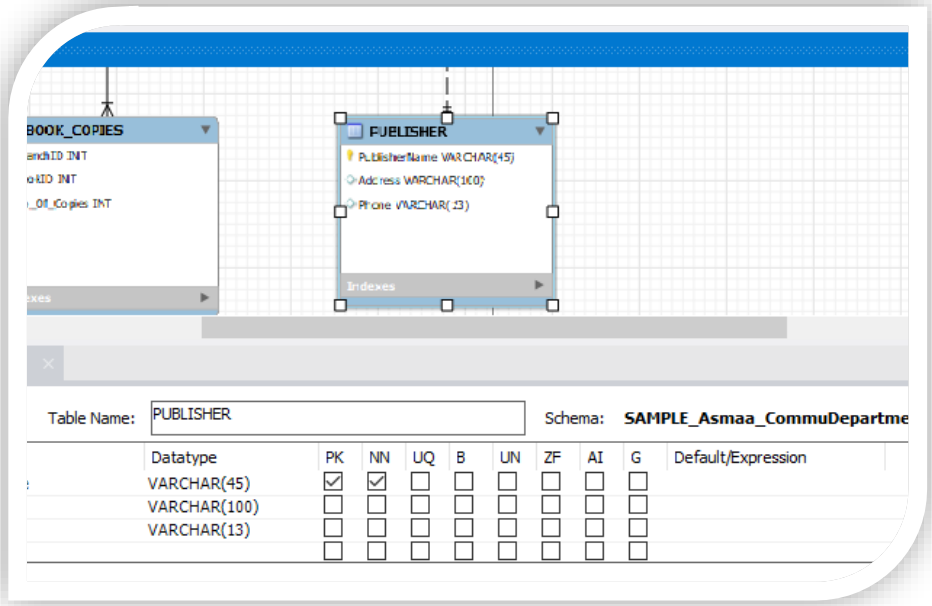
1. Assumptions:

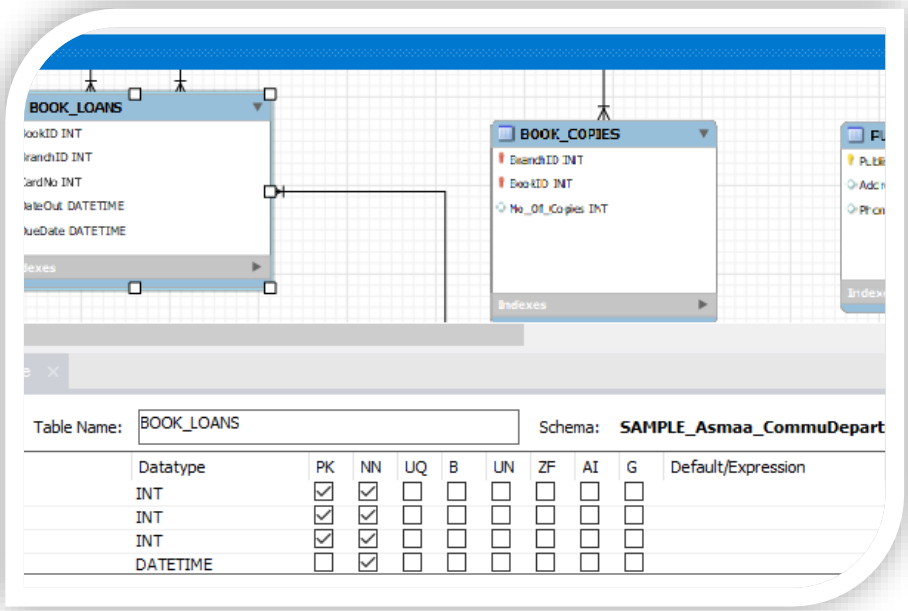
- ✓ Phone numbers are always chars of length 13 == 11 digits + 2 country code digits. "VarChar" because "Int" removes leading zeros. "VarChar" because "Char" is fixed length so "Char" will occupy more space.
- ✓ PK: Primary keys which indicate numbers are set to Auto Increment (AI).
- ✓ All our relationships are one-to-many relationships.
- ✓ The weak entities are:
  - BOOK_AUTHORS
  - BOOK_COPIES
  - BOOK_LOANS

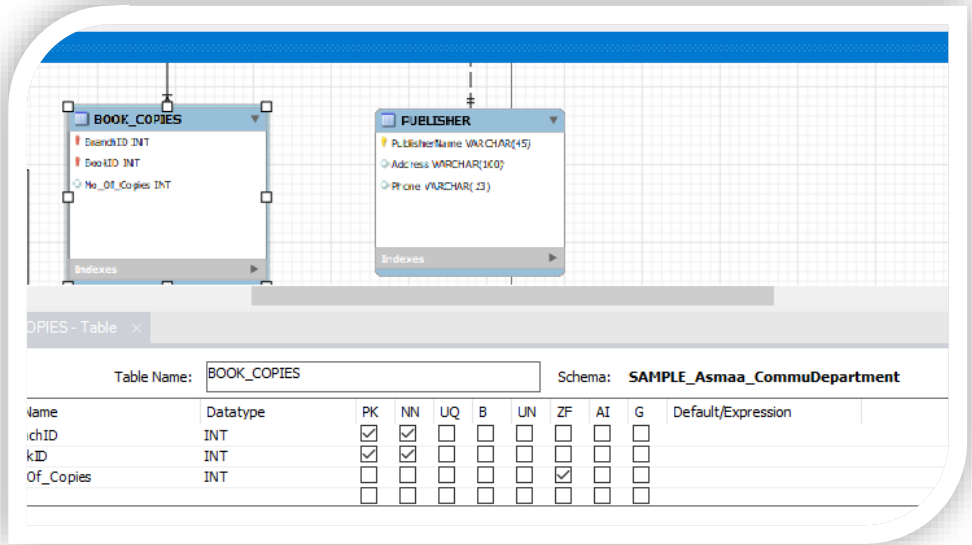
2. Sequential Steps:

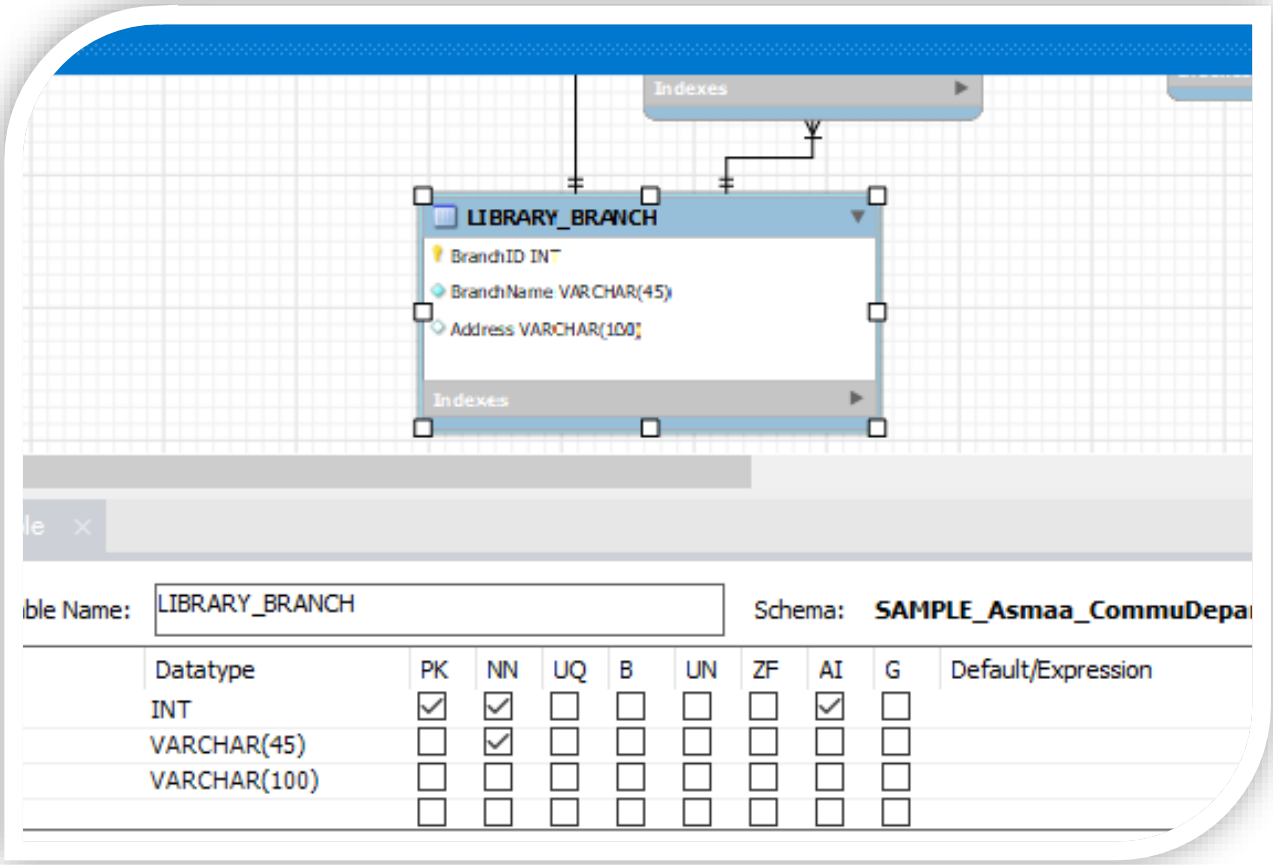


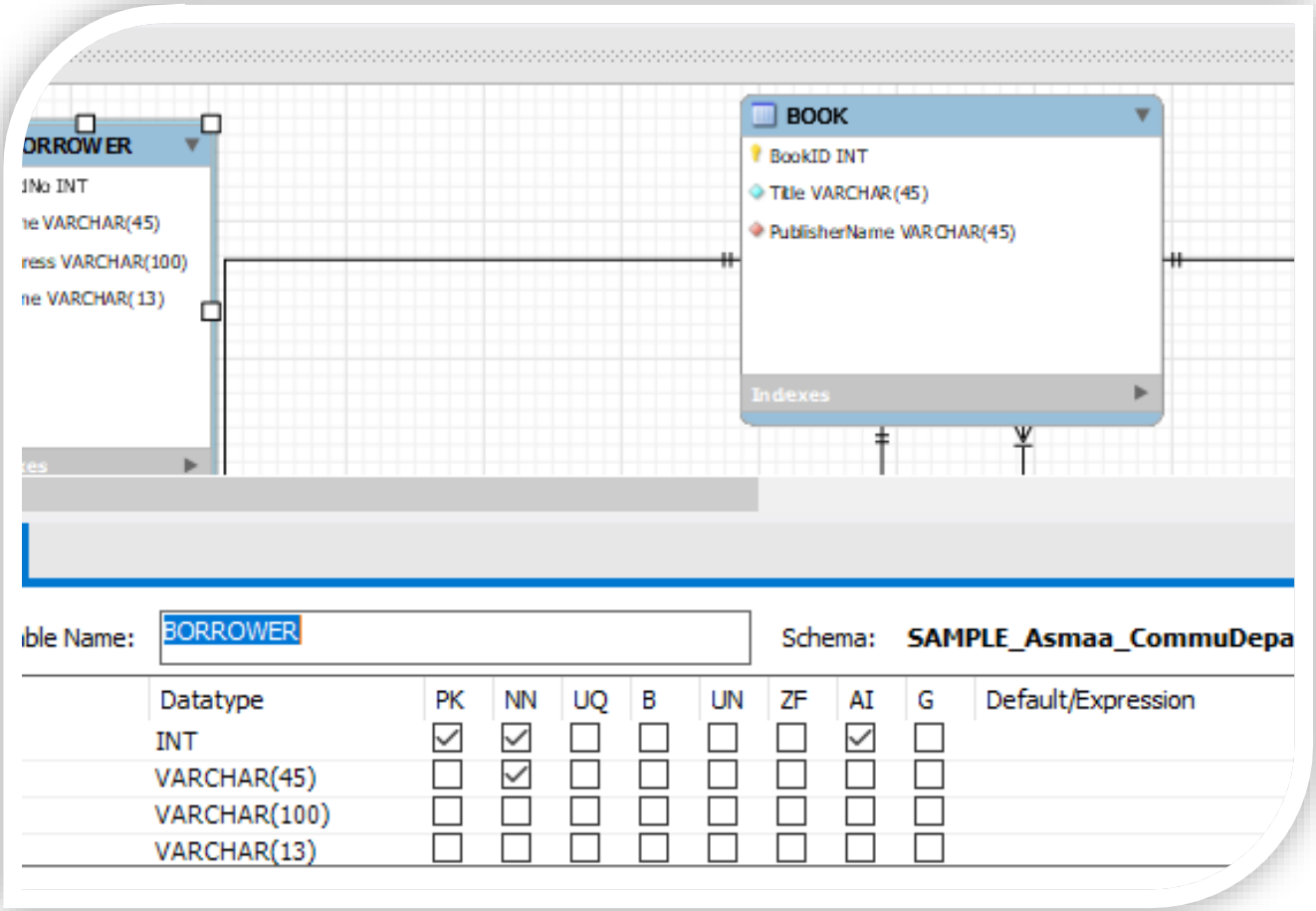



BOOK_COPIES
PUBLISHER
PublisherName VARCHAR(45)
Address VARCHAR(100)
Phone VARCHAR(13)
Indexes
Table Name: PUBLISHER
Schema: SAMPLE_Asmaa_CommuDepartme
Datatype PK NN UQ B UN ZF AI G Default/Expression
VARCHAR(45)
VARCHAR(100)
VARCHAR(13)

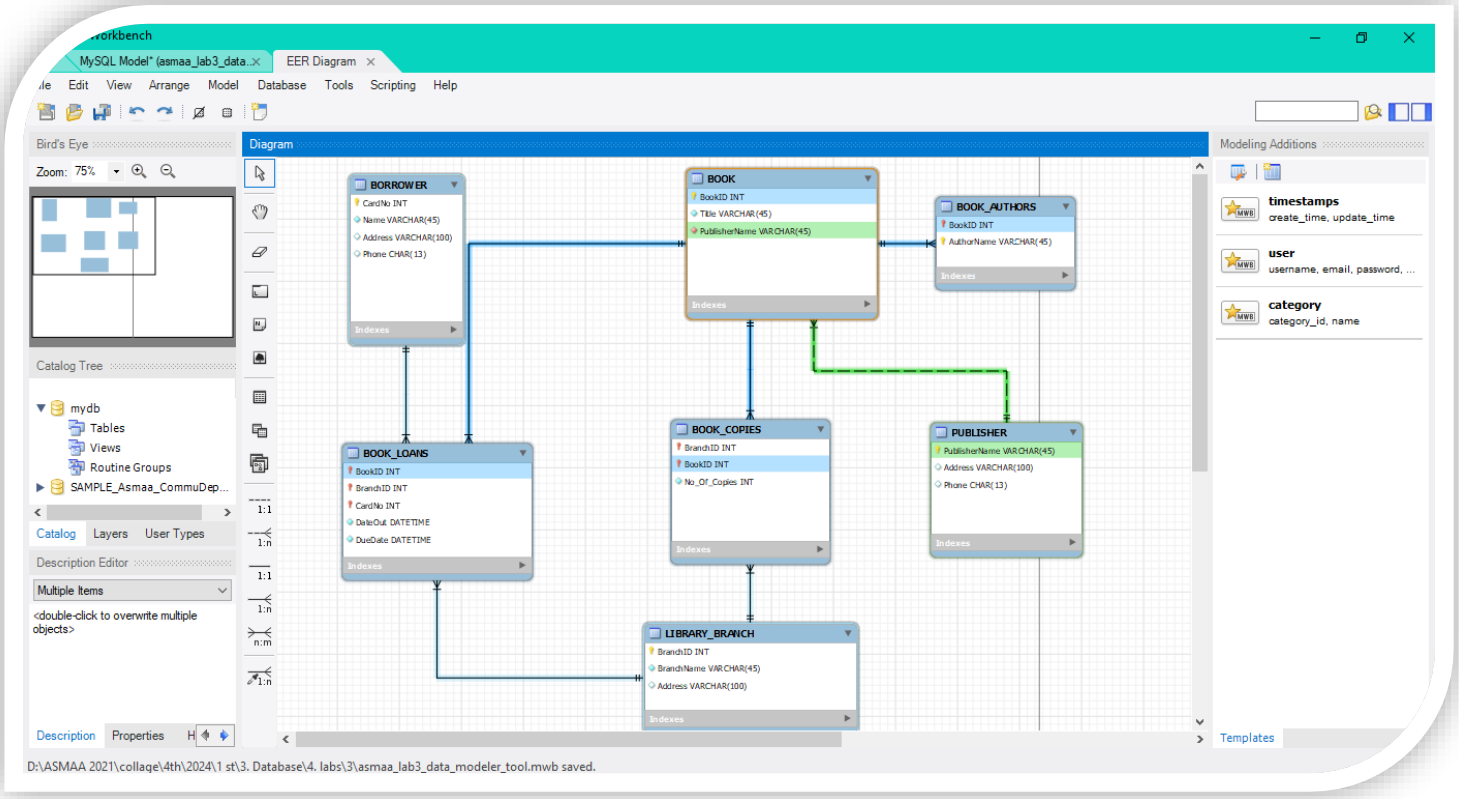

BOOK_LOANS
BOOK_COPIES
Table Name: BOOK_LOANS
Schema: SAMPLE_Asmaa_CommuDepart
Datatype PK NN UQ B UN ZF AI G Default/Expression
INT
INT
INT
DATETIME

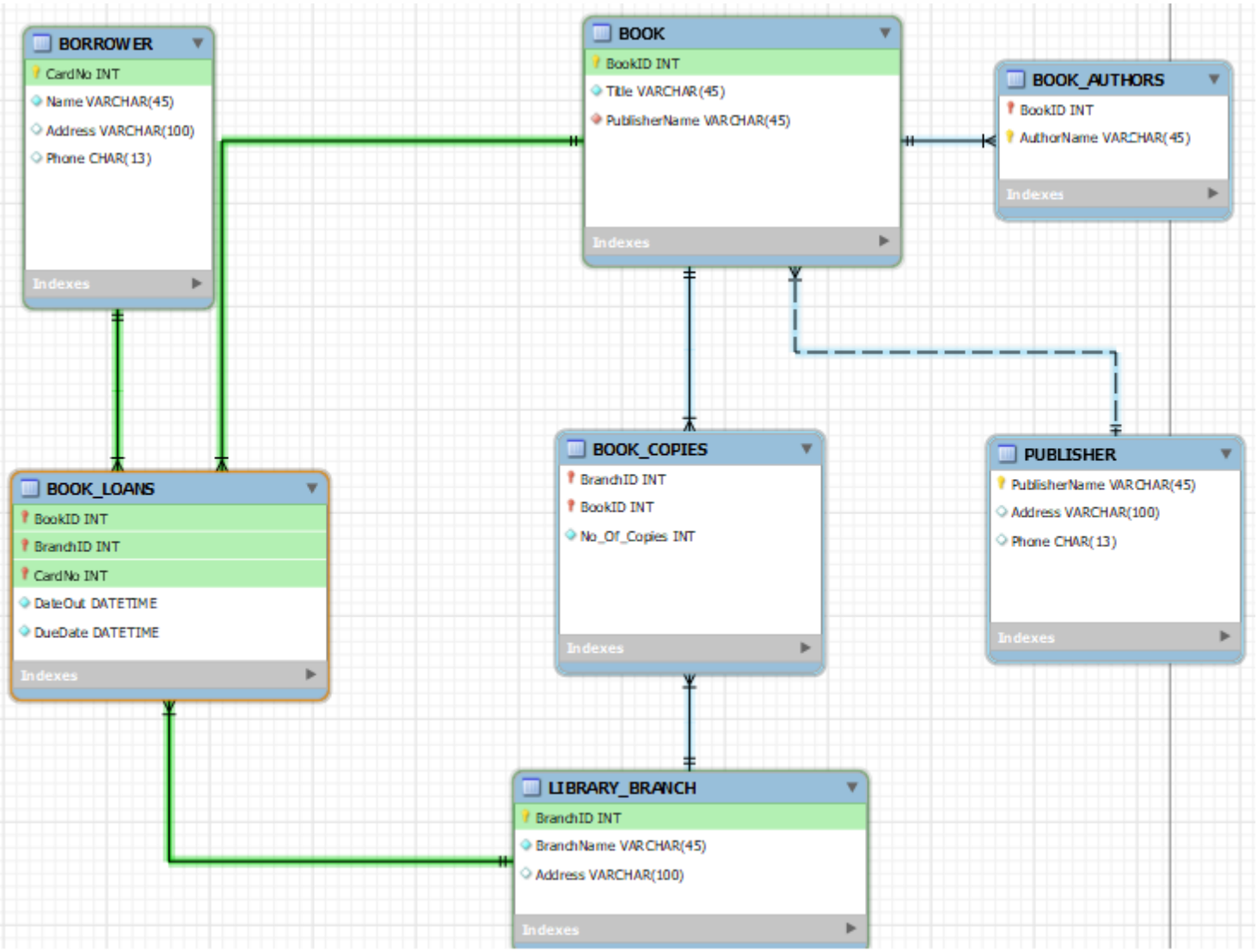

BOOK_COPIES
PUBLISHER
PublisherName VARCHAR(45)
Address VARCHAR(100)
Phone VARCHAR(13)
Indexes
Table Name: BOOK_COPIES
Schema: SAMPLE_Asmaa_CommuDepartment
Datatype PK NN UQ B UN ZF AI G Default/Expression
INT
INT
INT


Indexes
LIBRARY_BRANCH
BranchID INT
BranchName VARCHAR(45)
Address VARCHAR(100)
Indexes
ble Name:
LIBRARY_BRANCH
Schema:
SAMPLE_Asmaa_CommuDepa
Datatype
PK
NN
UQ
B
UN
ZF
AI
G
Default/Expression
INT
VARCHAR(45)
VARCHAR(100)


BOOK
BookID INT
Title VARCHAR(45)
PublisherName VARCHAR(45)
Indexes
ble Name:
BORROWER
Schema:
SAMPLE_Asmaa_CommuDepa
Datatype
PK
NN
UQ
B
UN
ZF
AI
G
Default/Expression
INT
VARCHAR(45)
VARCHAR(100)
VARCHAR(13)

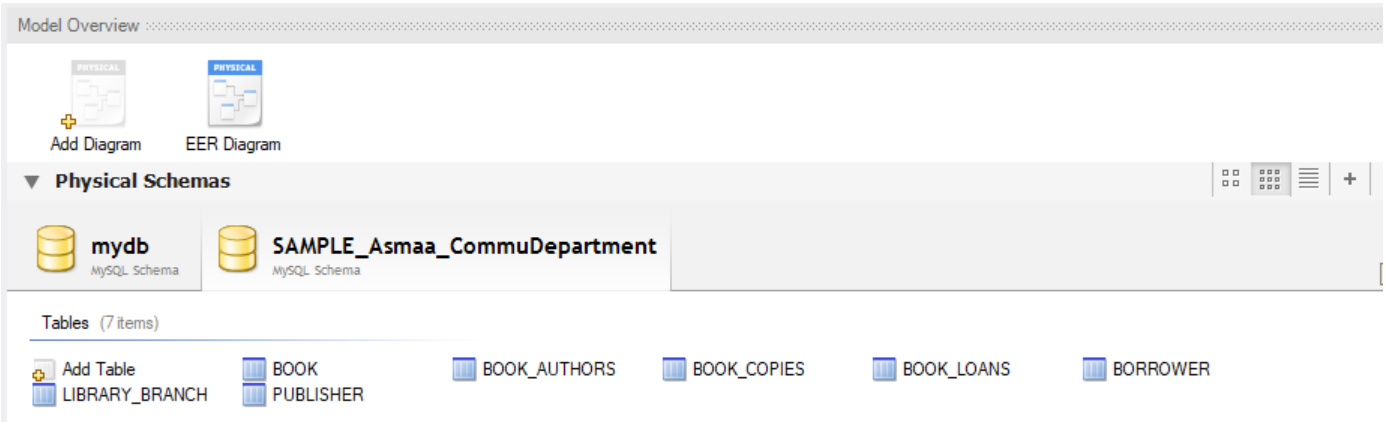
3. Total "LIBRARY ERD Model" Solution:



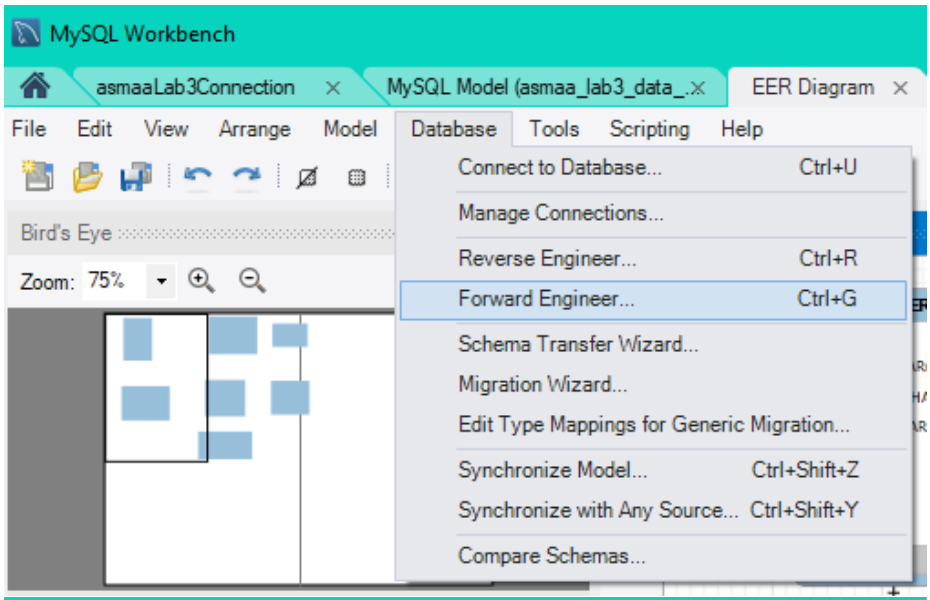


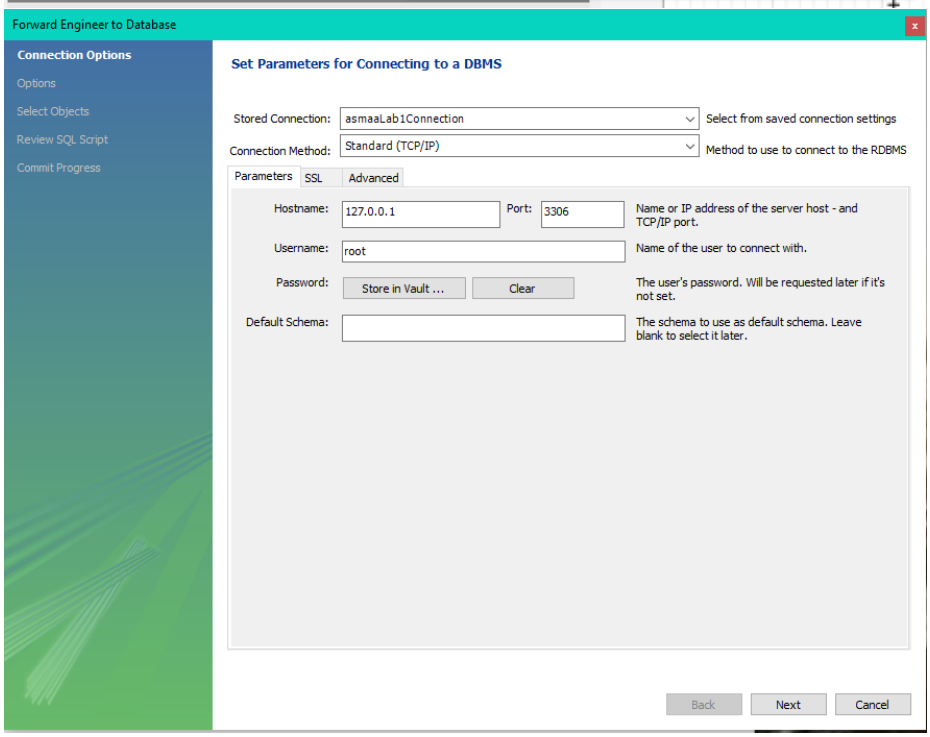
4. Generate the database tables in the schema and the creation DDL scripts.

Here, the 7 tables were generated:



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





Forward Engineer to Database

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
- [x] Include model attached scripts

Back Next Cancel

Forward Engineer to Database

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.

- [x] 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:

```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION

-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------
-- -----------------------------------------------------
-- Schema SAMPLE_Asmaa_CommuDepartment
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Schema SAMPLE_Asmaa_CommuDepartment
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `SAMPLE_Asmaa_CommuDepartment` ;
USE `SAMPLE_Asmaa_CommuDepartment` ;

-- -----------------------------------------------------
-- Table `SAMPLE_Asmaa_CommuDepartment`.`PUBLISHER`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_CommuDepartment`.`PUBLISHER` (
  `PublisherName` VARCHAR(45) NOT NULL,
  `Address` VARCHAR(100) NULL,
  `Phone` VARCHAR(13) NULL,
  PRIMARY KEY (`PublisherName`),

ENGINE = InnoDB;



-- -----------------------------------------------------
-- Table `SAMPLE_Asmaa_CommuDepartment`.`BOOK`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_CommuDepartment`.`BOOK` (
  `BookID` INT NOT NULL AUTO_INCREMENT,
  `Title` VARCHAR(45) NOT NULL,
  `PublisherName` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`BookID`),
  UNIQUE INDEX `BookID_UNIQUE` (`BookID` ASC) VISIBLE,
  INDEX `PublisherName_idx` (`PublisherName` ASC) VISIBLE,
  CONSTRAINT `PublisherName`
    FOREIGN KEY (`PublisherName`)
    REFERENCES `SAMPLE_Asmaa_CommuDepartment`.`PUBLISHER` (`PublisherName`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;



-- -----------------------------------------------------
-- Table `SAMPLE_Asmaa_CommuDepartment`.`BOOK_AUTHORS`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_CommuDepartment`.`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_Asmaa_CommuDepartment`.`BOOK` (`BookID`)
62        ON DELETE NO ACTION
63        ON UPDATE NO ACTION)
64    ENGINE = InnoDB;
65
66
67    -- -----------------------------------------------------
68    -- Table `SAMPLE_Asmaa_CommuDepartment`.`LIBRARY_BRANCH`
69    -- -----------------------------------------------------
70    CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_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_Asmaa_CommuDepartment`.`BOOK_COPIES`
81    -- -----------------------------------------------------
82    CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_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_Asmaa_CommuDepartment`.`BOOK` (`BookID`)
90        ON DELETE NO ACTION
91        ON UPDATE NO ACTION,
92      CONSTRAINT `BranchID`
93        FOREIGN KEY (`BranchID`)
94        REFERENCES `SAMPLE_Asmaa_CommuDepartment`.`LIBRARY_BRANCH` (`BranchID`)
95        ON DELETE NO ACTION
96        ON UPDATE NO ACTION)
97    ENGINE = InnoDB;
98
99
100   -- -----------------------------------------------------
101   -- Table `SAMPLE_Asmaa_CommuDepartment`.`BORROWER`
102   -- -----------------------------------------------------
103   CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_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_Asmaa_CommuDepartment`.`BOOK_LOANS`
116    -- -----------------------------------------------------
117  CREATE TABLE IF NOT EXISTS `SAMPLE_Asmaa_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_Asmaa_CommuDepartment`.`BOOK` (`BookID`)
130      ON DELETE NO ACTION
131      ON UPDATE NO ACTION,
132    CONSTRAINT `BranchID`
133      FOREIGN KEY (`BranchID`)
134      REFERENCES `SAMPLE_Asmaa_CommuDepartment`.`LIBRARY_BRANCH` (`BranchID`)
135      ON DELETE NO ACTION
136      ON UPDATE NO ACTION,
137    CONSTRAINT `CardNo`
138      FOREIGN KEY (`CardNo`)
139      REFERENCES `SAMPLE_Asmaa_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
```

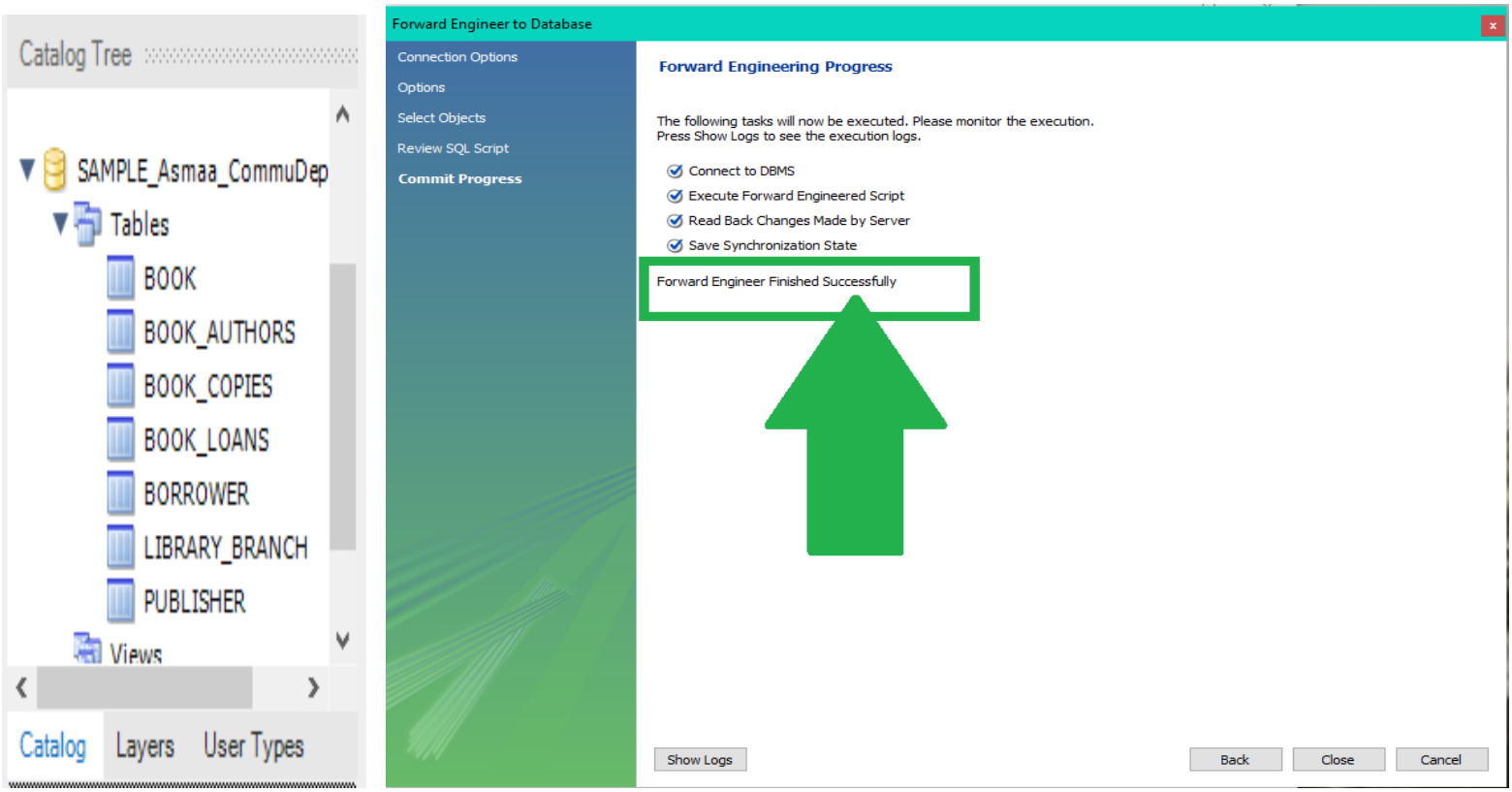
5. **Write the steps and show the ERD, DDL scripts, created tables.**

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.

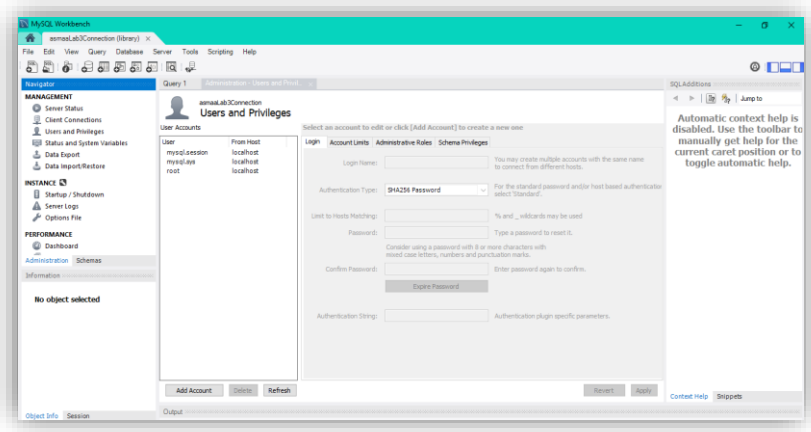


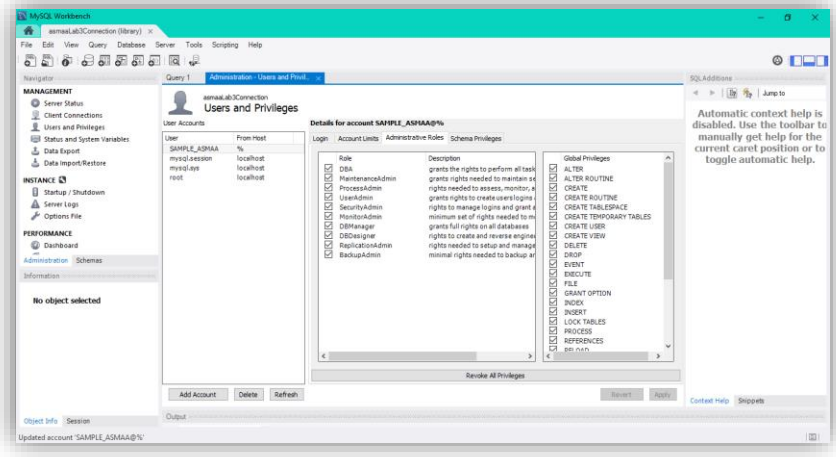
7. Save the generated scripts to your hard disk.

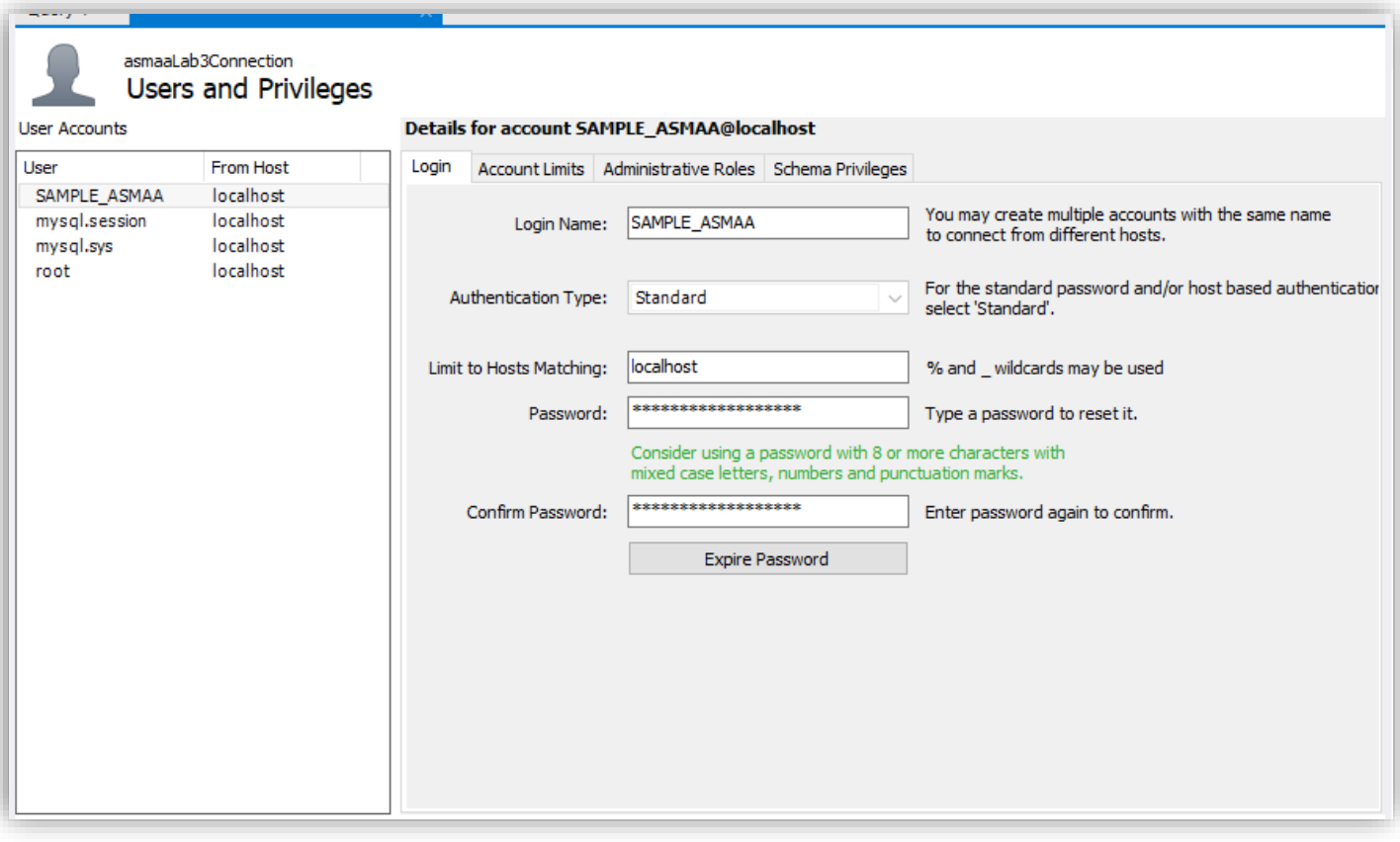
asmaa_lab3_DDL script

- # Database Administration:

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







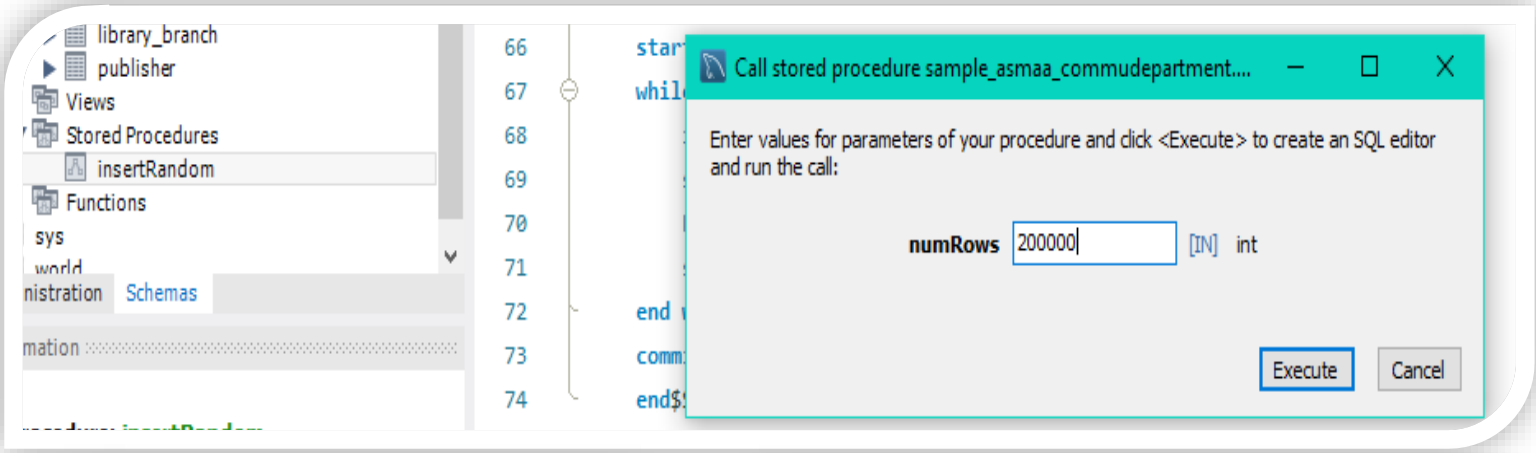
9. Write a simple program or use an offline tool, to insert about 200,000 records or more for each table

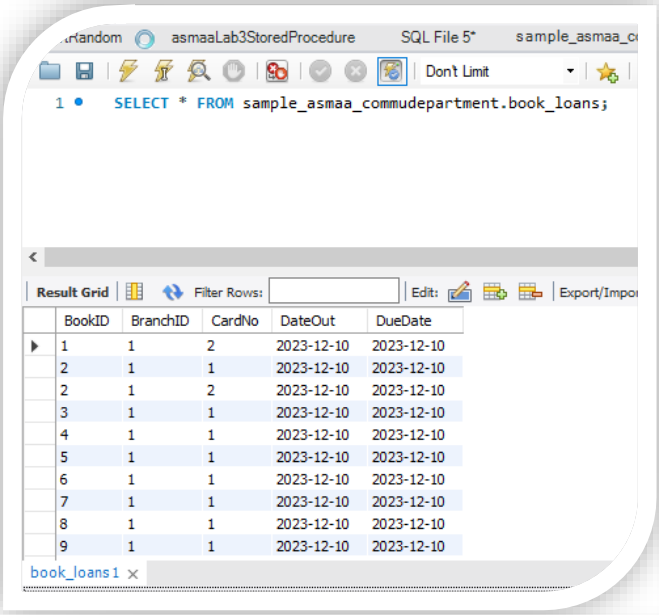
Code:

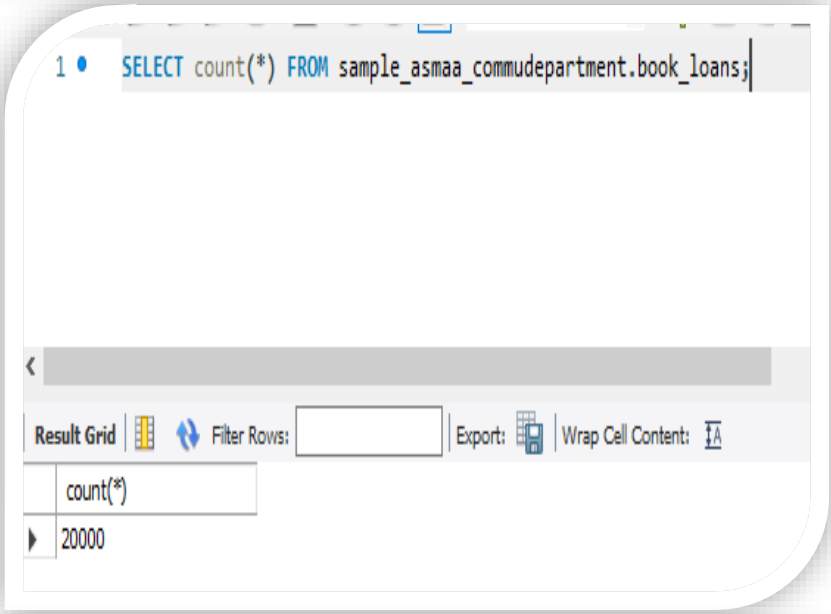
```
DELIMITER $$
/*
Delimiters other than the default ; are typically used when defining functions,
stored procedures, and triggers wherein you must define multiple statements.
defining a different delimiter like $$ which is used to define the end of the entire procedure,
but inside it, individual statements are each terminated by ;.
*/
create procedure insertRandom(in numRows int)
begin
    declare i int ;
    declare pName varchar(45) ;
    declare count int default 0;
    set i=1;
    start transaction;
    loop1: while i<= numRows do
        set pName= CONCAT(left(MD5(RAND()), 39),i);
        select count(*) into count from publisher where 'PublisherName'=pName;
        if count > 0 then
            iterate loop1;
        end if;
        insert into publisher values (pName, left(MD5(RAND()),100), left(MD5(RAND()),13) );
        set i=i+1;
    end while;
    commit;

    set i=1;
    start transaction;
    while i<= numRows do
        insert into book(  Title, PublisherName)
        select left(MD5(RAND()), 45), PublisherName
        FROM publisher limit i,1;
        set i=i+1;
    end while;
    commit;


    set i=1;
    start transaction;
    while i<= numRows do
        set pName= CONCAT(left(MD5(RAND()), 39),i);
        select count(*) into count from book_authors where 'AuthorName'=pName;
        insert into book_authors(AuthorName,BookID)
        select pName , BookID from book limit i,1;
        set i=i+1;
    end while;
    commit;
```

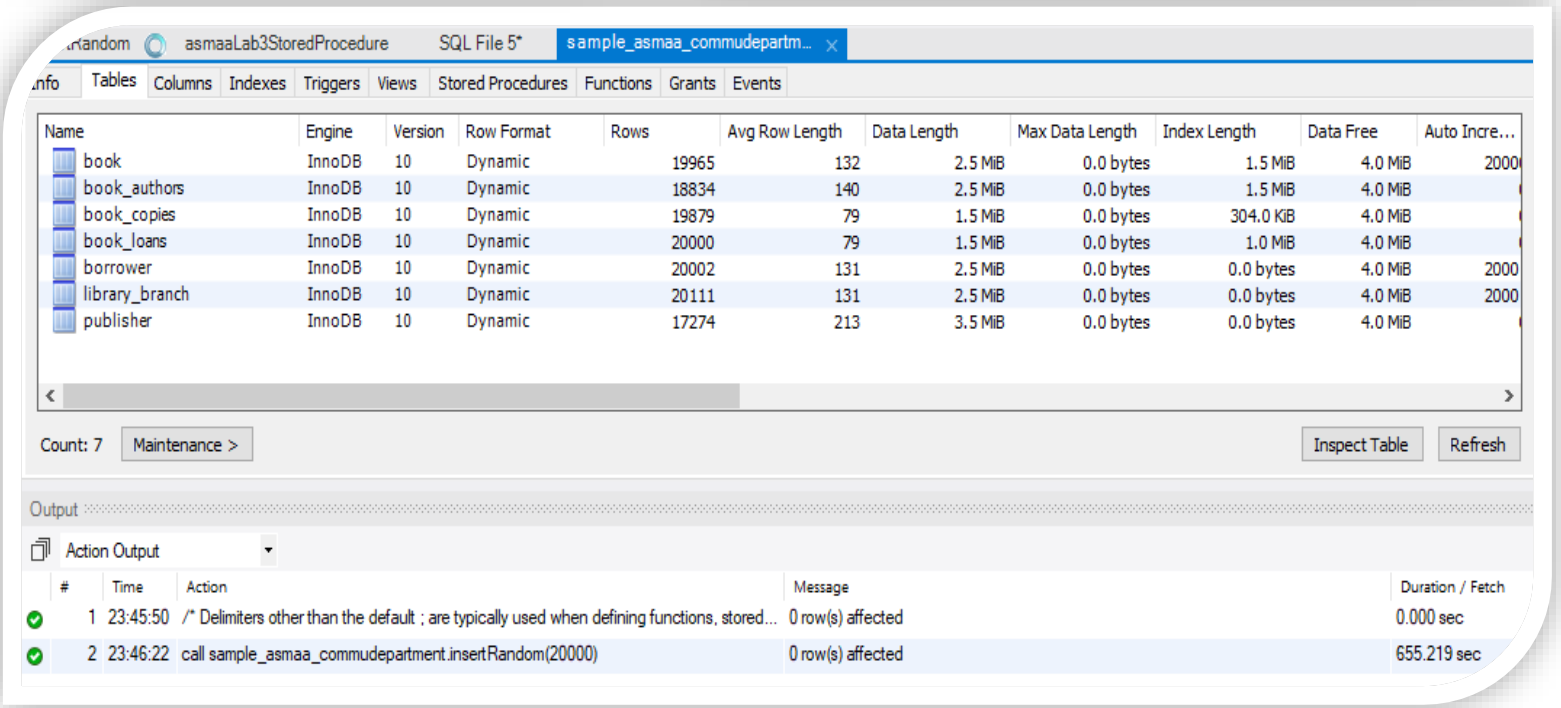
```
47
48      set i=1;
49      start transaction;
50      while i<= numRows do
51          insert into library_branch(BranchName,Address)
52          Values(left(MD5(RAND()),45), left(MD5(RAND()),100));
53          set i=i+1;
54      end while;
55      commit;
56
57      set i=1;
58      start transaction;
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    /* Finally, reset the delimiter to the default ; */
87    DELIMITER ;
88
```

Steps:


library_branch
publisher
Views
Stored Procedures
insertRandom
Functions
sys
Schemas
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


asmaaLab3StoredProcedure
SQL File 5*
SELECT * FROM sample_asmaa_commudepartment.book_loans;
Result Grid
Filter Rows:
Edit:
Export/Impo
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
book_loans 1


SELECT count(*) FROM sample_asmaa_commudepartment.book_loans;
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
count(*)
20000


asmaaLab3StoredProcedure
SQL File 5*
sample_asmaa_commudepartm...
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

10. Using the SAMPLE account, report the storage details about: data files, free space, offline table spaces, table space quotas, and table space segments before and after the insertion.

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	

11. Compact the database tables storage, and report whether it recovered the unused spaces or not.

asmaaLab3AdministrationQuesti... sample_asmaa_commudepartm...

Don't Limit

```
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_commudepartment"
18 order by(data_length+ index_length) DESC;
```

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

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

Result Grid | Form Editor | Read Only

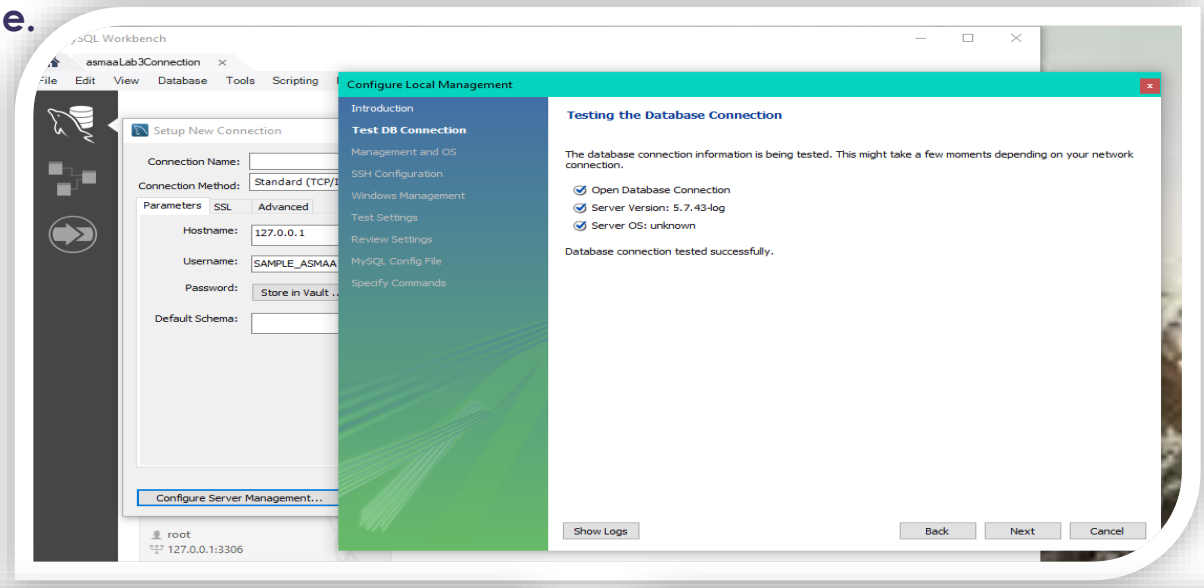
Result 1

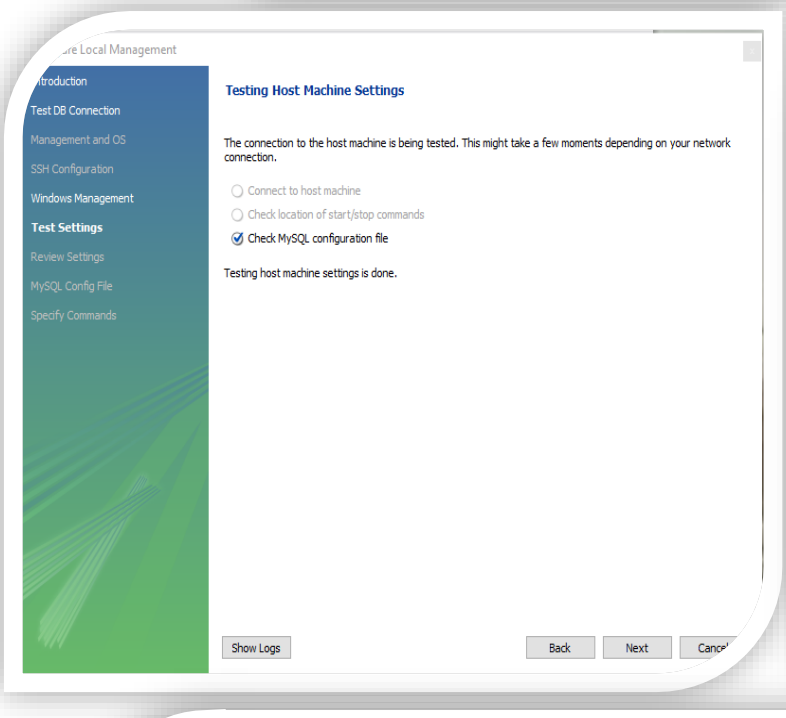
Output

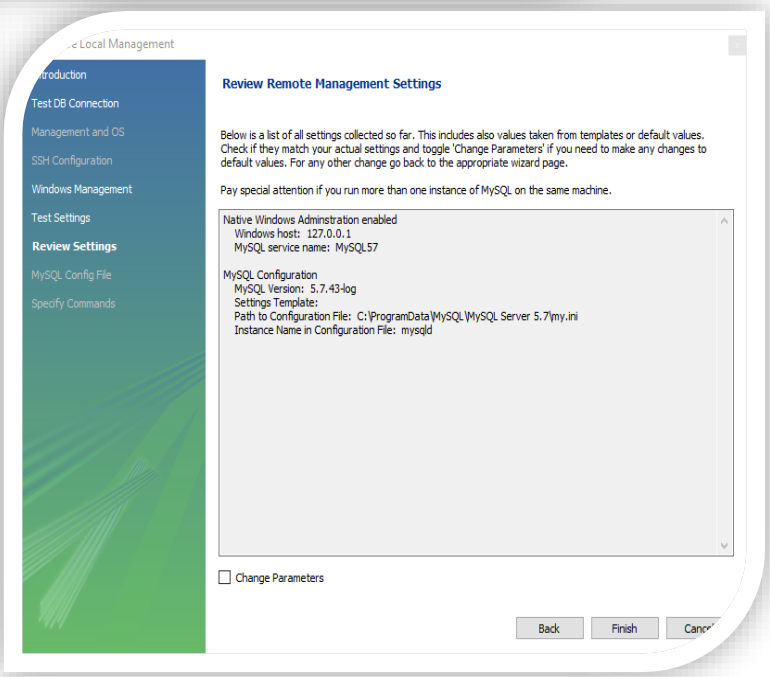
Action Output

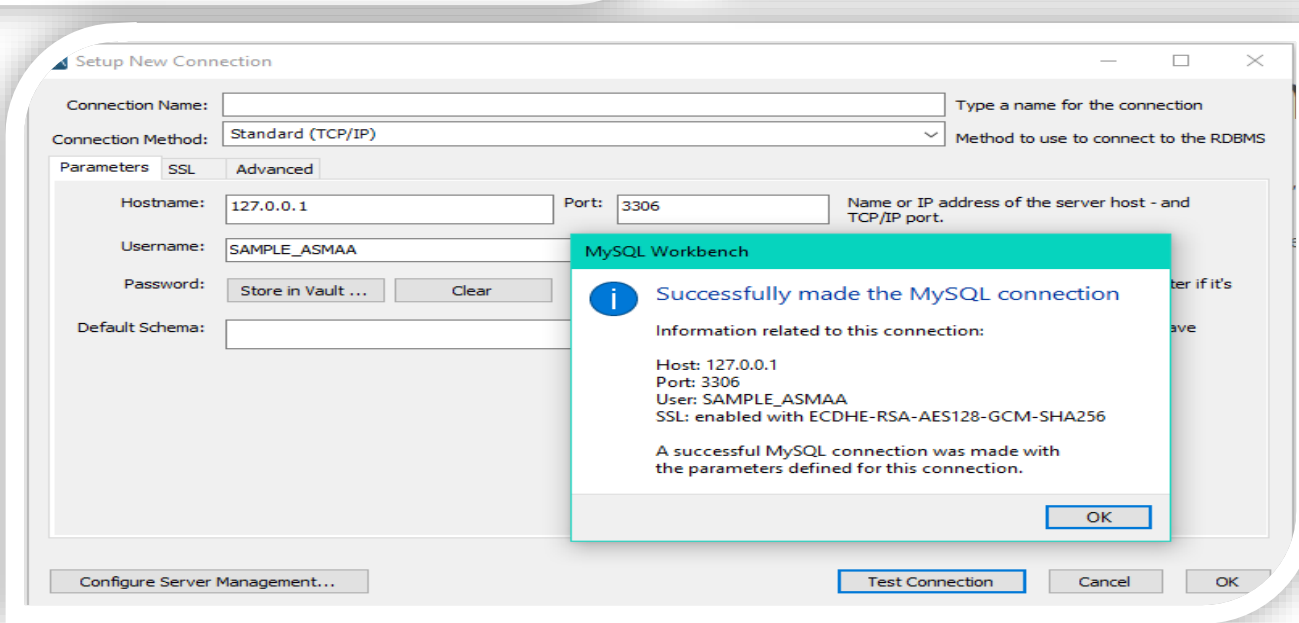
#	Time	Action	Message	Duration / Fetch
6	01:04:29	alter table library_branch row_format= c...	0 row(s) affected Records: 0 Duplicate...	0.500 sec
7	01:04:29	alter table publisher row_format= compr...	0 row(s) affected Records: 0 Duplicate...	0.453 sec

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.









13. Select the publishers that have phone numbers ended with '39' and show this result.

ANS) There are 64 matched row records.

asmaaLab3AdministrationQuesti... | publisher

Don't Limit

```
1 •  SELECT * FROM sample_asmaa_commudepartment.publisher where Phone like '%\39';
```

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

PublisherName	Address	Phone
0064f23bb69986caef1226316268ea968072	68160febf11661c5ffbab8578a40cd9e	c4c59a2539639
010f1fea468114b4b87f05d643bf6dd58798	d31991c390f0c5b4d79ae87719936d08	3e2bf62f21a39
06c8c9549dba2c6a0d96e34277df74974326	182dd1695233248138d0f14fde836a39	8d2c373d13e39
070a46ed3041af1465b799b8de69262718352	4a50373a8416fbc1af1ebbd8eee58209	c53c44c3a1e39
0bb61fef3260e5678514fc891818ff5419440	551a626e773ca7c71d759f955a4e746d	1dc9762c6c439
10b82f98da43705287ec83b85119e32117918	c6b5a80d8d5142b4b0d1507c9bc7083c	1420d767d0339
11a338e6ef5a20294c454ba4db7f45733353	201ed1434ad6f60a0850231842f543e6	3fc4b41a8d239
1b2d830e4307aa2d74f42c5ce25e552016865	fe29a6d9dc6037a99dc53f171b1228dd	8f5ea2569e139
243847fe1b10cc45c4c1b5ce1f024f0c6692	26e0f5b601e5f0ec0d21fe6fb53cf5c7	b2278d7e35039
267d7ace7bc196191f9aa6eefe53cfc49133	5866a24edeaa00cbb56c6fde9857d4fa	6b36987a65539
2b52ec44c2a8e9ab5c787b23f2fb6b652109	bf1d573c4963af46bb73c2a4f3849e3b	e7611a5ac7d39

publisher 2

Output

Action Output

#	Time	Action	Message
1	13:16:17	SELECT * FROM sample_asmaa_commudepartment.publisher	20000 row(s) returned
2	13:16:44	SELECT * FROM sample_asmaa_commudepartment.publisher where Phone like '%...	64 row(s) returned

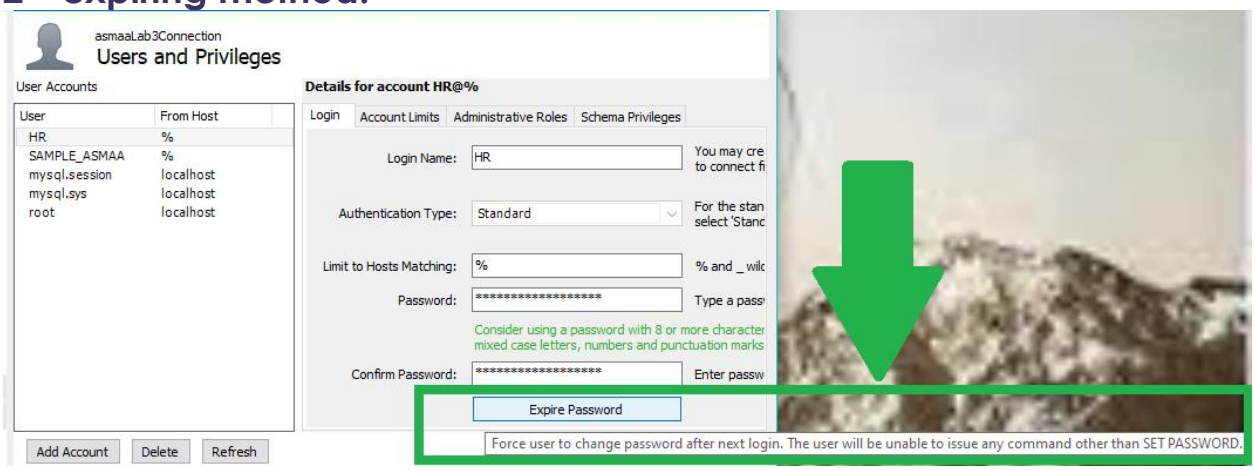
14. Expire the password of the HR account and drop the SAMPLE one.

1ST expiring method:

```
27 •  alter user 'HR'@'%' password expire;
```

3	13:26:24	alter user 'HR'@'%' password expire	0 row(s) affected	0.000 sec

2nd expiring method:



Dropping:

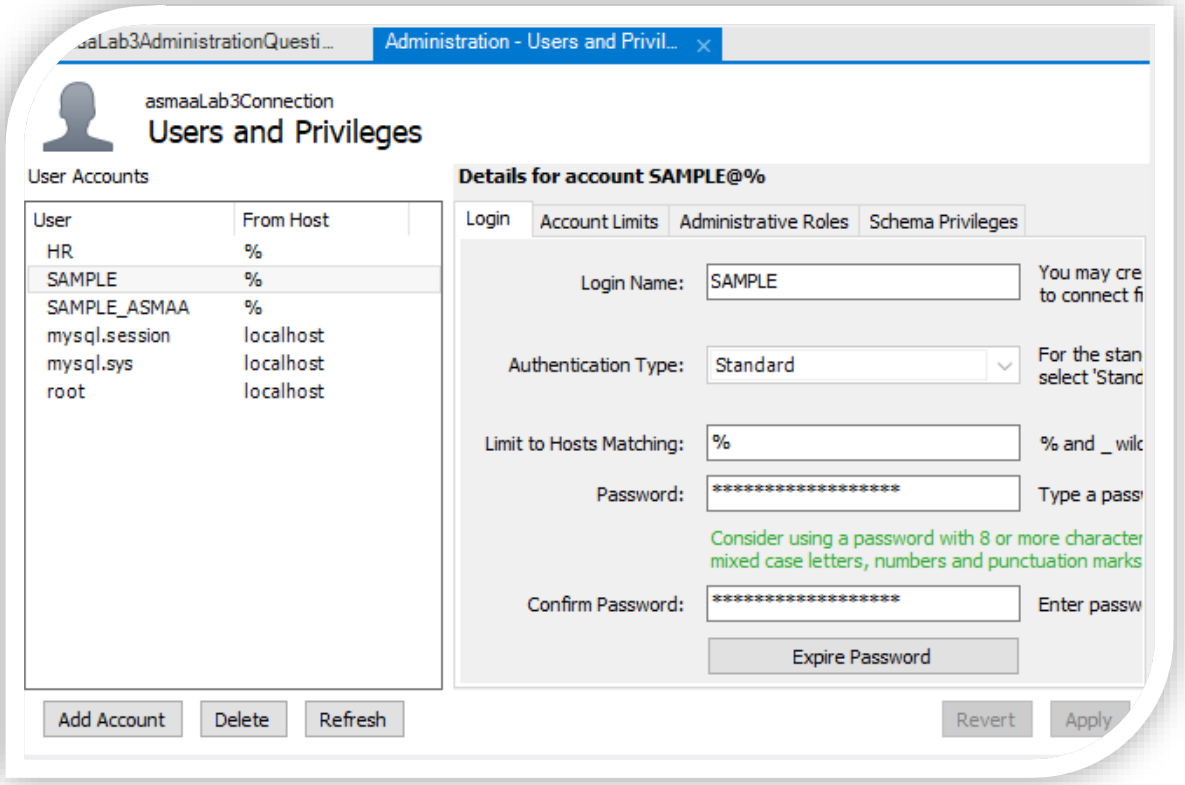
```
28 •  drop user 'SAMPLE' @'%';
```

Output

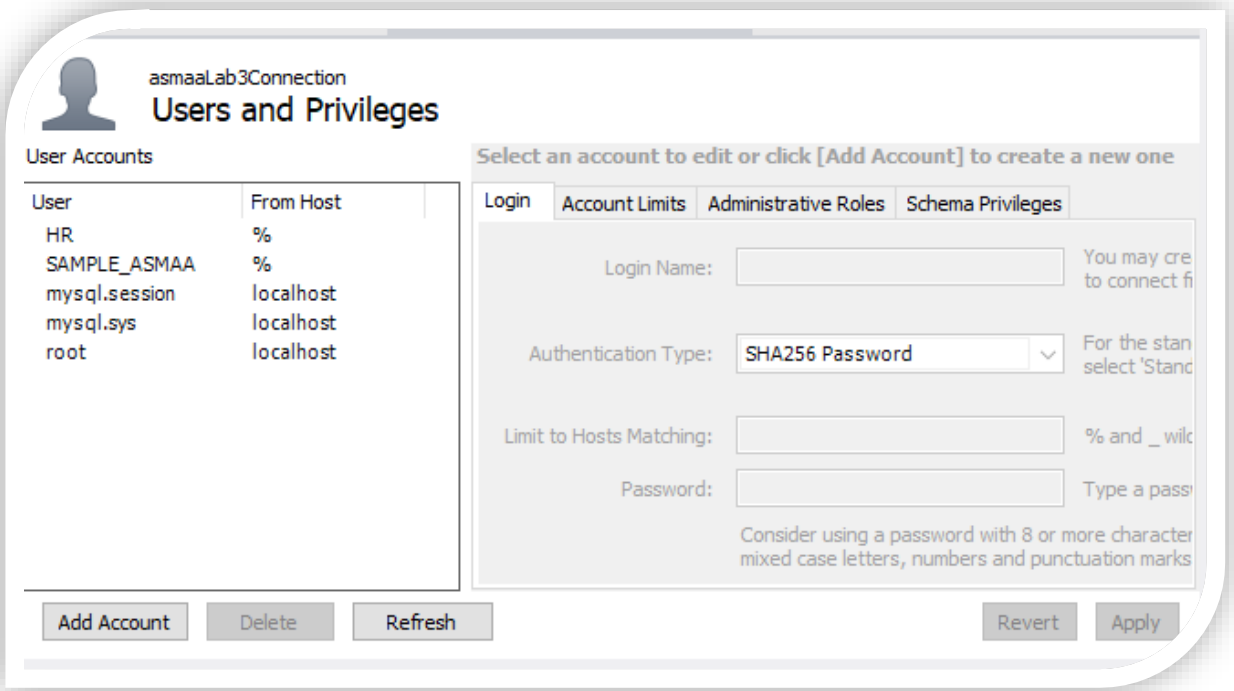
Action Output

#	Time	Action	Message	Duration / Fetch
1	13:16:17	SELECT * FROM sample_asmaa_comm...	20000 row(s) returned	0.016 sec / 0.063 sec
2	13:16:44	SELECT * FROM sample_asmaa_comm...	64 row(s) returned	0.015 sec / 0.000 sec
3	13:26:24	alter user 'HR'@'%' password expire	0 row(s) affected	0.000 sec
4	13:29:49	drop user 'SAMPLE' @'%'	0 row(s) affected	0.000 sec

BEFORE dropping:



AFTER dropping:



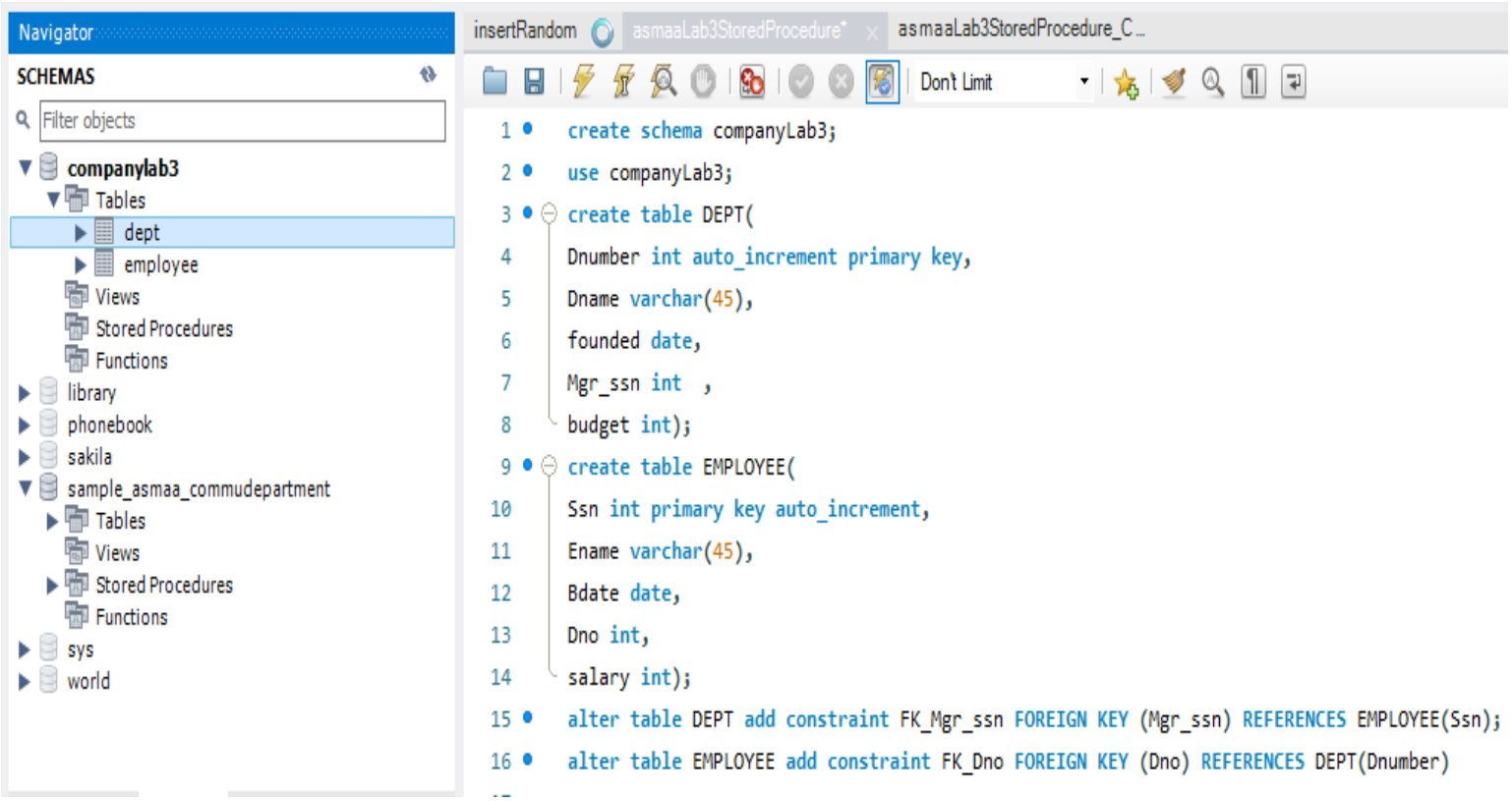
- # Stored Procedures

15. Consider the following database schema:

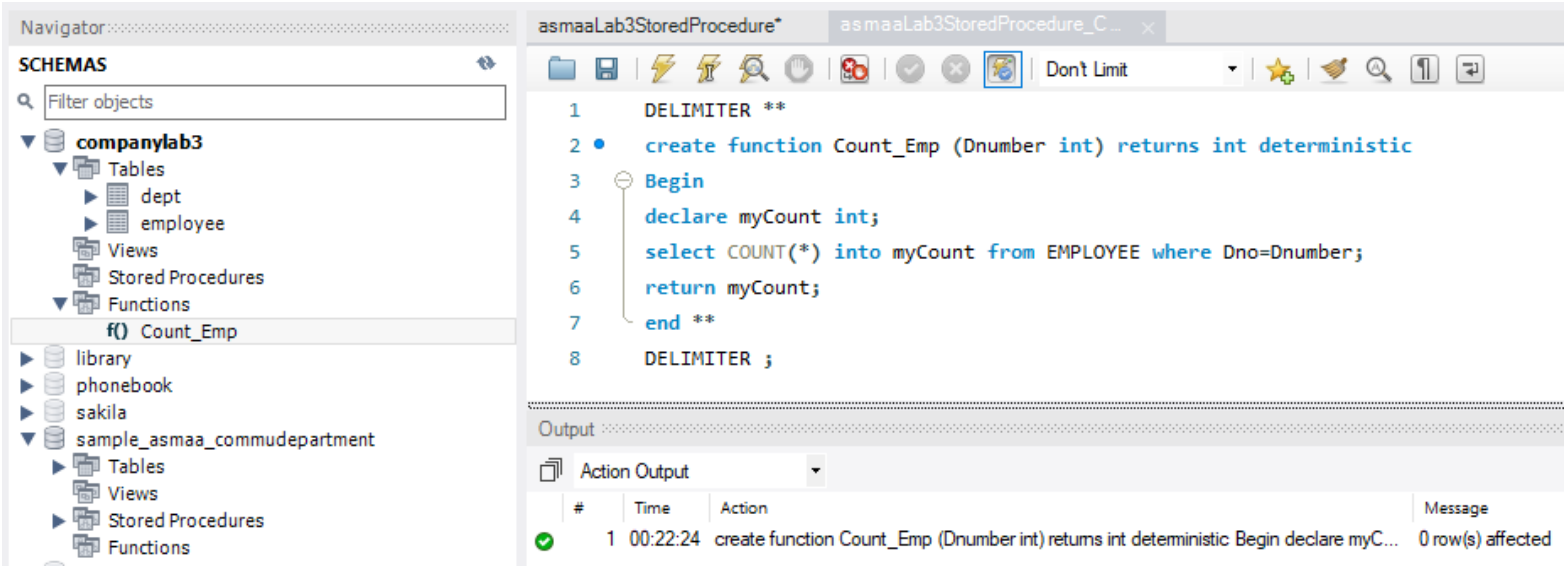
DEPT (Dnumber , Dname, Founded, Mgr_ssn, Budget)

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.



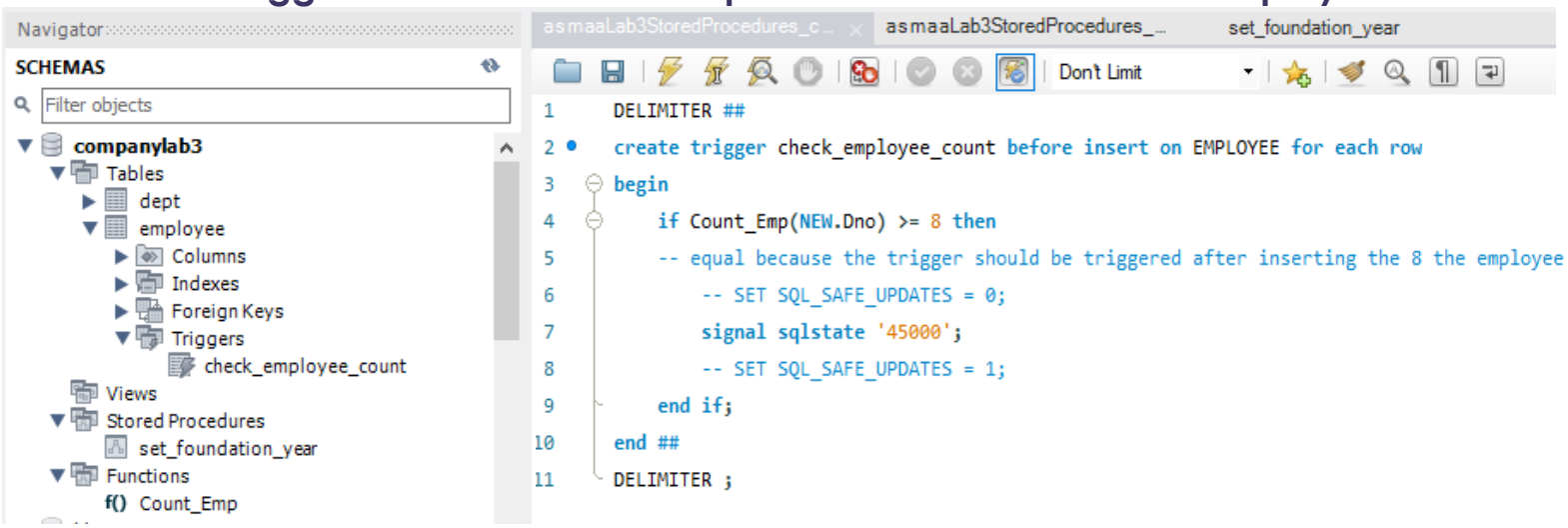
16. Create a stored function Count_Emp (Dnumber NUMBER) that returns the number of employees working for the department Dnumber.



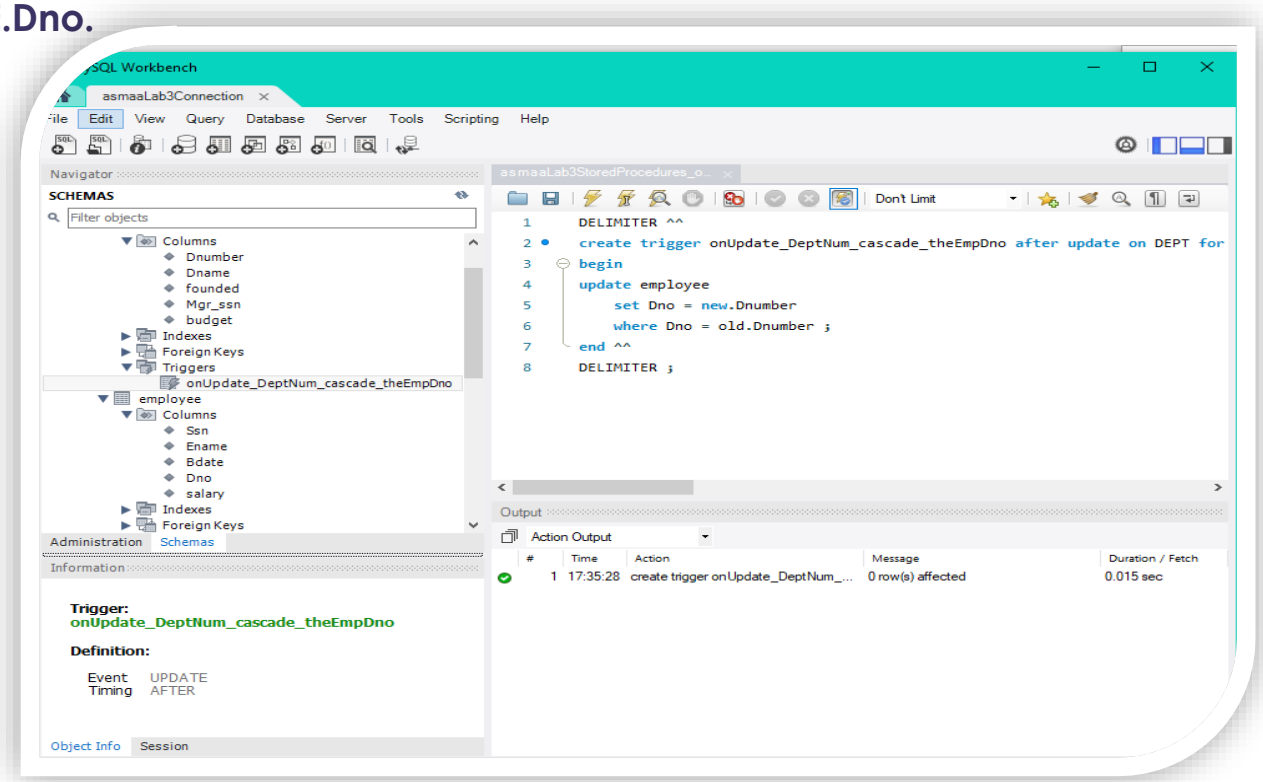
17. Create a stored procedure that ensures that Year(DEPT.Founded) >=1960 for all departments; if a row violates this constraint then set its date to be '01-JAN-1960'

```
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 ;
```

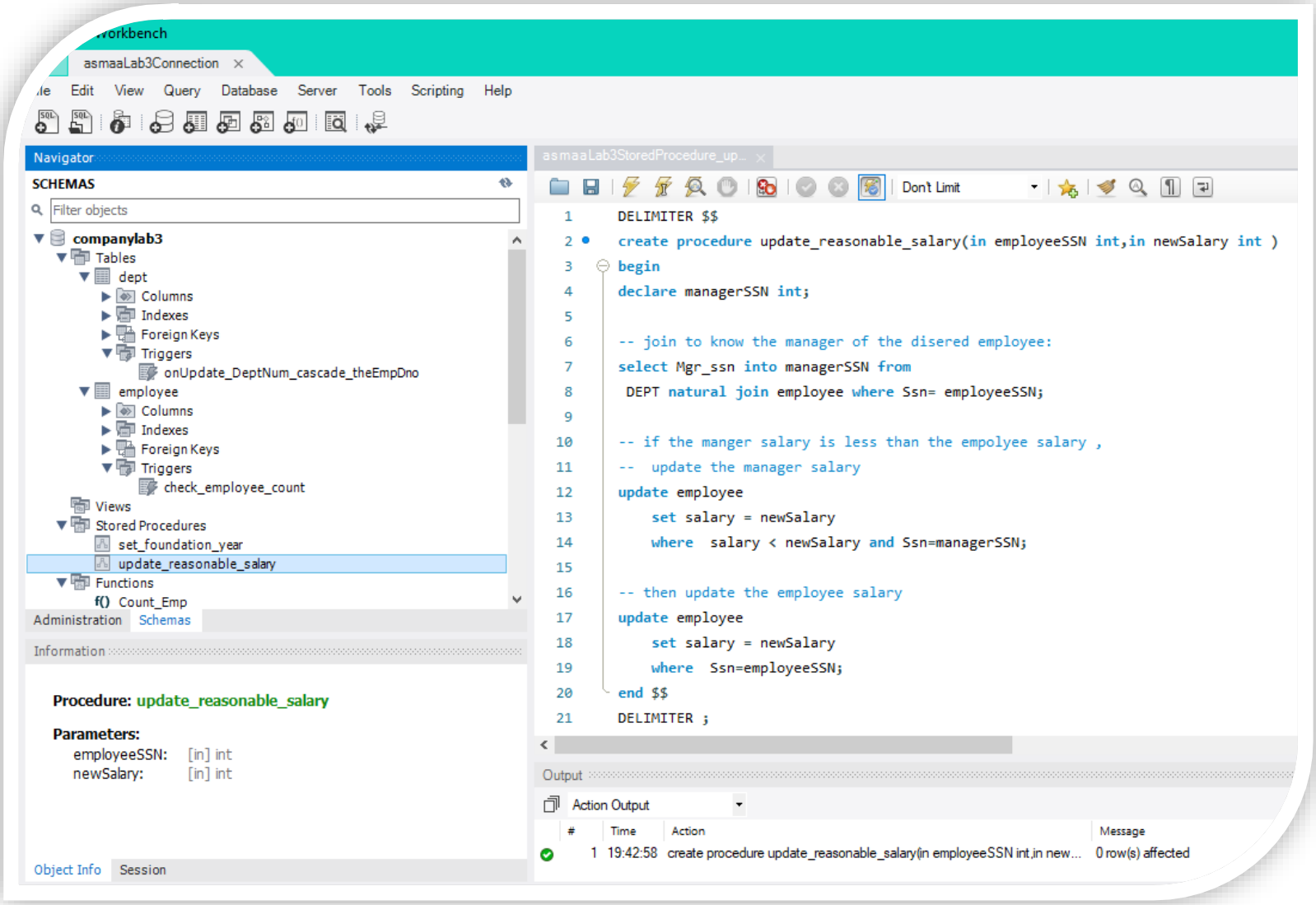
18. Create a trigger to ensure that no department has more than 8 employees.



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



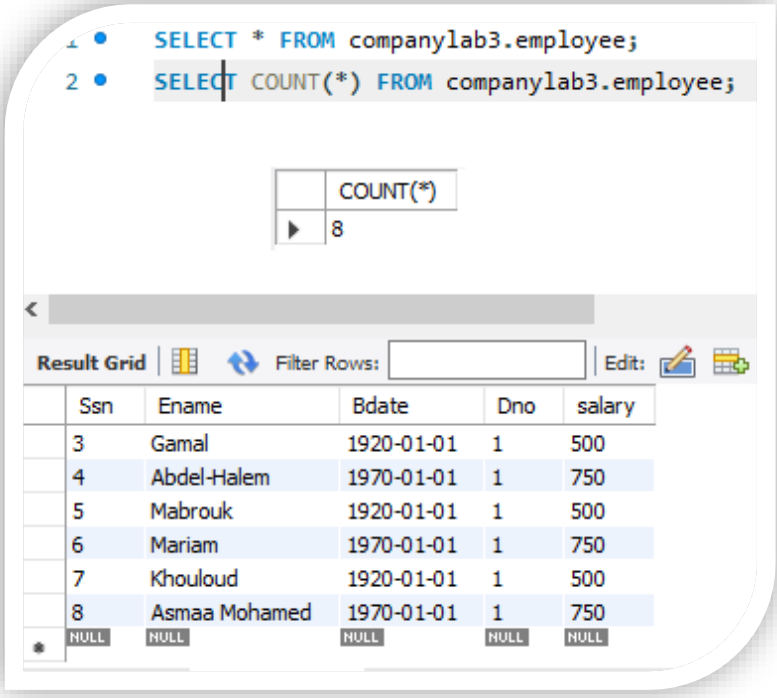
20. Create a trigger to ensure that whenever an employee is given a raise in salary, his department manager's salary must be increased to be at least as much.

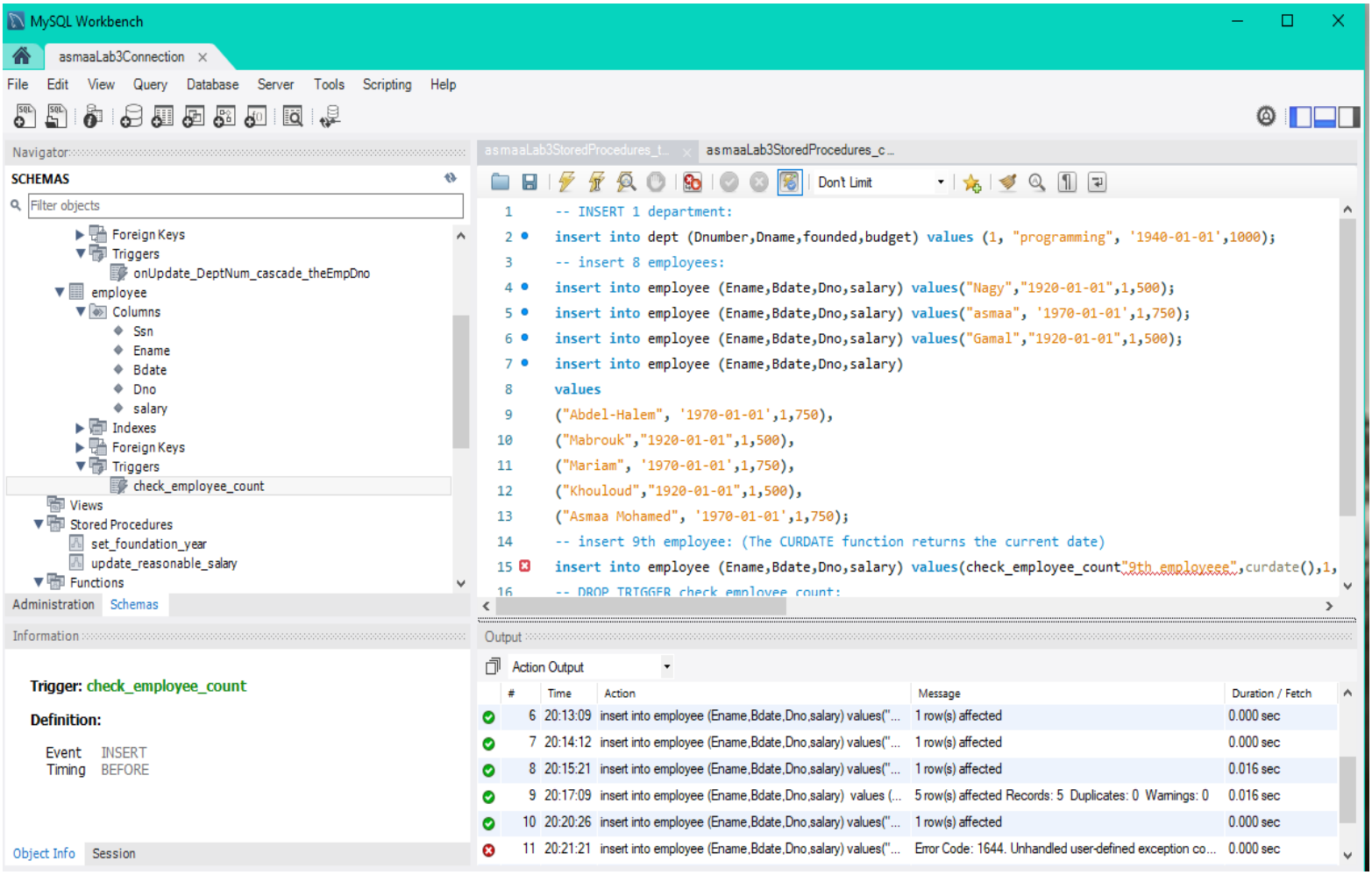


21. Test the created functions/procedures/triggers by suitable SQL statements to ensure their correctness.

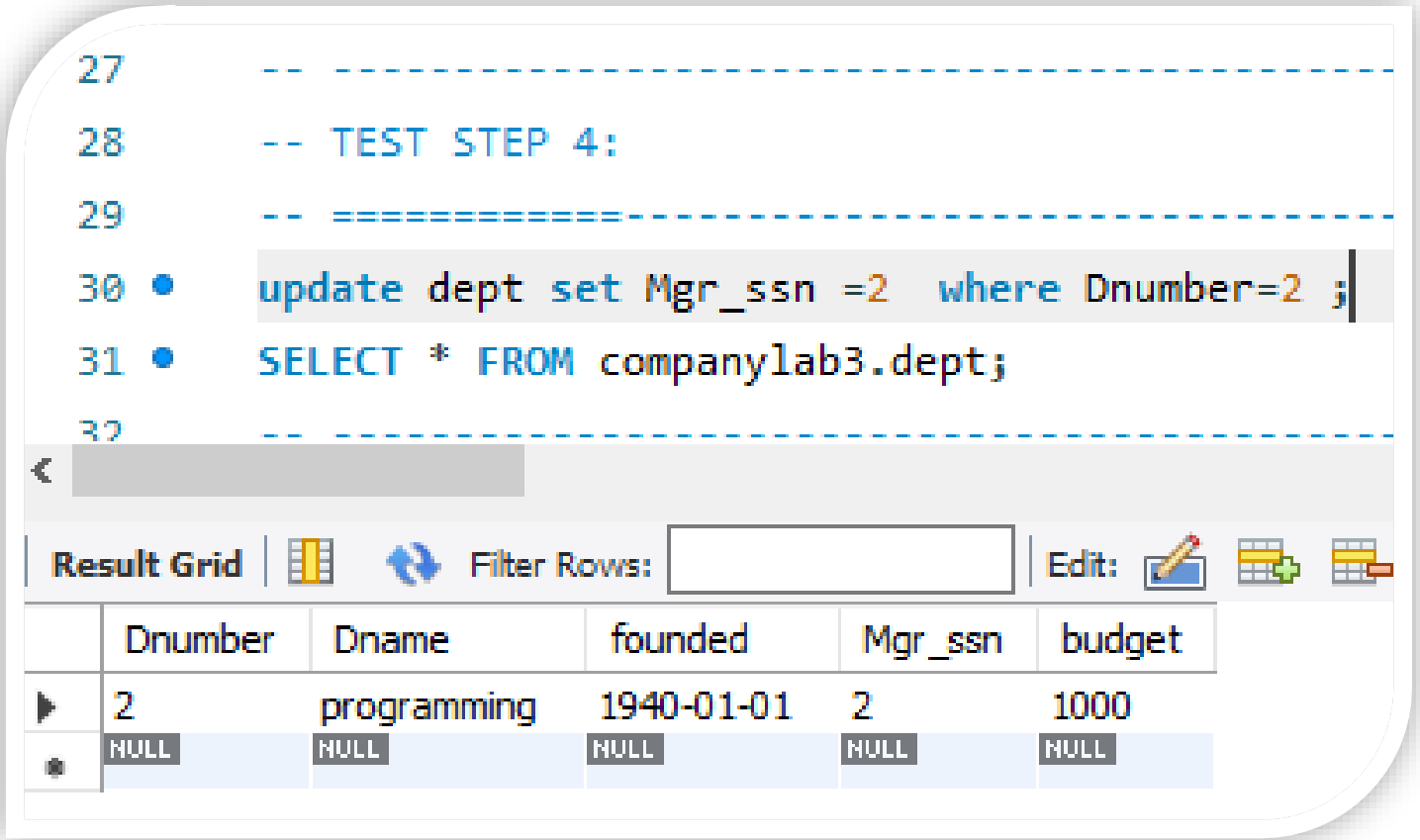
ANS) testing steps :

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.

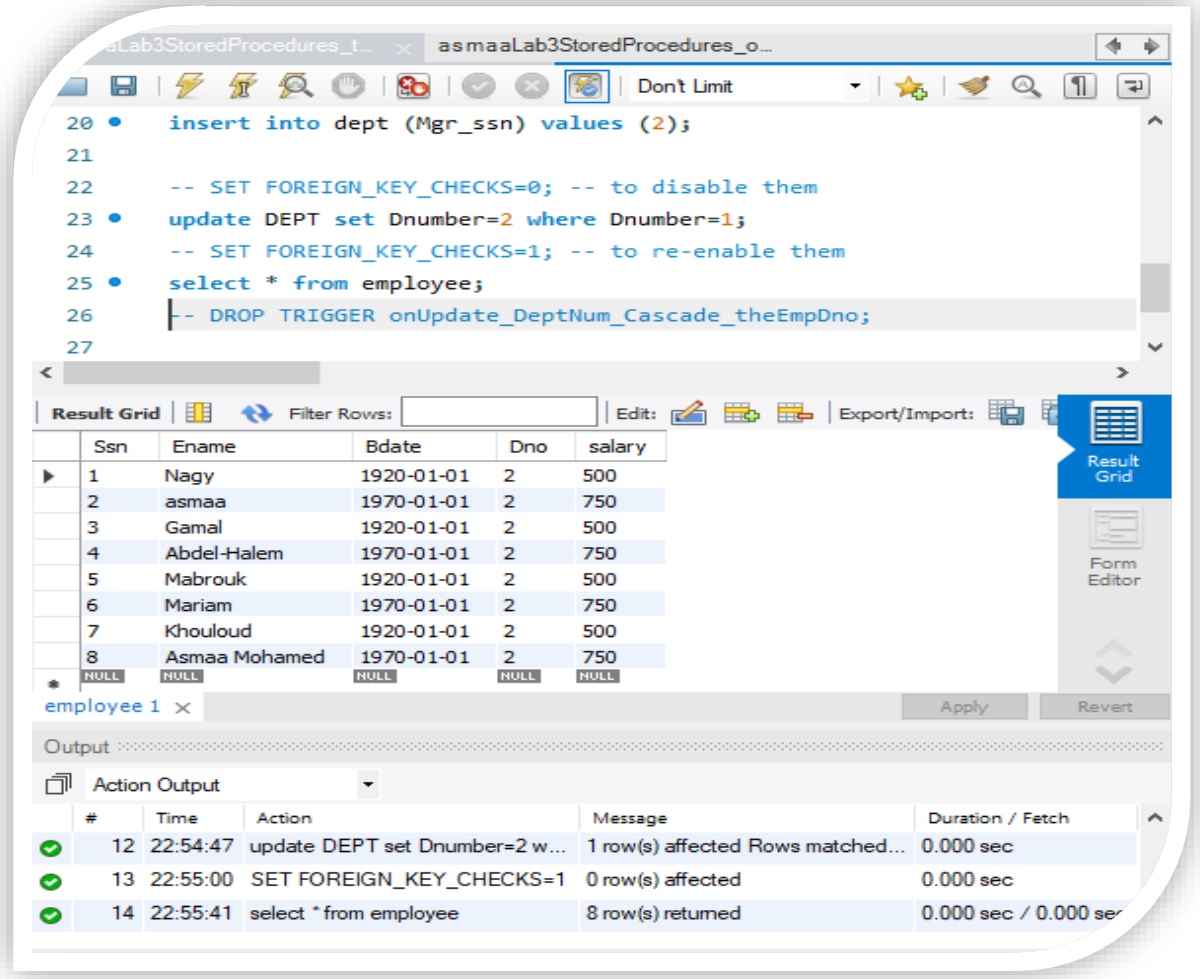




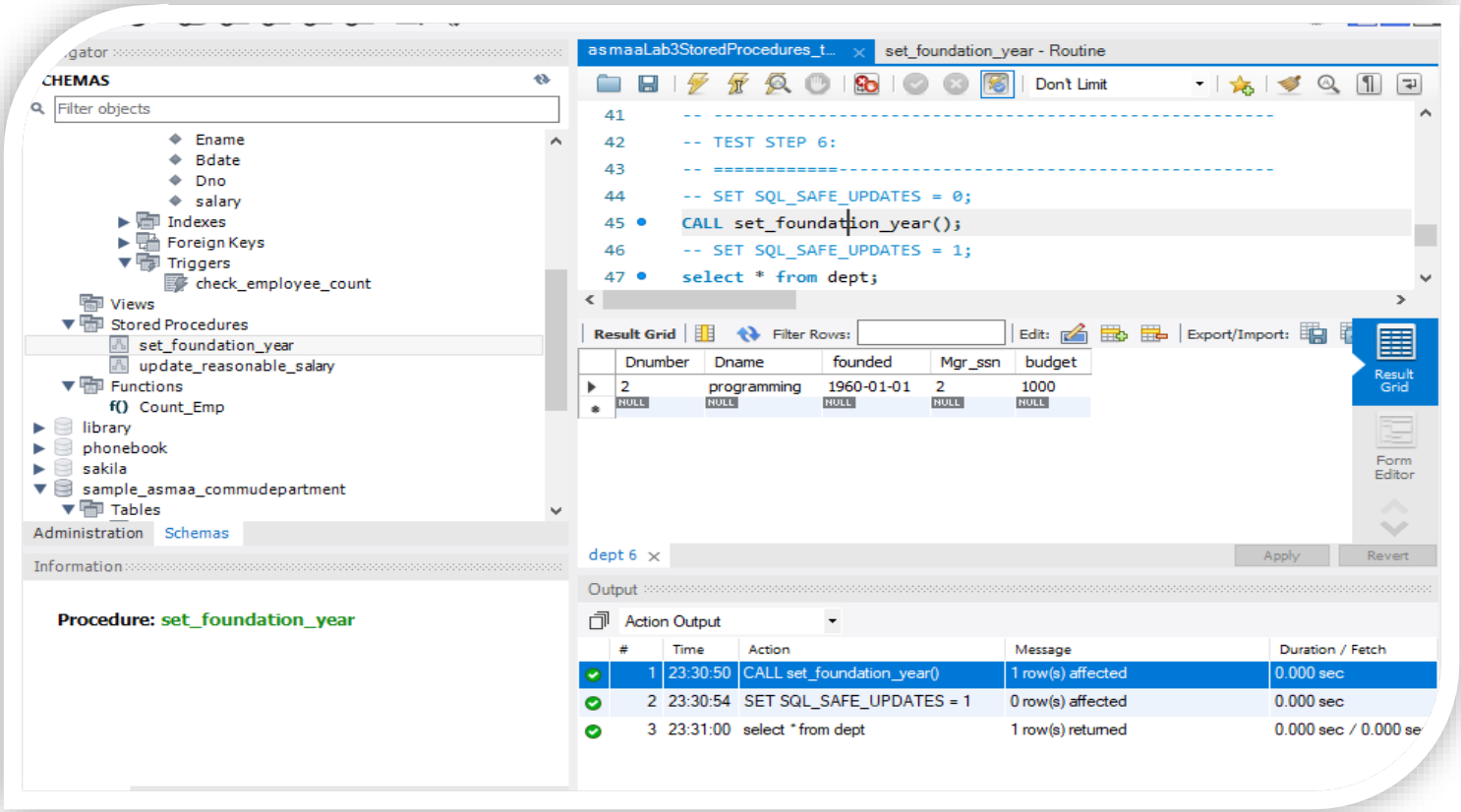
4. Insert the foreign key in the DEPT table:



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



6. Call set_foundation_year to update Founded date from 1940-01-01 to 1960-01-01.



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: