Introduction to SQL

1 Introduction to SQL

What is SQL?

Structured Query Language

- Language to communicate with database!
- 관계형 데이터베이스 시스템(RDBMS)에서 데이터를 관리 및 처리하기 위해 설계된 언어

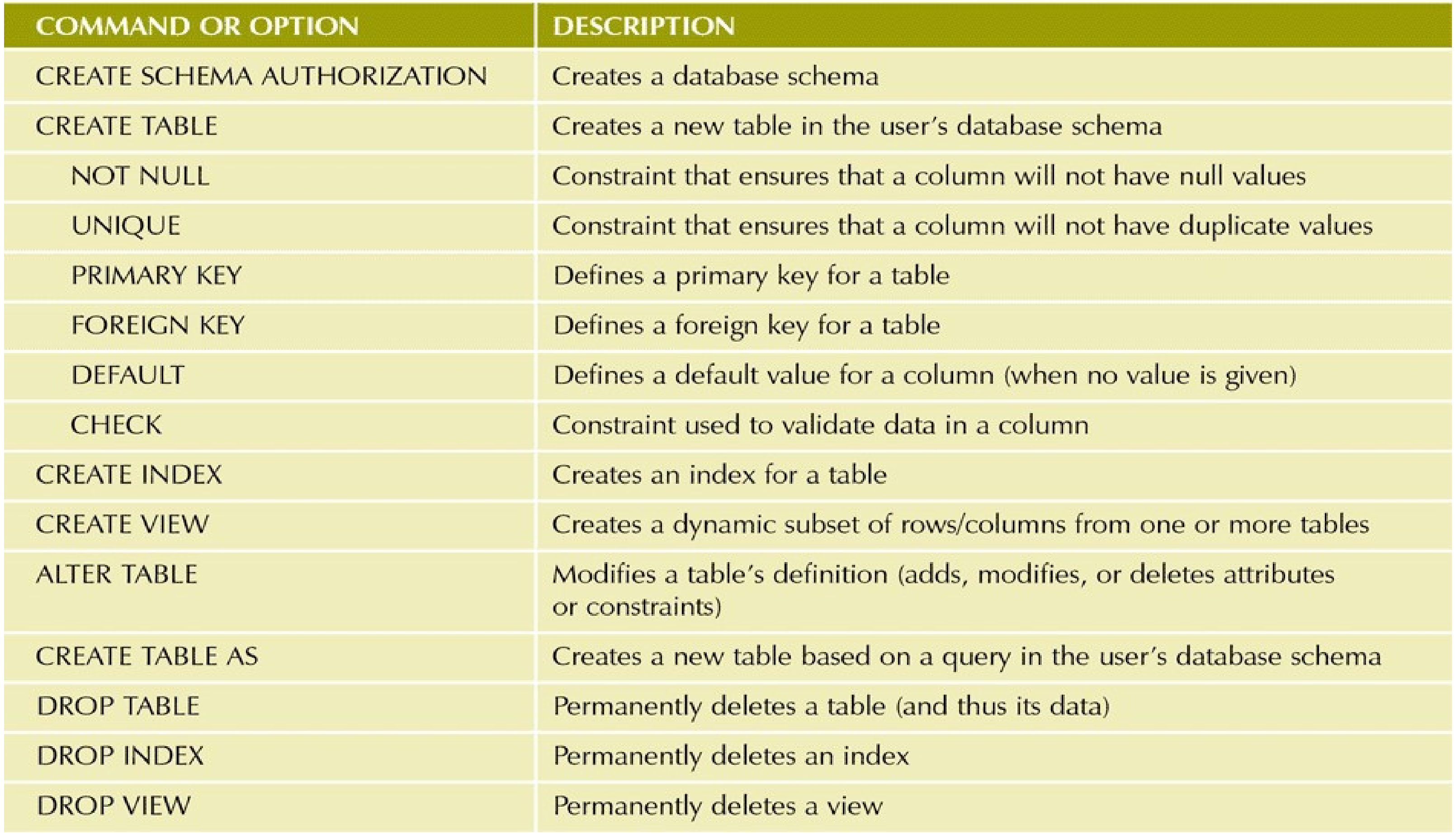
Characteristics of SQL

- SQL is relatively easy to learn
- Basic command set has a vocabulary of less than 100 words
- American National Standards Institute (ANSI) prescribes a standard SQL (표준)
- Several SQL dialects exist (ANSI SQL을 기반으로 개별 SQL을 개발)
- 대소문자 구분 없음. 명령은 반드시 세미콜론(;)으로 끝남.

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

SQL Data Definition Commands (데이터 정의)

- Commands to create database objects and to define access rights to those database objects



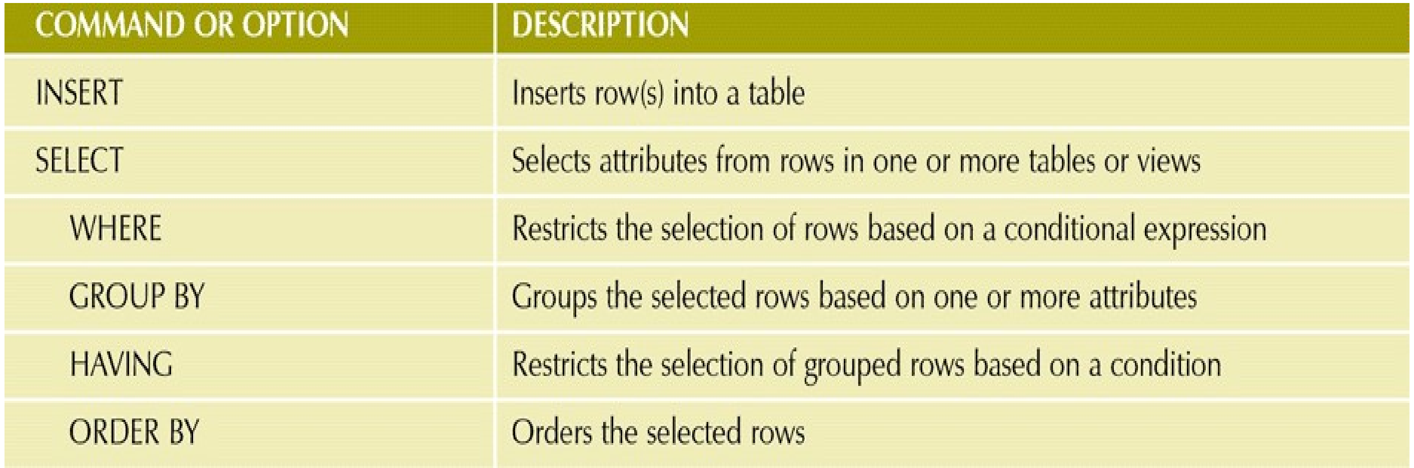
COMMAND OR OPTION	DESCRIPTION
CREATE SCHEMA AUTHORIZATION	Creates a database schema
CREATE TABLE	Creates a new table in the user's database schema
NOT NULL	Constraint that ensures that a column will not have null values
UNIQUE	Constraint that ensures that a column will not have duplicate values
PRIMARY KEY	Defines a primary key for a table
FOREIGN KEY	Defines a foreign key for a table
DEFAULT	Defines a default value for a column (when no value is given)
CHECK	Constraint used to validate data in a column
CREATE INDEX	Creates an index for a table
CREATE VIEW	Creates a dynamic subset of rows/columns from one or more tables
ALTER TABLE	Modifies a table's definition (adds, modifies, or deletes attributes or constraints)
CREATE TABLE AS	Creates a new table based on a query in the user's database schema
DROP TABLE	Permanently deletes a table (and thus its data)
DROP INDEX	Permanently deletes an index
DROP VIEW	Permanently deletes a view

Index: Utilized for fast operation of ordering, searching, etc.

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Data Manipulation Commands (데이터 조작)

- To insert, update, delete, and retrieve data within the database tables



COMMAND OR OPTION	DESCRIPTION
INSERT	Inserts row(s) into a table
SELECT	Selects attributes from rows in one or more tables or views
WHERE	Restricts the selection of rows based on a conditional expression
GROUP BY	Groups the selected rows based on one or more attributes
HAVING	Restricts the selection of grouped rows based on a condition
ORDER BY	Orders the selected rows

(참고) Data Control Commands (데이터 제어): GRANT(권한부여), REVOKE(권한회수) 등

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Creating the Database

- **Two tasks must be completed**
  - Create the database structure
  - Create the tables that will hold the end-user data

- **Create the database structure**
  - RDBMS creates the physical files that will hold the database
  - Interact with the operating system and the file systems
  - Tends to differ substantially from one RDBMS to another

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

The Database Schema

Schema

- Group of database objects—such as tables and indexes—that are related to each other
- Logical grouping of database objects

Authentication

- Process through which the DBMS verifies that only registered users are able to access the database
- Log on to the RDBMS using a user ID and a password created by the database administrator

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

2 SQL Tutorials

MySQL Installation

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Table: VENDOR

V_CODE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE	V_STATE	V_ORDER
21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
21226	SuperLoo, Inc.	Flushing	904	215-8995	FL	N
21231	D&E Supply	Singh	615	228-3245	TN	Y
21344	Gomez Bros.	Ortega	615	889-2546	KY	N
22567	Dome Supply	Smith	901	678-1419	GA	N
23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
24004	Brackman Bros.	Browning	615	228-1410	TN	N
24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
25443	B&K, Inc.	Smith	904	227-0093	FL	N
25501	Damal Supplies	Smythe	615	890-3529	TN	N
25595	Rubicon Systems	Orton	904	456-0092	FL	Y

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Table: PRODUCT

P_CODE	P_DESCRIPT	P_INDATE	P_QOH	P_MIN	P_PRICE	P_DISCOUNT	V_CODE
11QER/31	Power painter, 15 psi., 3-nozzle	2005-11-03	8	5	109.99	0.00	25595
13-Q2/P2	7.25-in. pwr. saw blade	2005-12-13	32	15	14.99	0.05	21344
14-Q1/L3	9.00-in. pwr. saw blade	2005-11-13	18	12	17.49	0.00	21344
1546-QQ2	Hrd. cloth, 1/4-in., 2x50	2006-01-15	15	8	39.95	0.00	23119
1558-QW1	Hrd. cloth, 1/2-in., 3x50	2006-01-15	23	5	43.99	0.00	23119
2232/QTY	B&D jigsaw, 12-in. blade	2005-12-30	8	5	109.92	0.05	24288
2232/QWE	B&D jigsaw, 8-in. blade	2005-12-24	6	5	99.87	0.05	24288
2238/QPD	B&D cordless drill, 1/2-in.	2006-01-20	12	5	38.95	0.05	25595
23109-HB	Claw hammer	2006-01-20	23	10	9.95	0.10	21225
23114-AA	Sledge hammer, 12 lb.	2006-01-02	8	5	14.40	0.05	
54778-2T	Rat-tail file, 1/8-in. fine	2005-12-15	43	20	4.99	0.00	21344
89-WRE-Q	Hicut chain saw, 16 in.	2006-02-07	11	5	256.99	0.05	24288
PVC23DRT	PVC pipe, 3.5-in., 8-ft.	2006-02-20	188	75	5.87	0.00	
SM-18277	1.25-in. metal screw, 25	2006-03-01	172	75	6.99	0.00	21225
SW-23116	2.5-in. wd. screw, 50	2006-02-24	237	100	8.45	0.00	21231
WR3/TT3	Steel matting, 4'x8'x1/6", .5" mesh	2006-01-17	18	5	119.95	0.10	25595

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Table: EMPLOYEE

EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_AREACODE	EMP_PHONE
Diante	Jorge	D	615	890-4567
Johnson	Edward	E	615	898-4387
Jones	Anne	M	615	898-3456
Kolmycz	George	D	615	324-5456
Lewis	Rhonda	G	615	324-4472
Saranda	Hermine	R	615	324-5505
Smith	George	A	615	890-2984
Smith	Jeanine	K	615	324-7883
Smythe	Melanie	P	615	324-9006
Washington	Rupert	E	615	890-4925
Wiesenbach	Paul	R	615	897-4358
Williams	Robert	D	615	890-3220
Brandon	Marie	G	901	882-0845
Genkazi	Leighla	W	901	569-0093
Lange	John	P	901	504-4430
Smith	George	K	901	504-3339
Vandam	Rhett		901	675-8993

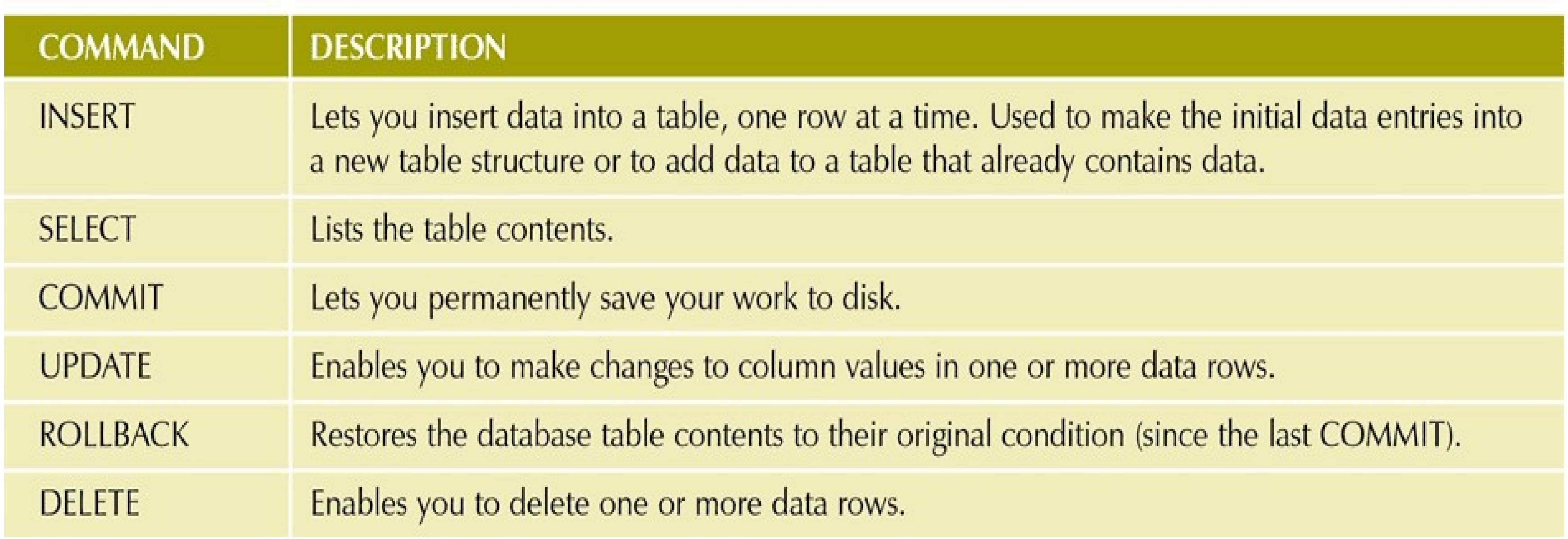
From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Data Manipulation Commands

- Adding table rows
- Listing table rows
- Updating table rows
- Saving table changes
- Restoring table contents
- Deleting table rows
- Inserting table rows with a select subquery

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Common SQL Data Manipulation Commands



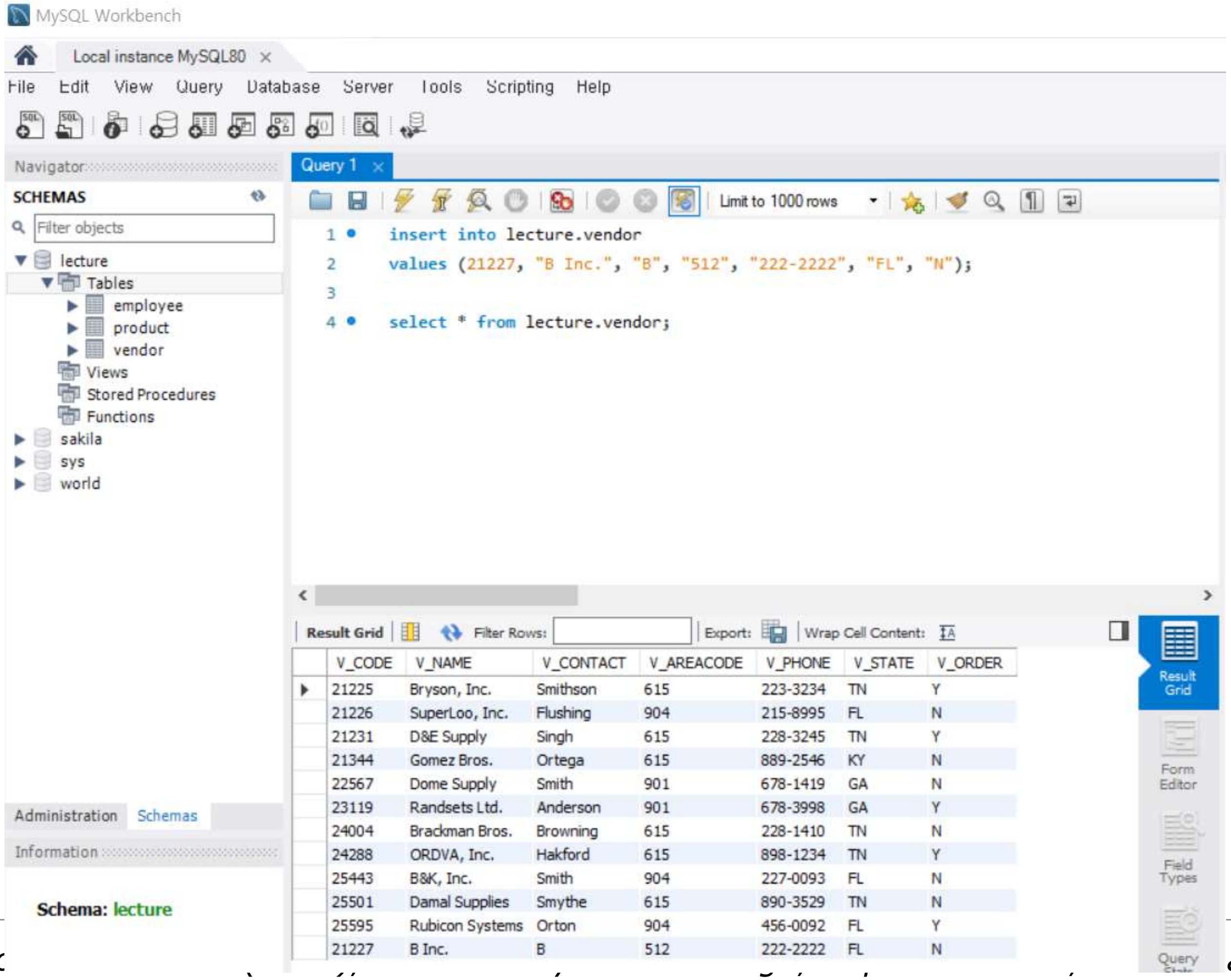
COMMAND	DESCRIPTION
INSERT	Lets you insert data into a table, one row at a time. Used to make the initial data entries into a new table structure or to add data to a table that already contains data.
SELECT	Lists the table contents.
COMMIT	Lets you permanently save your work to disk.
UPDATE	Enables you to make changes to column values in one or more data rows.
ROLLBACK	Restores the database table contents to their original condition (since the last COMMIT).
DELETE	Enables you to delete one or more data rows.

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Adding Table Rows

INSERT

- INSERT INTO tablename VALUES (value 1, value 2, …, value n);



Schema: lecture

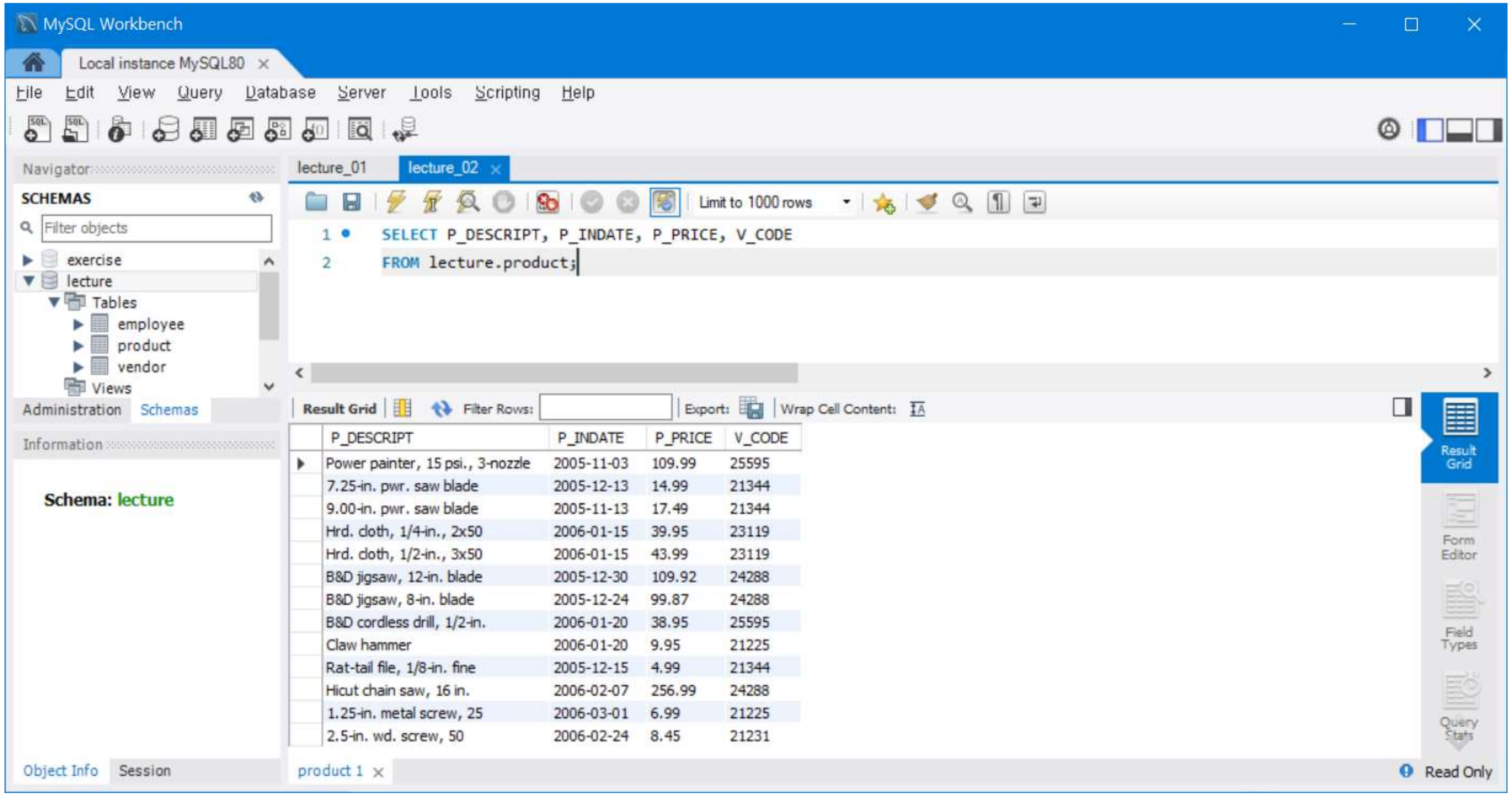
```
insert into lecture.vendor
values (21227, "B Inc.", "B", "512", "222-2222", "FL", "N");

select * from lecture.vendor;
```

V_CODE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE	V_STATE	V_ORDER
21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
21226	SuperLoo, Inc.	Flushing	904	215-8995	FL	N
21231	D&E Supply	Singh	615	228-3245	TN	Y
21344	Gomez Bros.	Ortega	615	889-2546	KY	N
22567	Dome Supply	Smith	901	678-1419	GA	N
23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
24004	Brackman Bros.	Browning	615	228-1410	TN	N
24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
25443	B&K, Inc.	Smith	904	227-0093	FL	N
25501	Damal Supplies	Smythe	615	890-3529	TN	N
25595	Rubicon Systems	Orton	904	456-0092	FL	Y
21227	B Inc.	B	512	222-2222	FL	N

Listing Table Rows (1)

SELECT

- SELECT columnlist FROM tablename;
- Columnlist represents one or more attributes, separated by commas



```
SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM lecture.product;
```

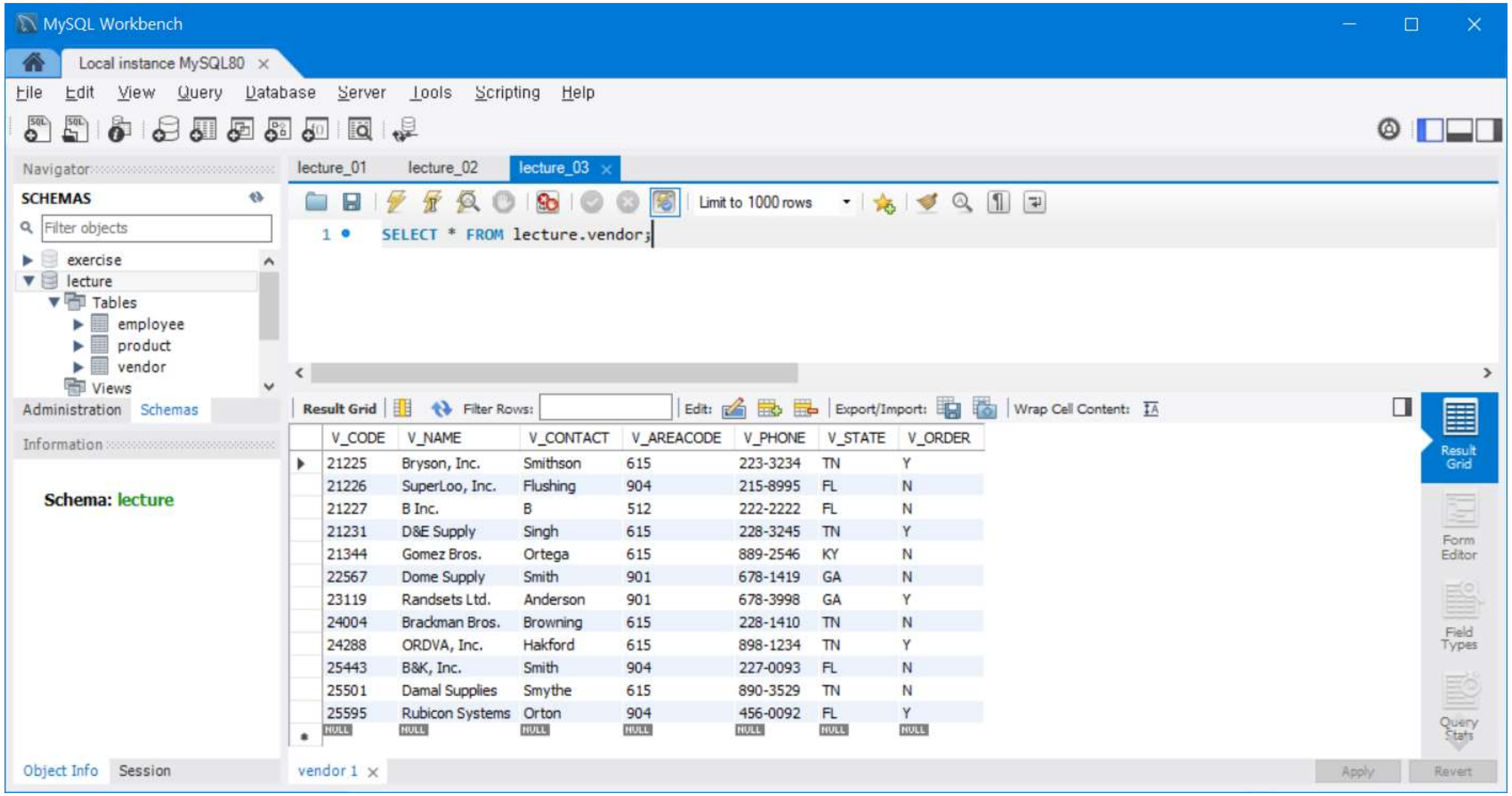
P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
Power painter, 15 psi., 3-nozzle	2005-11-03	109.99	25595
7.25-in. pwr. saw blade	2005-12-13	14.99	21344
9.00-in. pwr. saw blade	2005-11-13	17.49	21344
Hrd. cloth, 1/4-in., 2x50	2006-01-15	39.95	23119
Hrd. cloth, 1/2-in., 3x50	2006-01-15	43.99	23119
B&D jigsaw, 12-in. blade	2005-12-30	109.92	24288
B&D jigsaw, 8-in. blade	2005-12-24	99.87	24288
B&D cordless drill, 1/2-in.	2006-01-20	38.95	25595
Claw hammer	2006-01-20	9.95	21225
Rat-tail file, 1/8-in. fine	2005-12-15	4.99	21344
Hicut chain saw, 16 in.	2006-02-07	256.99	24288
1.25-in. metal screw, 25	2006-03-01	6.99	21225
2.5-in. wd. screw, 50	2006-02-24	8.45	21231

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Listing Table Rows (2)

SELECT

- SELECT columnlist FROM tablename;
- Asterisk can be used as wildcard character to list all attributes



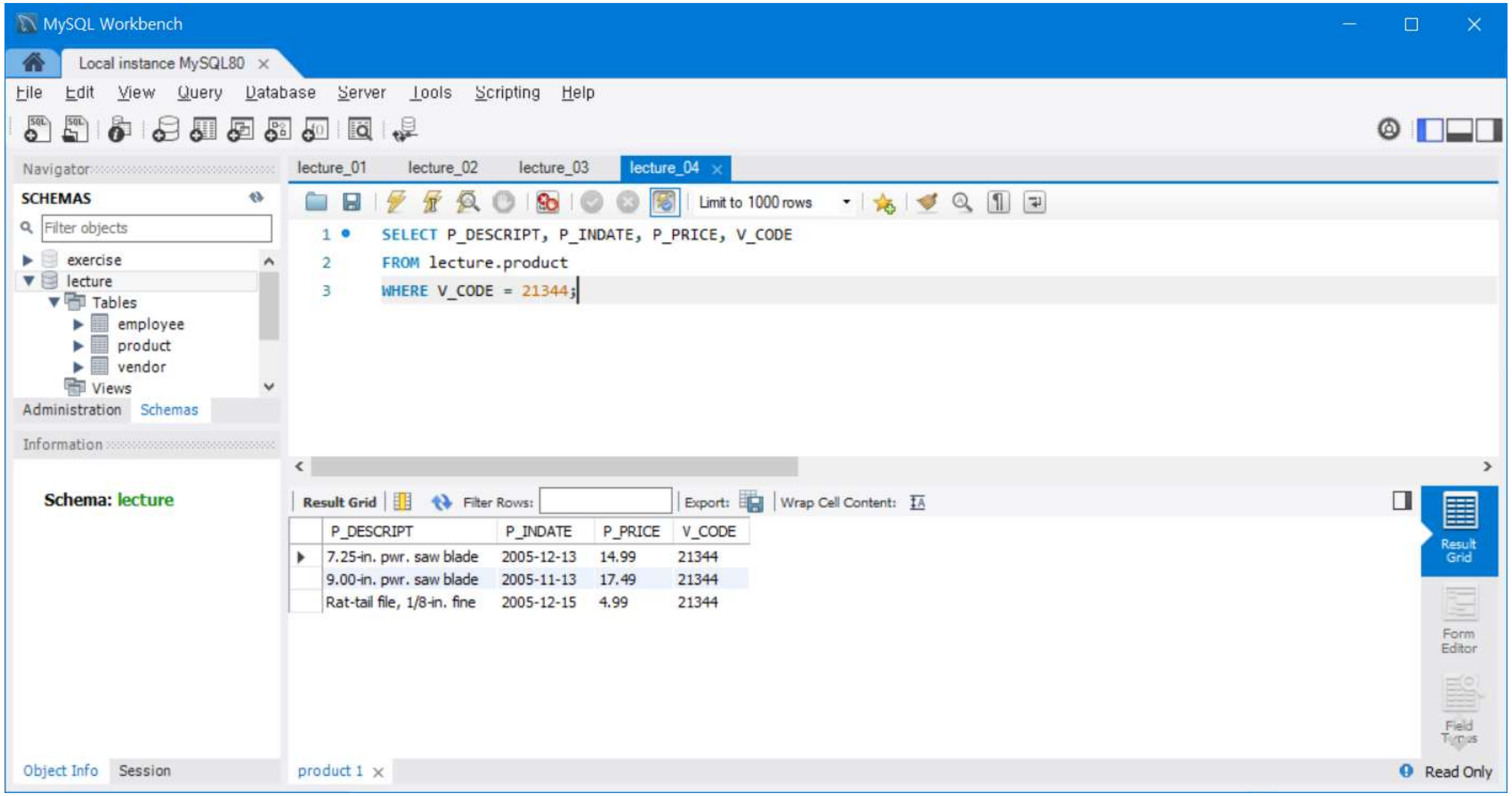
```
SELECT * FROM lecture.vendor;
```

V_CODE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE	V_STATE	V_ORDER
21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
21226	SuperLoo, Inc.	Flushing	904	215-8995	FL	N
21227	B Inc.	B	512	222-2222	FL	N
21231	D&E Supply	Singh	615	228-3245	TN	Y
21344	Gomez Bros.	Ortega	615	889-2546	KY	N
22567	Dome Supply	Smith	901	678-1419	GA	N
23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
24004	Brackman Bros.	Browning	615	228-1410	TN	N
24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
25443	B&K, Inc.	Smith	904	227-0093	FL	N
25501	Damal Supplies	Smythe	615	890-3529	TN	N
25595	Rubicon Systems	Orton	904	456-0092	FL	Y
NULL	NULL	NULL	NULL	NULL	NULL	NULL

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Selecting Rows with Conditional Restrictions (1)

- **Select partial table contents by placing restrictions on rows to be included in output**
  - Add conditional restrictions to the SELECT statement, using WHERE clause
  - SELECT *columnlist* FROM *tablename* [WHERE *conditionlist*] ;



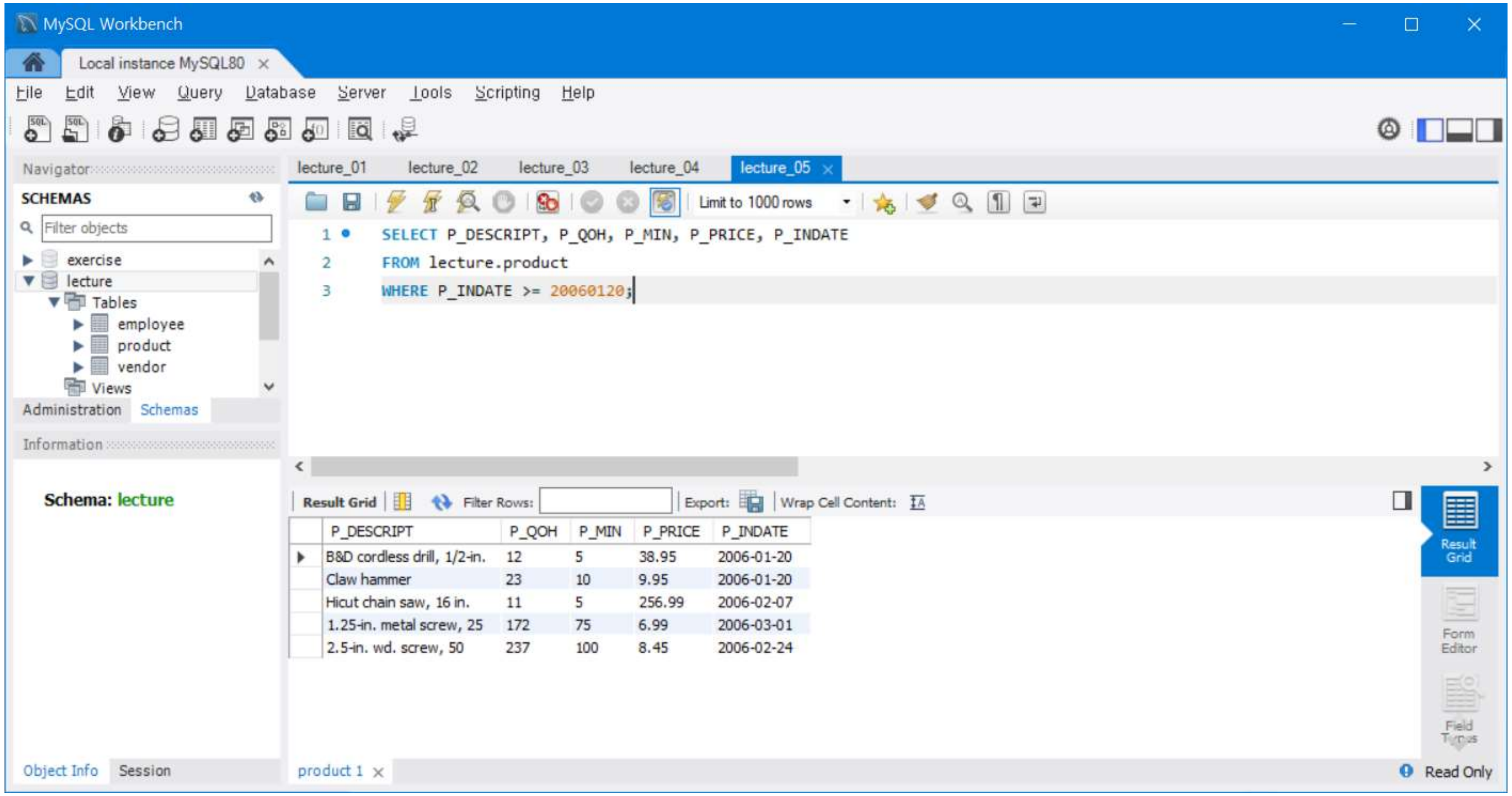
```
SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM lecture.product
WHERE V_CODE = 21344;
```

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
7.25-in. pwr. saw blade	2005-12-13	14.99	21344
9.00-in. pwr. saw blade	2005-11-13	17.49	21344
Rat-tail file, 1/8-in. fine	2005-12-15	4.99	21344

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Selecting Rows with Conditional Restrictions (2)

- **Select partial table contents by placing restrictions on rows to be included in output**
  - Add conditional restrictions to the SELECT statement, using WHERE clause
  - SELECT *columnlist* FROM *tablename* [WHERE *conditionlist*] ;



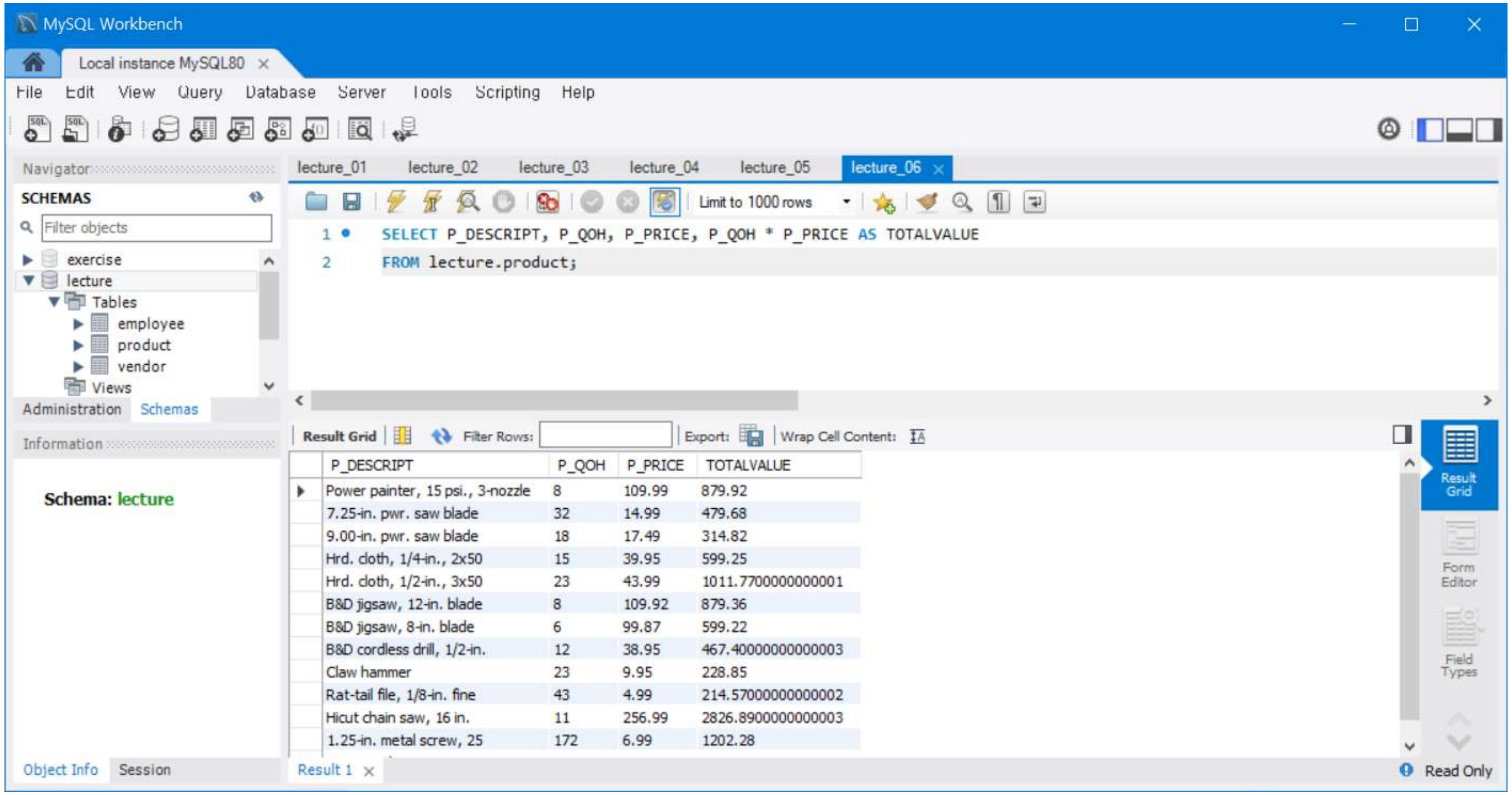
```sql
SELECT P_DESCRIPT, P_QOH, P_MIN, P_PRICE, P_INDATE
FROM lecture.product
WHERE P_INDATE >= 20060120;
```

P_DESCRIPT	P_QOH	P_MIN	P_PRICE	P_INDATE
B&D cordless drill, 1/2-in.	12	5	38.95	2006-01-20
Claw hammer	23	10	9.95	2006-01-20
Hicut chain saw, 16 in.	11	5	256.99	2006-02-07
1.25-in. metal screw, 25	172	75	6.99	2006-03-01
2.5-in. wd. screw, 50	237	100	8.45	2006-02-24

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Selecting Rows with Conditional Restrictions (3)

- **Select partial table contents by placing restrictions on rows to be included in output**
  - Add conditional restrictions to the SELECT statement, using column calculation
  - SELECT *columnlist* FROM *tablename* [WHERE *conditionlist*] ;



```sql
SELECT P_DESCRIPT, P_QOH, P_PRICE, P_QOH * P_PRICE AS TOTALVALUE
FROM lecture.product;
```

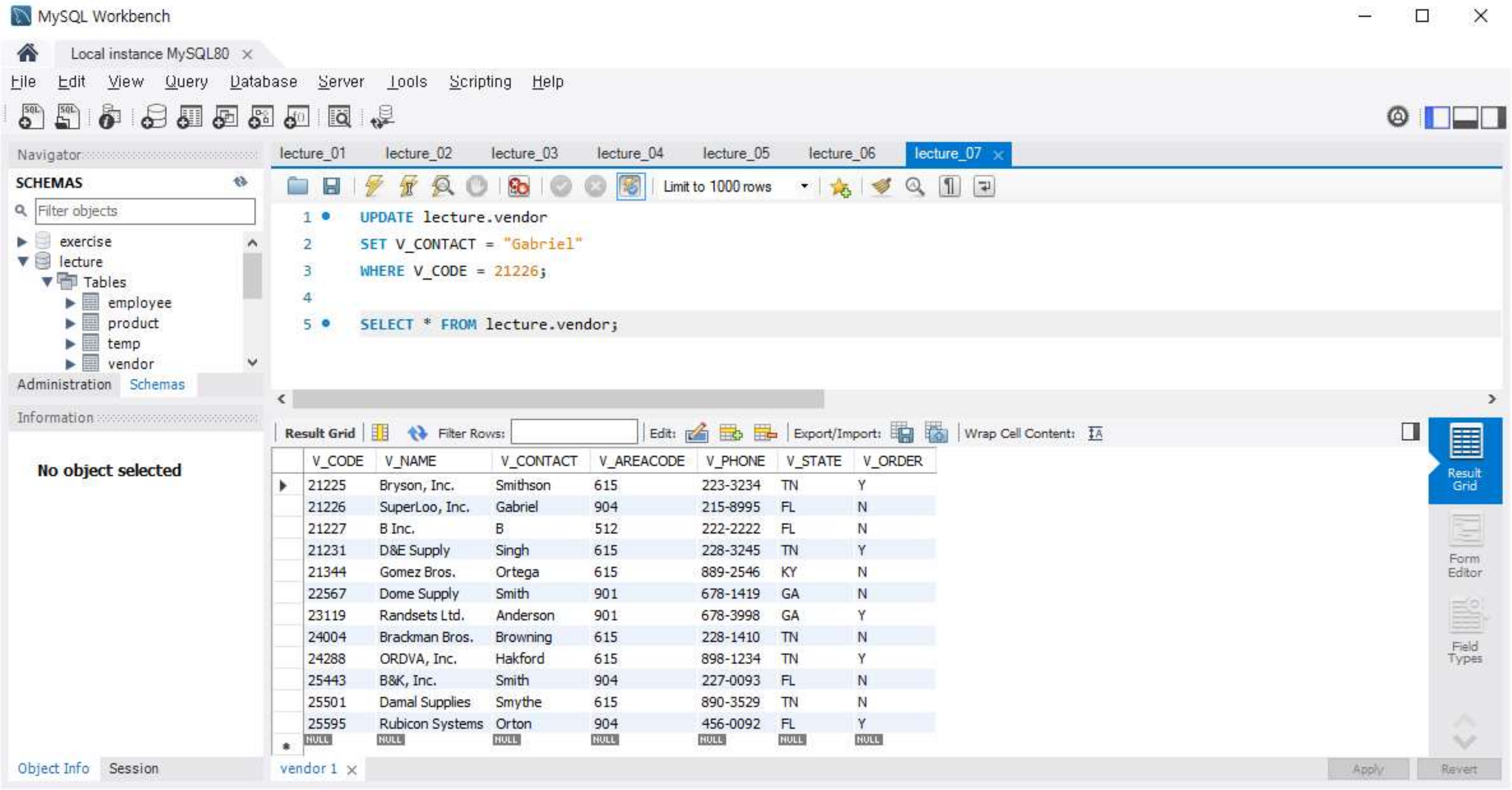
P_DESCRIPT	P_QOH	P_PRICE	TOTALVALUE
Power painter, 15 psi., 3-nozzle	8	109.99	879.92
7.25-in. pwr. saw blade	32	14.99	479.68
9.00-in. pwr. saw blade	18	17.49	314.82
Hrd. cloth, 1/4-in., 2x50	15	39.95	599.25
Hrd. cloth, 1/2-in., 3x50	23	43.99	1011.7700000000001
B&D jigsaw, 12-in. blade	8	109.92	879.36
B&D jigsaw, 8-in. blade	6	99.87	599.22
B&D cordless drill, 1/2-in.	12	38.95	467.40000000000003
Claw hammer	23	9.95	228.85
Rat-tail file, 1/8-in. fine	43	4.99	214.57000000000002
Hicut chain saw, 16 in.	11	256.99	2826.8900000000003
1.25-in. metal screw, 25	172	6.99	1202.28

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Updating Table Rows

UPDATE

- UPDATE tablename SET columnname = expression [, columnname = expression] [WHERE conditionlist];
- If more than one attribute is to be updated in the row, separate corrections with commas



```sql
UPDATE lecture.vendor
SET V_CONTACT = "Gabriel"
WHERE V_CODE = 21226;

SELECT * FROM lecture.vendor;
```

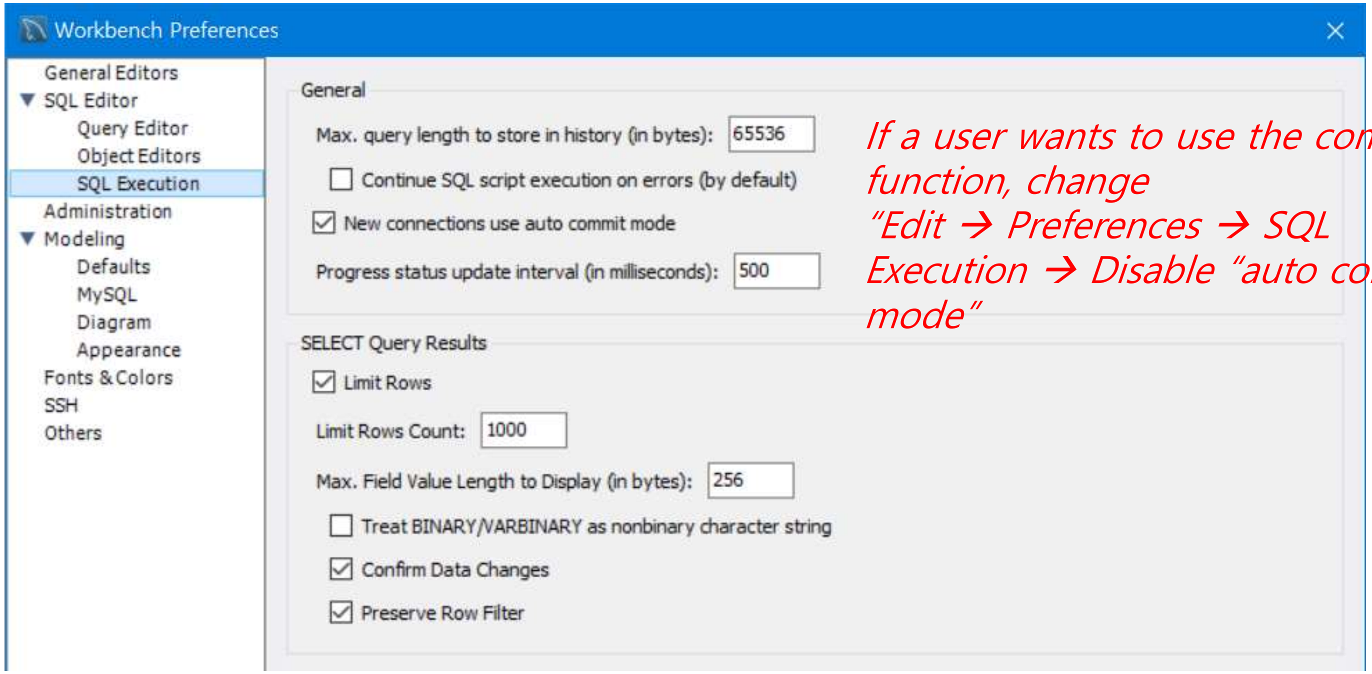
V_CODE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE	V_STATE	V_ORDER
21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
21226	SuperLoo, Inc.	Gabriel	904	215-8995	FL	N
21227	B Inc.	B	512	222-2222	FL	N
21231	D&E Supply	Singh	615	228-3245	TN	Y
21344	Gomez Bros.	Ortega	615	889-2546	KY	N
22567	Dome Supply	Smith	901	678-1419	GA	N
23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
24004	Brackman Bros.	Browning	615	228-1410	TN	N
24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
25443	B&K, Inc.	Smith	904	227-0093	FL	N
25501	Damal Supplies	Smythe	615	890-3529	TN	N
25595	Rubicon Systems	Orton	904	456-0092	FL	Y
NULL	NULL	NULL	NULL	NULL	NULL	NULL

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Saving Table Changes

COMMIT

- Changes made to table to table contents are not physically saved on disk until
  - Database/Program is closed
  - COMMIT command is used
- MySQL workbench commits automatically by default.



Workbench Preferences

General

Max. query length to store in history (in bytes): 65536

□ Continue SQL script execution on errors (by default)

☑ New connections use auto commit mode

Progress status update interval (in milliseconds): 500

SELECT Query Results

☑ Limit Rows

Limit Rows Count: 1000

Max. Field Value Length to Display (in bytes): 256

□ Treat BINARY/VARBINARY as nonbinary character string

☑ Confirm Data Changes

☑ Preserve Row Filter

If a user wants to use the commit function, change "Edit → Preferences → SQL Execution → Disable "auto commit mode"

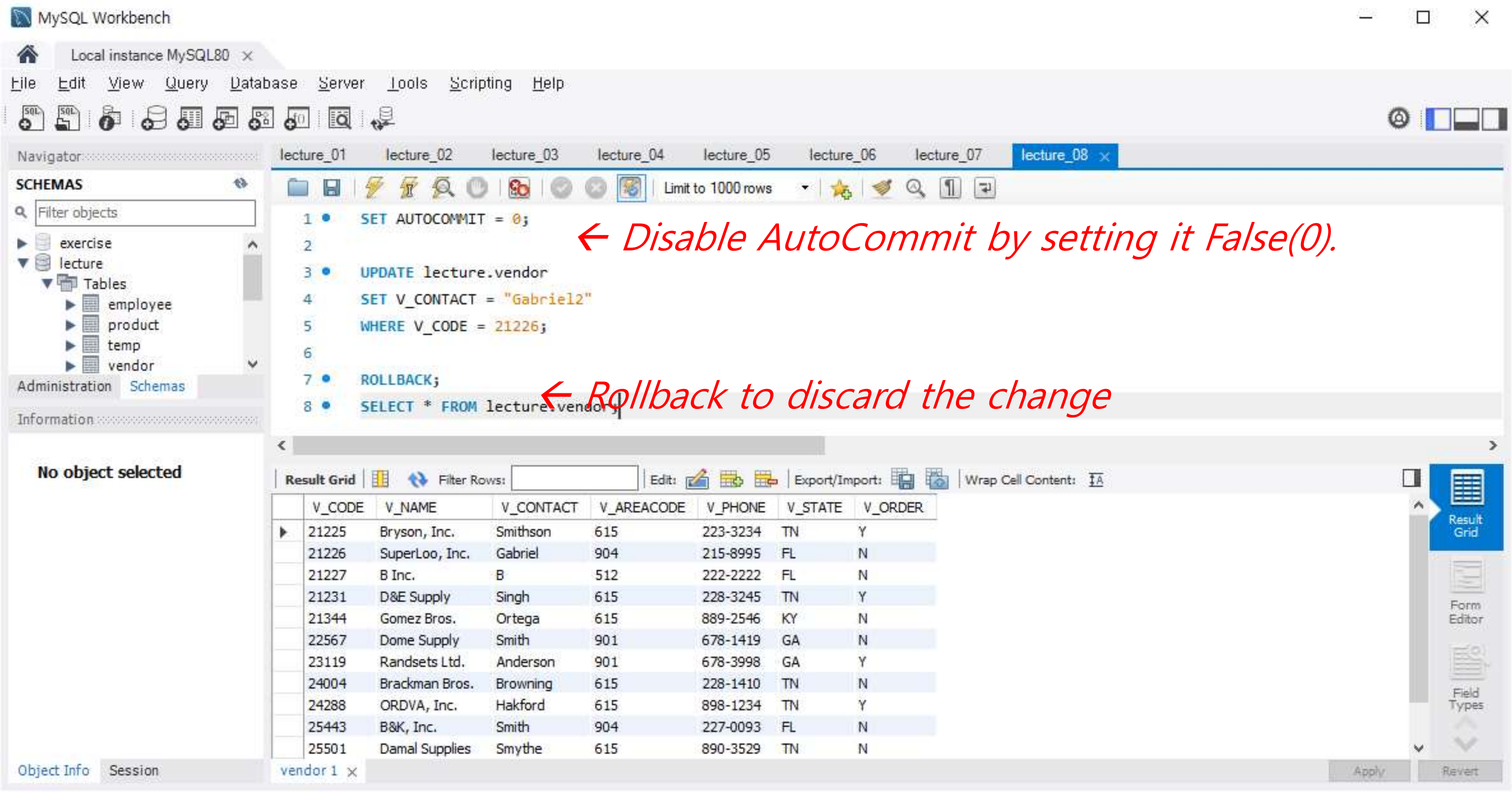
From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Restoring Table Contents

COMMIT and ROLLBACK only work with data manipulation commands that are used to add, modify, or delete table rows

ROLLBACK

- Used restore the database to its previous condition
- Only applicable if COMMIT command has not been used to permanently store the changes in the database



```sql
SET AUTOCOMMIT = 0;

UPDATE lecture.vendor
SET V_CONTACT = "Gabriel2"
WHERE V_CODE = 21226;

ROLLBACK;
SELECT * FROM lecture.vendor;
```

← Disable AutoCommit by setting it False(0).

← Rollback to discard the change

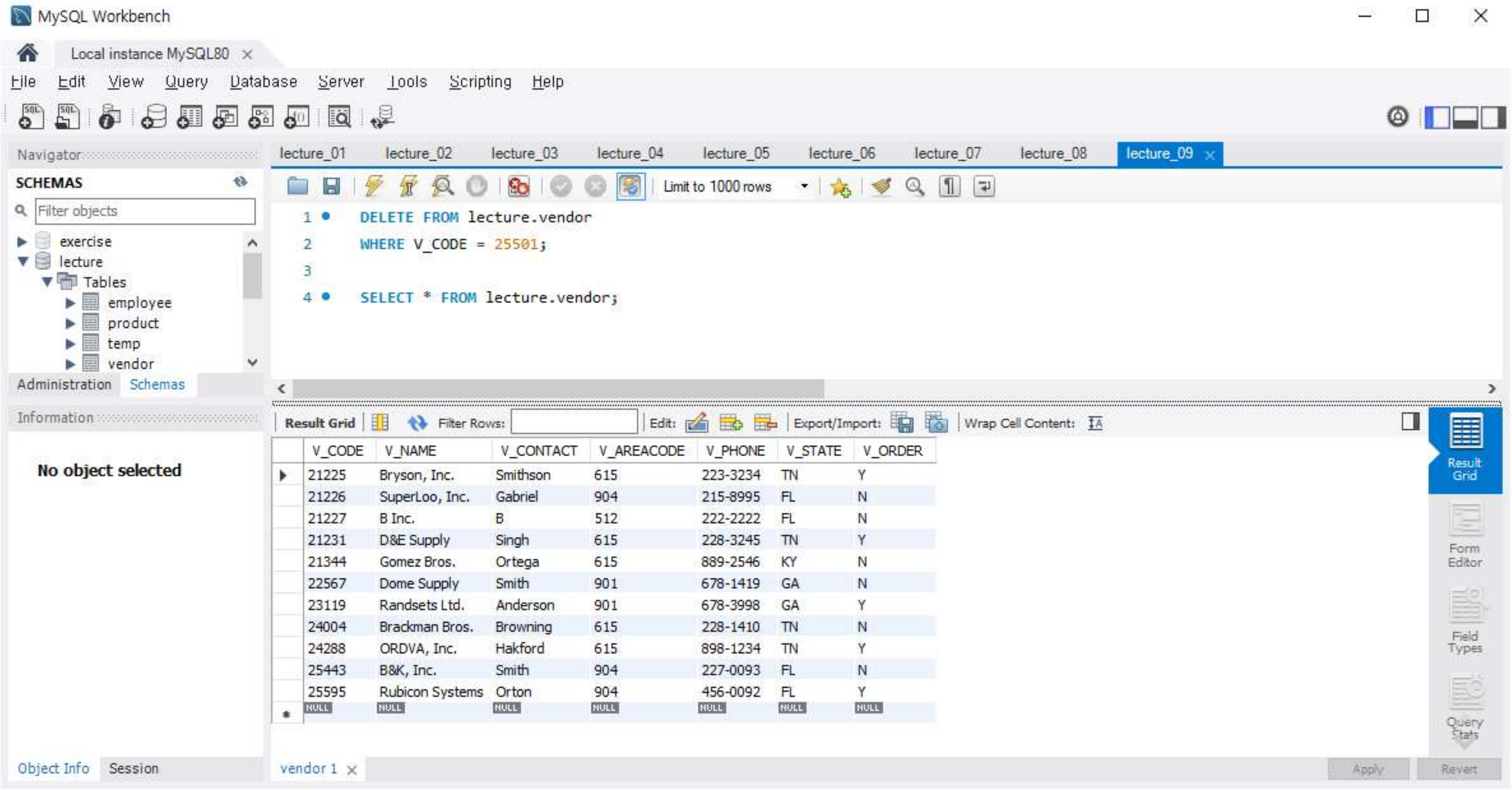
V_CODE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE	V_STATE	V_ORDER
21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
21226	SuperLoo, Inc.	Gabriel	904	215-8995	FL	N
21227	B Inc.	B	512	222-2222	FL	N
21231	D&E Supply	Singh	615	228-3245	TN	Y
21344	Gomez Bros.	Ortega	615	889-2546	KY	N
22567	Dome Supply	Smith	901	678-1419	GA	N
23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
24004	Brackman Bros.	Browning	615	228-1410	TN	N
24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
25443	B&K, Inc.	Smith	904	227-0093	FL	N
25501	Damal Supplies	Smythe	615	890-3529	TN	N

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Deleting Table Rows

DELETE

- DELETE FROM tablename [WHERE conditionlist];
- WHERE condition is optional (Unless being specified WHERE, all rows from the specified table will be deleted)



```
DELETE FROM lecture.vendor
WHERE V_CODE = 25501;

SELECT * FROM lecture.vendor;
```

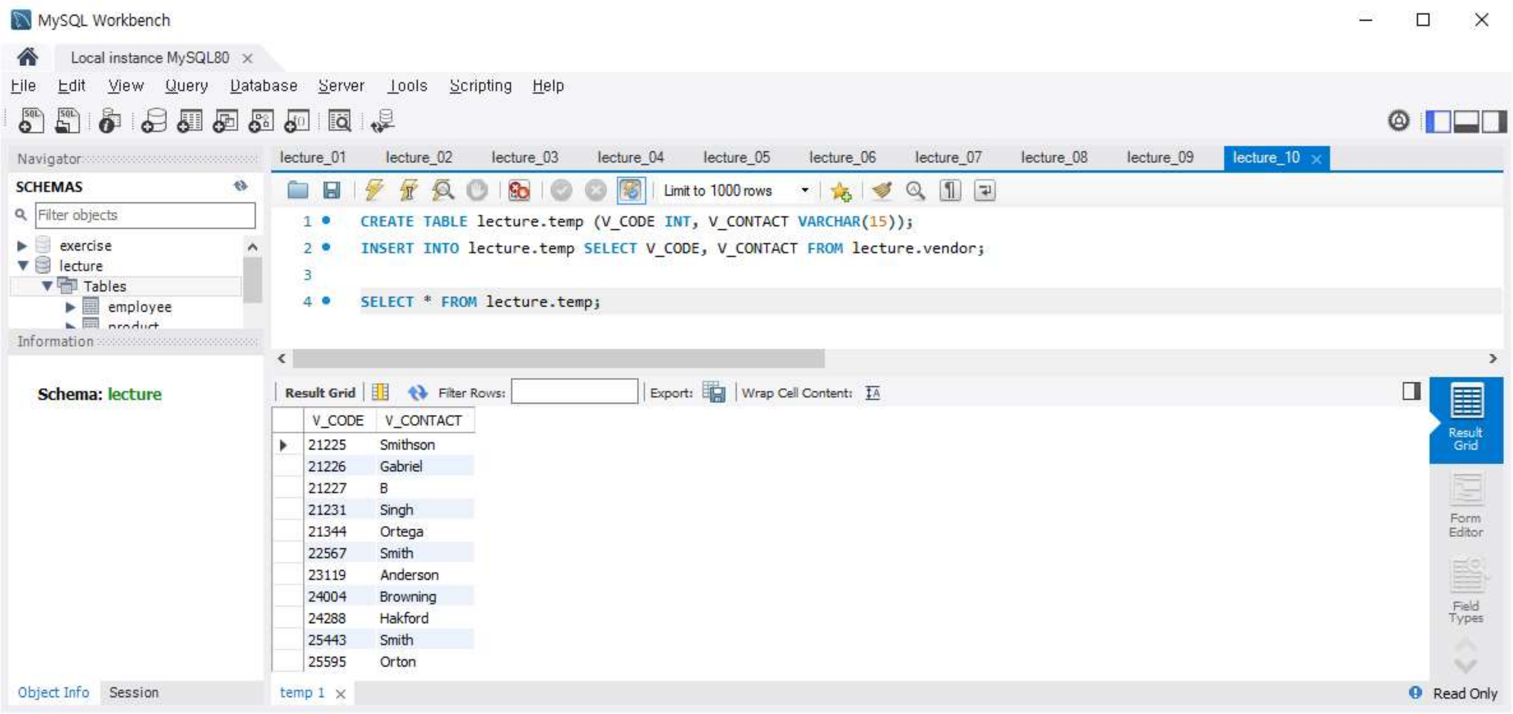
V_CODE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE	V_STATE	V_ORDER
21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
21226	SuperLoo, Inc.	Gabriel	904	215-8995	FL	N
21227	B Inc.	B	512	222-2222	FL	N
21231	D&E Supply	Singh	615	228-3245	TN	Y
21344	Gomez Bros.	Ortega	615	889-2546	KY	N
22567	Dome Supply	Smith	901	678-1419	GA	N
23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
24004	Brackman Bros.	Browning	615	228-1410	TN	N
24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
25443	B&K, Inc.	Smith	904	227-0093	FL	N
25595	Rubicon Systems	Orton	904	456-0092	FL	Y
NULL	NULL	NULL	NULL	NULL	NULL	NULL

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Inserting Table Rows with a Select Subquery

INSERT

- Inserts multiple rows from another table (source)
- Uses SELECT subquery: Query that is embedded (or nested) inside another query executed first
- INSERT INTO tablename SELECT columnlist FROM tablename;



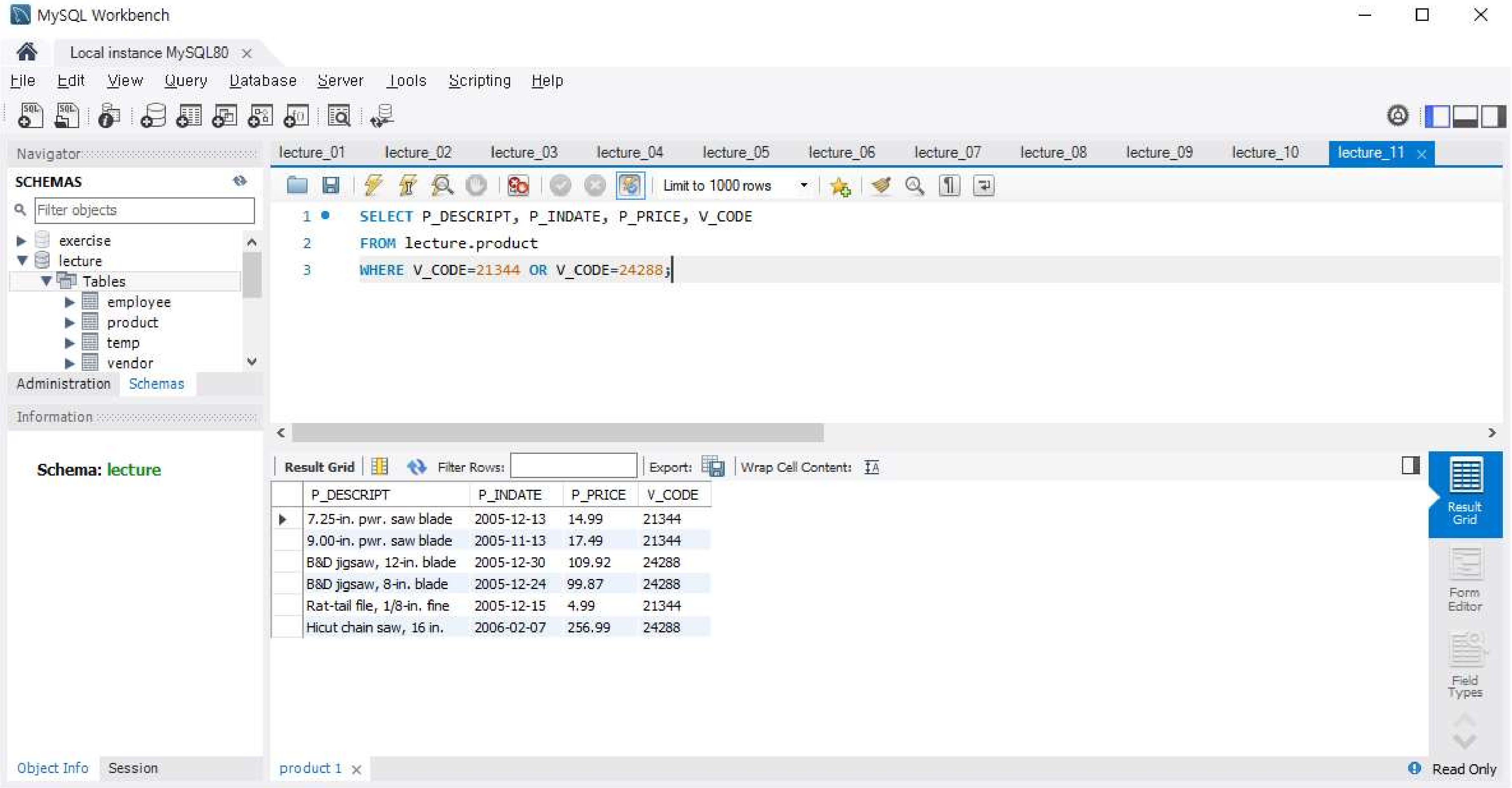
```
CREATE TABLE lecture.temp (V_CODE INT, V_CONTACT VARCHAR(15));
INSERT INTO lecture.temp SELECT V_CODE, V_CONTACT FROM lecture.vendor;

SELECT * FROM lecture.temp;
```

V_CODE	V_CONTACT
21225	Smithson
21226	Gabriel
21227	B
21231	Singh
21344	Ortega
22567	Smith
23119	Anderson
24004	Browning
24288	Hakford
25443	Smith
25595	Orton

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Logical Operators (1)

LOGICAL “OR”



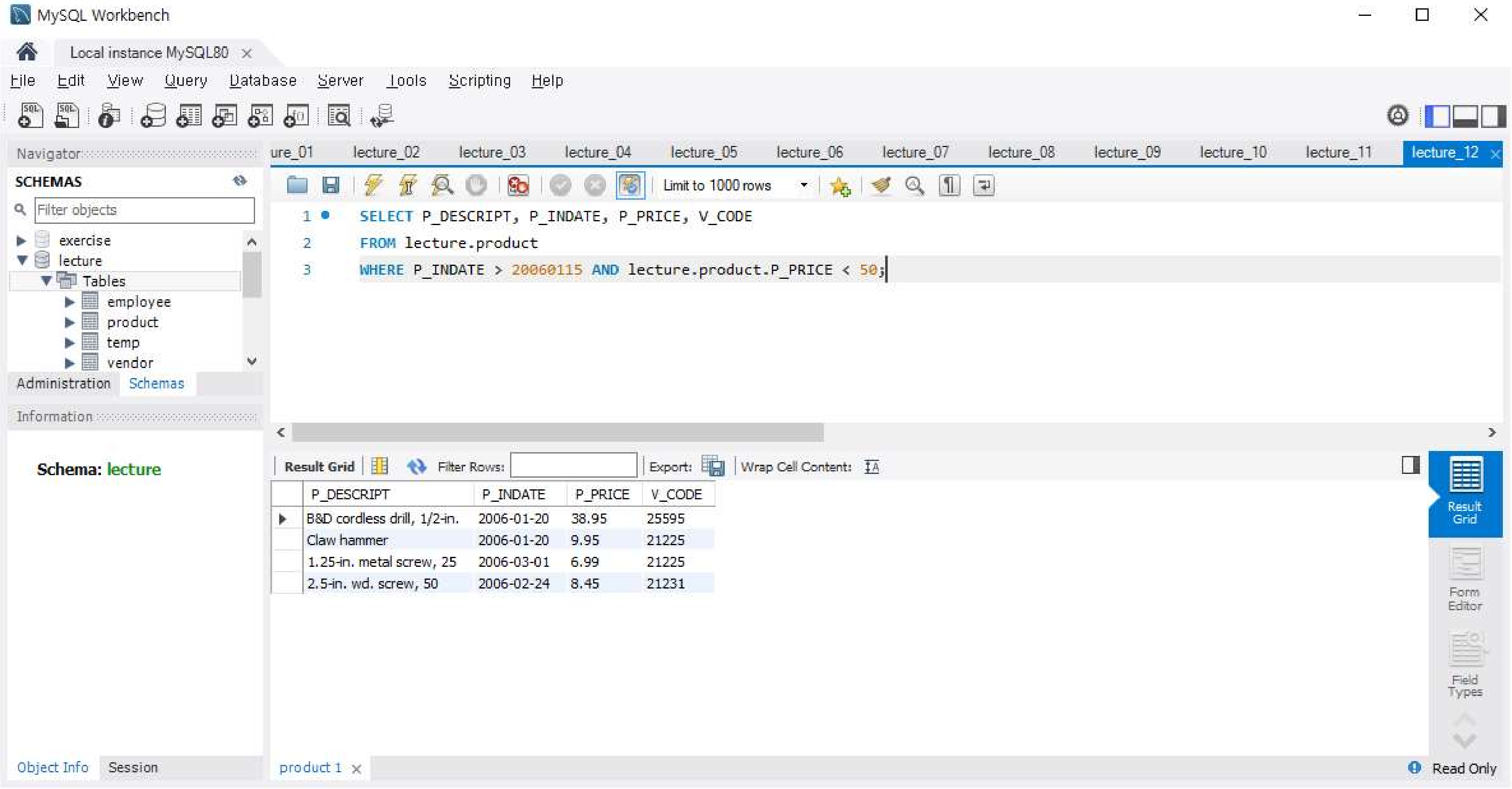
```sql
SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM lecture.product
WHERE V_CODE=21344 OR V_CODE=24288;
```

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
7.25-in. pwr. saw blade	2005-12-13	14.99	21344
9.00-in. pwr. saw blade	2005-11-13	17.49	21344
B&D jigsaw, 12-in. blade	2005-12-30	109.92	24288
B&D jigsaw, 8-in. blade	2005-12-24	99.87	24288
Rat-tail file, 1/8-in. fine	2005-12-15	4.99	21344
Hicut chain saw, 16 in.	2006-02-07	256.99	24288

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Logical Operators (2)

LOGICAL “AND”



MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
exercise
lecture
Tables
employee
product
temp
vendor
Administration Schemas
Information
ure_01 lecture_02 lecture_03 lecture_04 lecture_05 lecture_06 lecture_07 lecture_08 lecture_09 lecture_10 lecture_11 lecture_12
Limit to 1000 rows
Result Grid Filter Rows: Export: Wrap Cell Content:
Result Grid
Form Editor
Field Types
Object Info Session
product 1
Read Only

```sql
SELECT P_DESCRIPT, P_INDATE, P_PRICE, V_CODE
FROM lecture.product
WHERE P_INDATE > 20060115 AND lecture.product.P_PRICE < 50;
```

Schema: lecture

P_DESCRIPT	P_INDATE	P_PRICE	V_CODE
B&D cordless drill, 1/2-in.	2006-01-20	38.95	25595
Claw hammer	2006-01-20	9.95	21225
1.25-in. metal screw, 25	2006-03-01	6.99	21225
2.5-in. wd. screw, 50	2006-02-24	8.45	21231

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

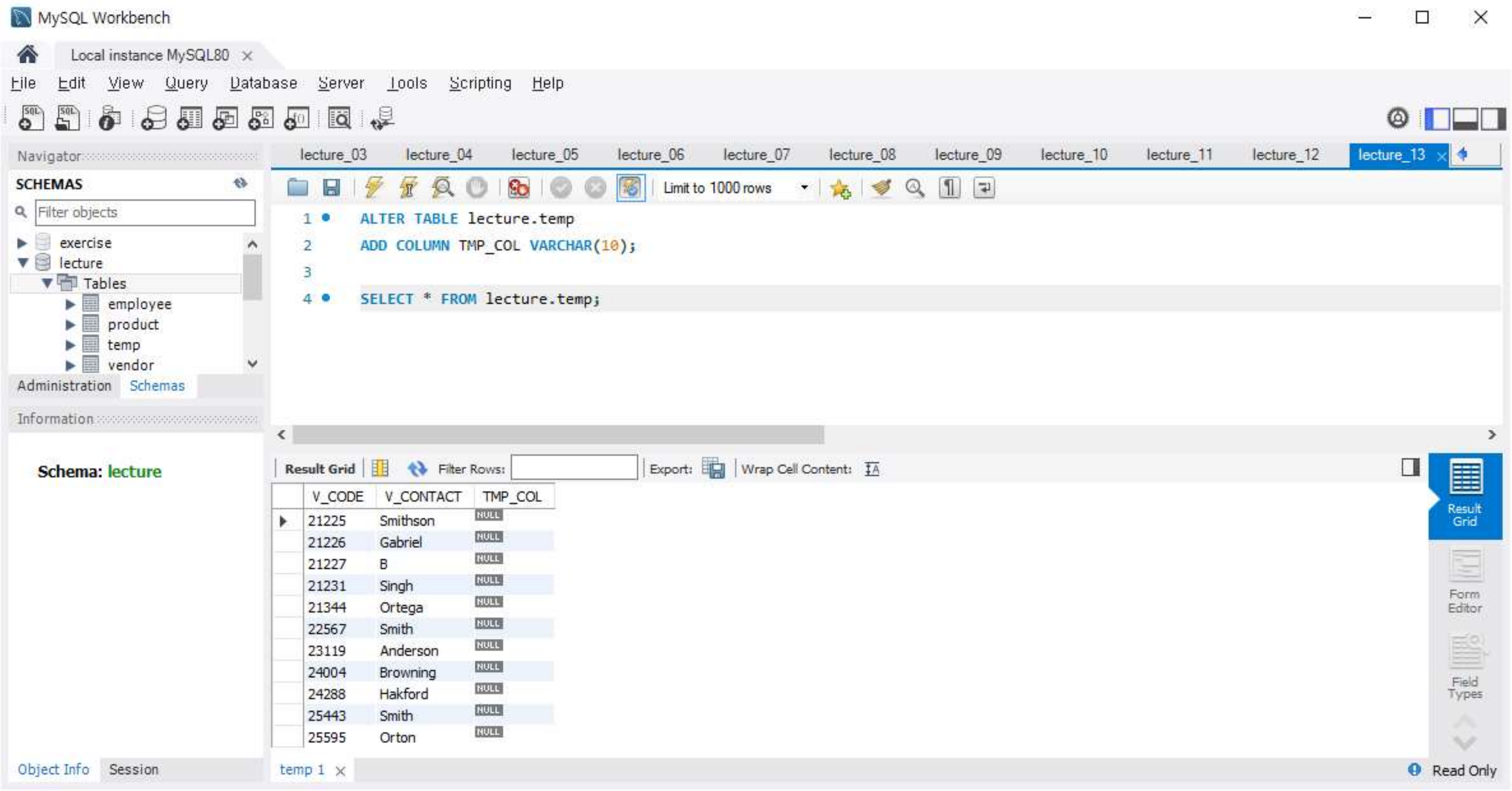
Special Operators

- **BETWEEN**
  - Used to check whether attribute value is within a range
- **IS NULL**
  - Used to check whether attribute value is null
- **LIKE**
  - Used to check whether attribute value matches a given string pattern
- **IN**
  - Used to check whether attribute value matches any value within a value list
- **EXISTS**
  - Used to check if a subquery returns any rows

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Advanced Data Definition Commands

- **All changes in the table structure are made by using the ALTER command**
  - Followed by a keyword that produces specific change
  - Three options are available: ADD, MODIFY, DROP



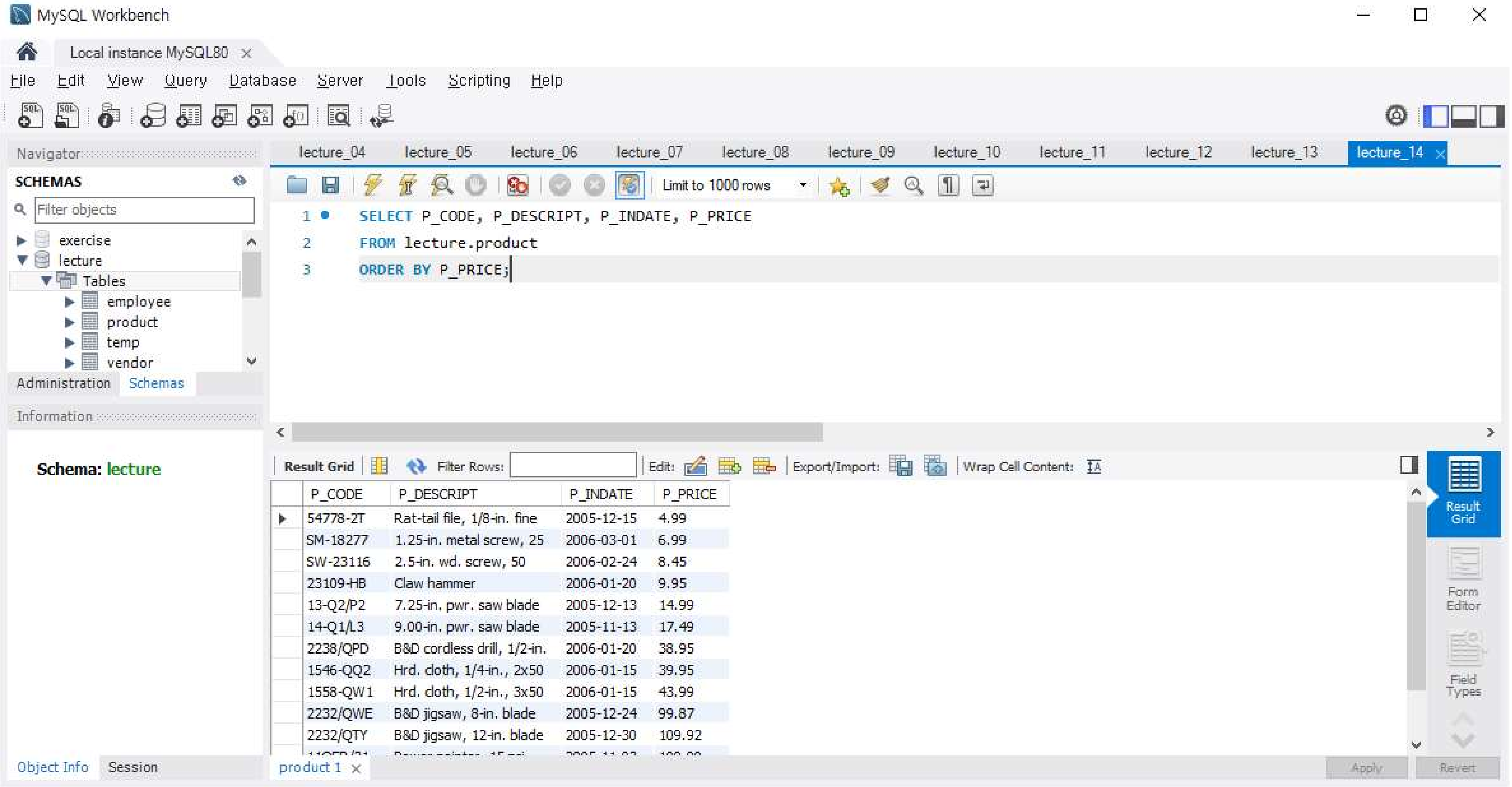
```
ALTER TABLE lecture.temp
ADD COLUMN TMP_COL VARCHAR(10);

SELECT * FROM lecture.temp;
```

V_CODE	V_CONTACT	TMP_COL
21225	Smithson	NULL
21226	Gabriel	NULL
21227	B	NULL
21231	Singh	NULL
21344	Ortega	NULL
22567	Smith	NULL
23119	Anderson	NULL
24004	Browning	NULL
24288	Hakford	NULL
25443	Smith	NULL
25595	Orton	NULL

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Ordering a Listing (1)

ORDER BY

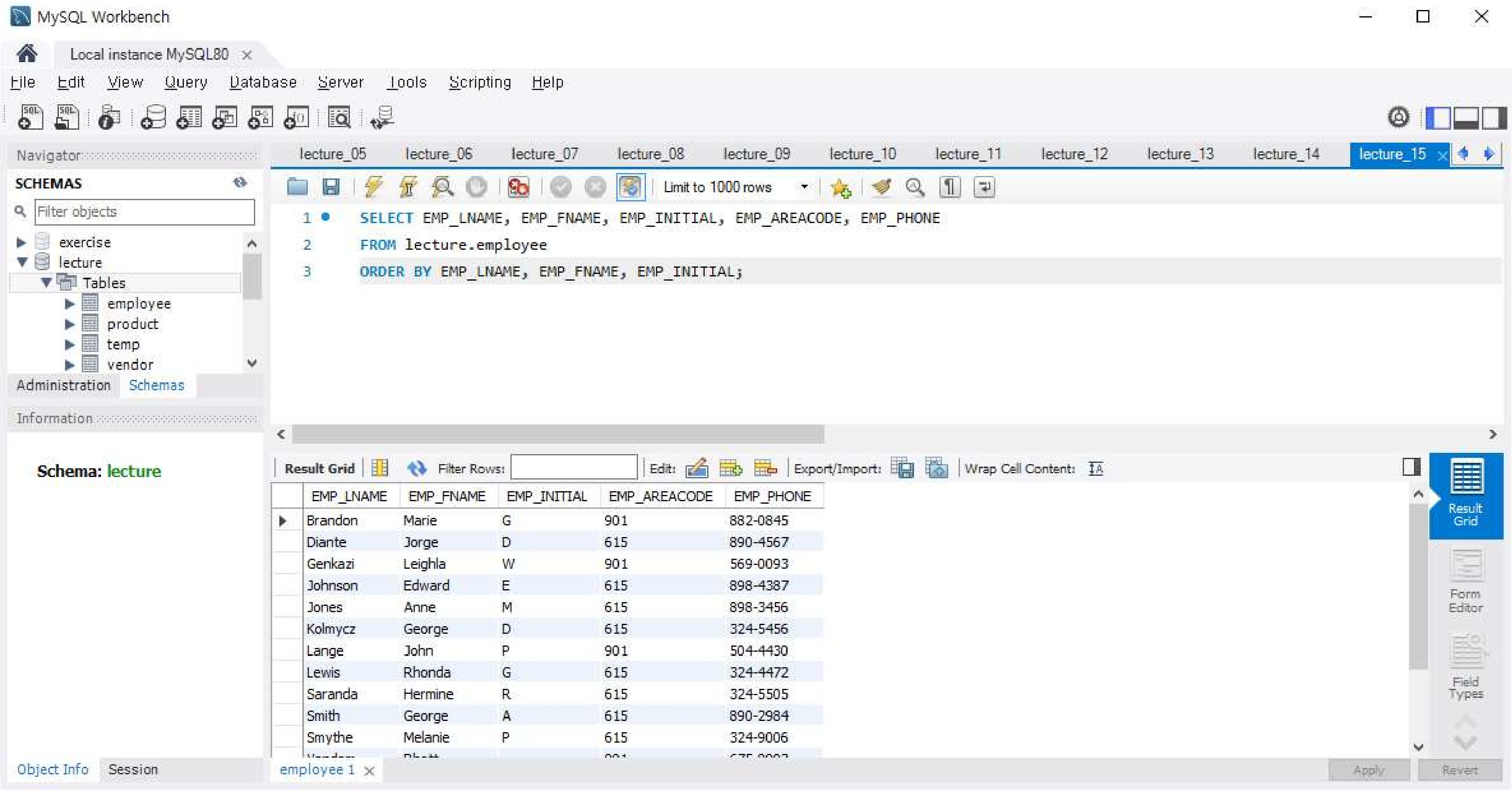


```sql
SELECT P_CODE, P_DESCRIPT, P_INDATE, P_PRICE
FROM lecture.product
ORDER BY P_PRICE;
```

P_CODE	P_DESCRIPT	P_INDATE	P_PRICE
54778-2T	Rat-tail file, 1/8-in. fine	2005-12-15	4.99
SM-18277	1.25-in. metal screw, 25	2006-03-01	6.99
SW-23116	2.5-in. wd. screw, 50	2006-02-24	8.45
23109-HB	Claw hammer	2006-01-20	9.95
13-Q2/P2	7.25-in. pwr. saw blade	2005-12-13	14.99
14-Q1/L3	9.00-in. pwr. saw blade	2005-11-13	17.49
2238/QPD	B&D cordless drill, 1/2-in.	2006-01-20	38.95
1546-QQ2	Hrd. cloth, 1/4-in., 2x50	2006-01-15	39.95
1558-QW1	Hrd. cloth, 1/2-in., 3x50	2006-01-15	43.99
2232/QWE	B&D jigsaw, 8-in. blade	2005-12-24	99.87
2232/QTY	B&D jigsaw, 12-in. blade	2005-12-30	109.92
11QER/31	Power painter, 15 psi	2005-11-03	109.99

Ordering a Listing (2)

ORDER BY

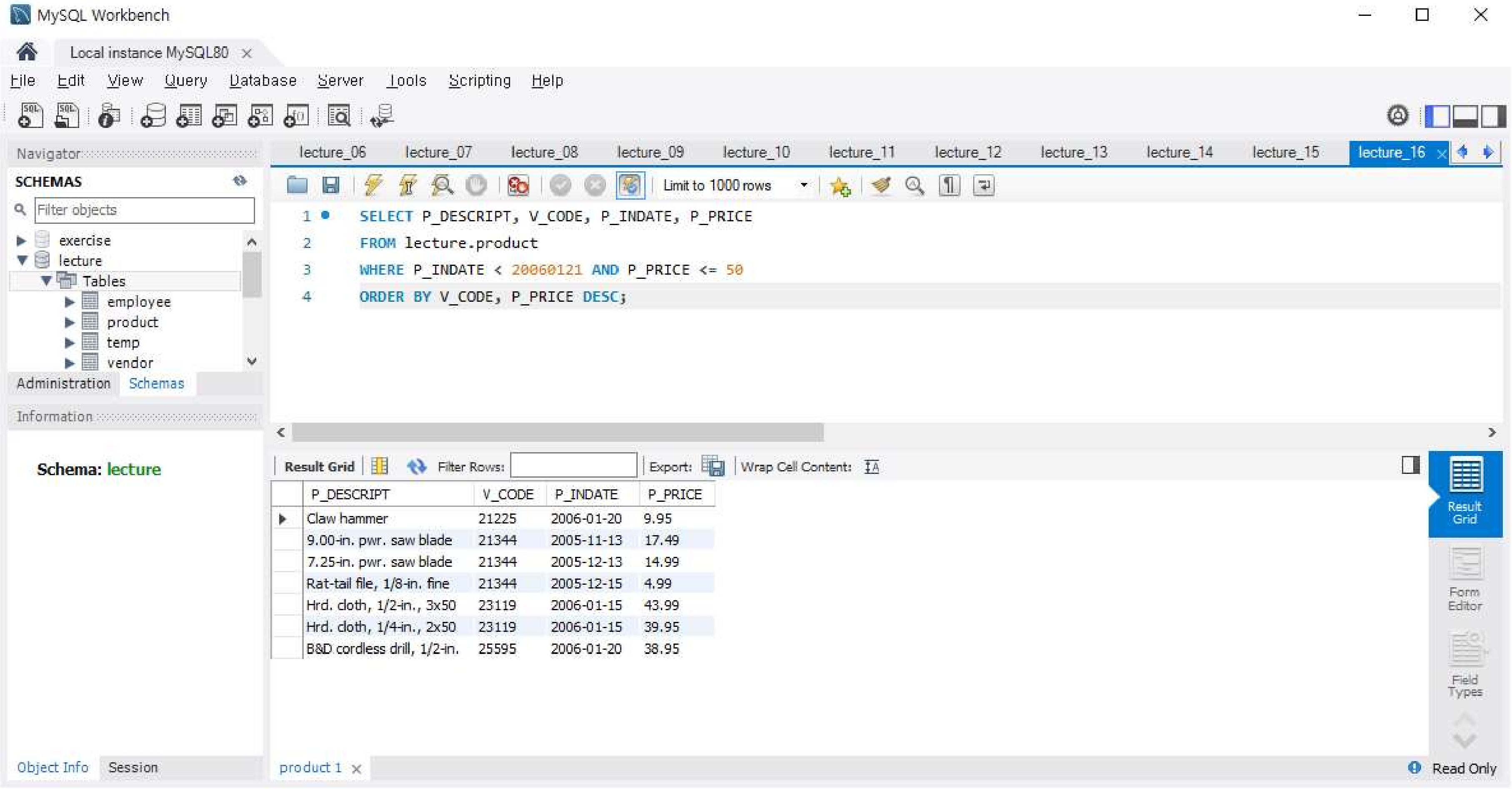


```sql
SELECT EMP_LNAME, EMP_FNAME, EMP_INITIAL, EMP_AREACODE, EMP_PHONE
FROM lecture.employee
ORDER BY EMP_LNAME, EMP_FNAME, EMP_INITIAL;
```

EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_AREACODE	EMP_PHONE
Brandon	Marie	G	901	882-0845
Diante	Jorge	D	615	890-4567
Genkazi	Leighla	W	901	569-0093
Johnson	Edward	E	615	898-4387
Jones	Anne	M	615	898-3456
Kolmycz	George	D	615	324-5456
Lange	John	P	901	504-4430
Lewis	Rhonda	G	615	324-4472
Saranda	Hermine	R	615	324-5505
Smith	George	A	615	890-2984
Smythe	Melanie	P	615	324-9006

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

A Query Based on Multiple Restrictions

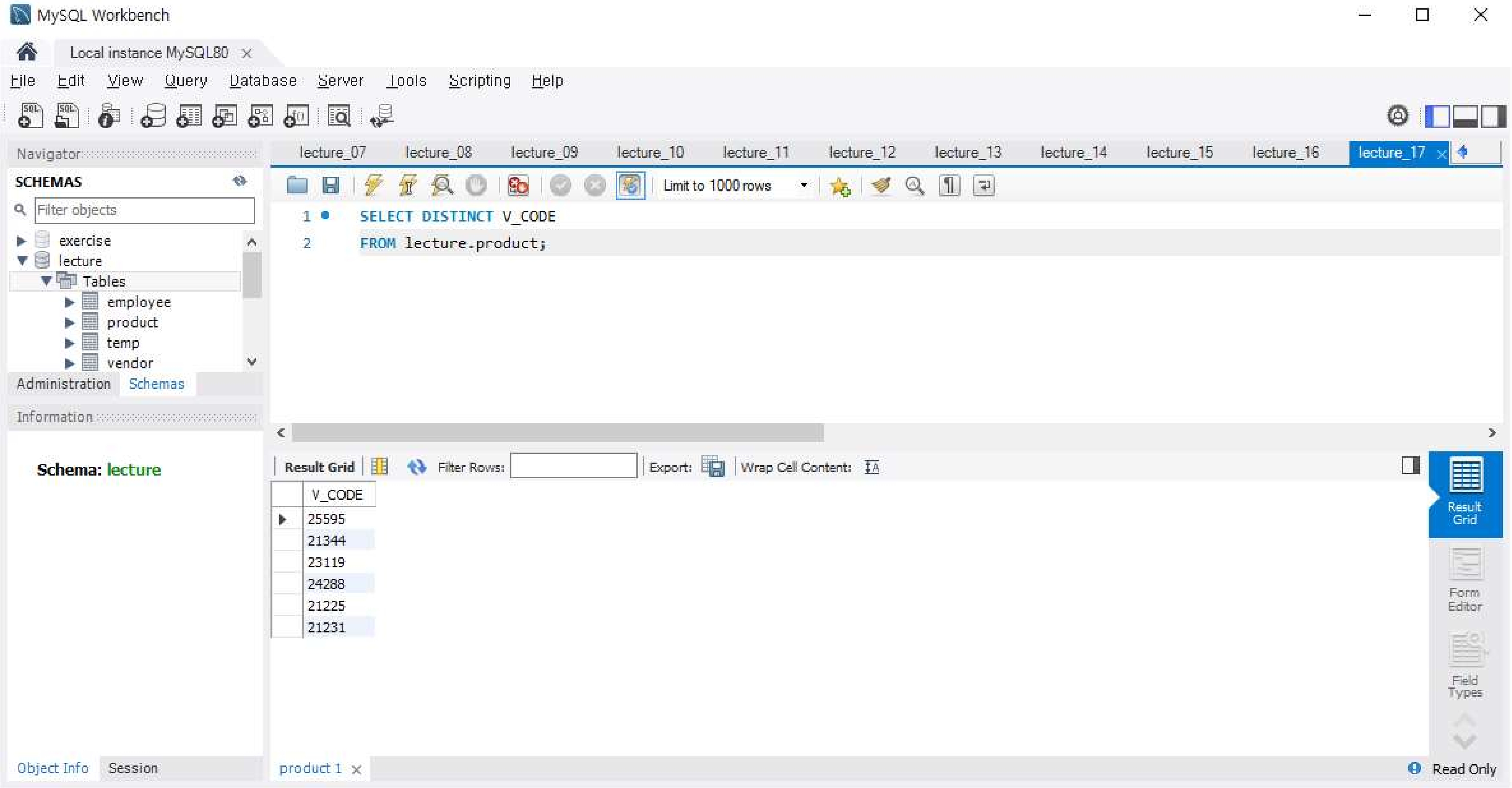


```
SELECT P_DESCRIPT, V_CODE, P_INDATE, P_PRICE
FROM lecture.product
WHERE P_INDATE < 20060121 AND P_PRICE <= 50
ORDER BY V_CODE, P_PRICE DESC;
```

P_DESCRIPT	V_CODE	P_INDATE	P_PRICE
Claw hammer	21225	2006-01-20	9.95
9.00-in. pwr. saw blade	21344	2005-11-13	17.49
7.25-in. pwr. saw blade	21344	2005-12-13	14.99
Rat-tail file, 1/8-in. fine	21344	2005-12-15	4.99
Hrd. cloth, 1/2-in., 3x50	23119	2006-01-15	43.99
Hrd. cloth, 1/4-in., 2x50	23119	2006-01-15	39.95
B&D cordless drill, 1/2-in.	25595	2006-01-20	38.95

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Listing Unique Values



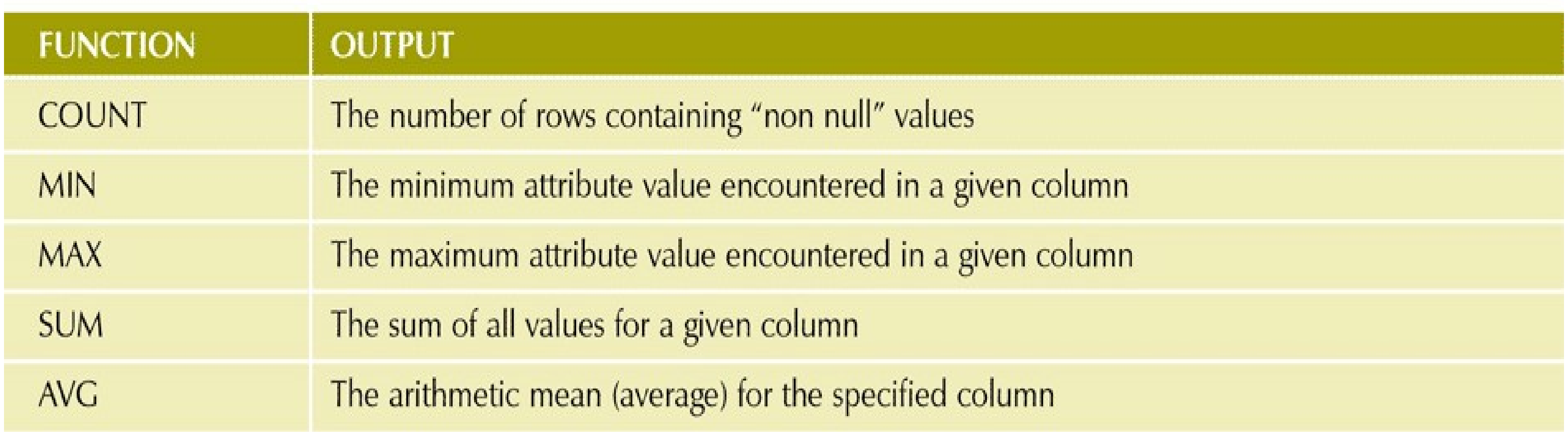
```sql
SELECT DISTINCT V_CODE
FROM lecture.product;
```

V_CODE
25595
21344
23119
24288
21225
21231

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

3 Basic Functions

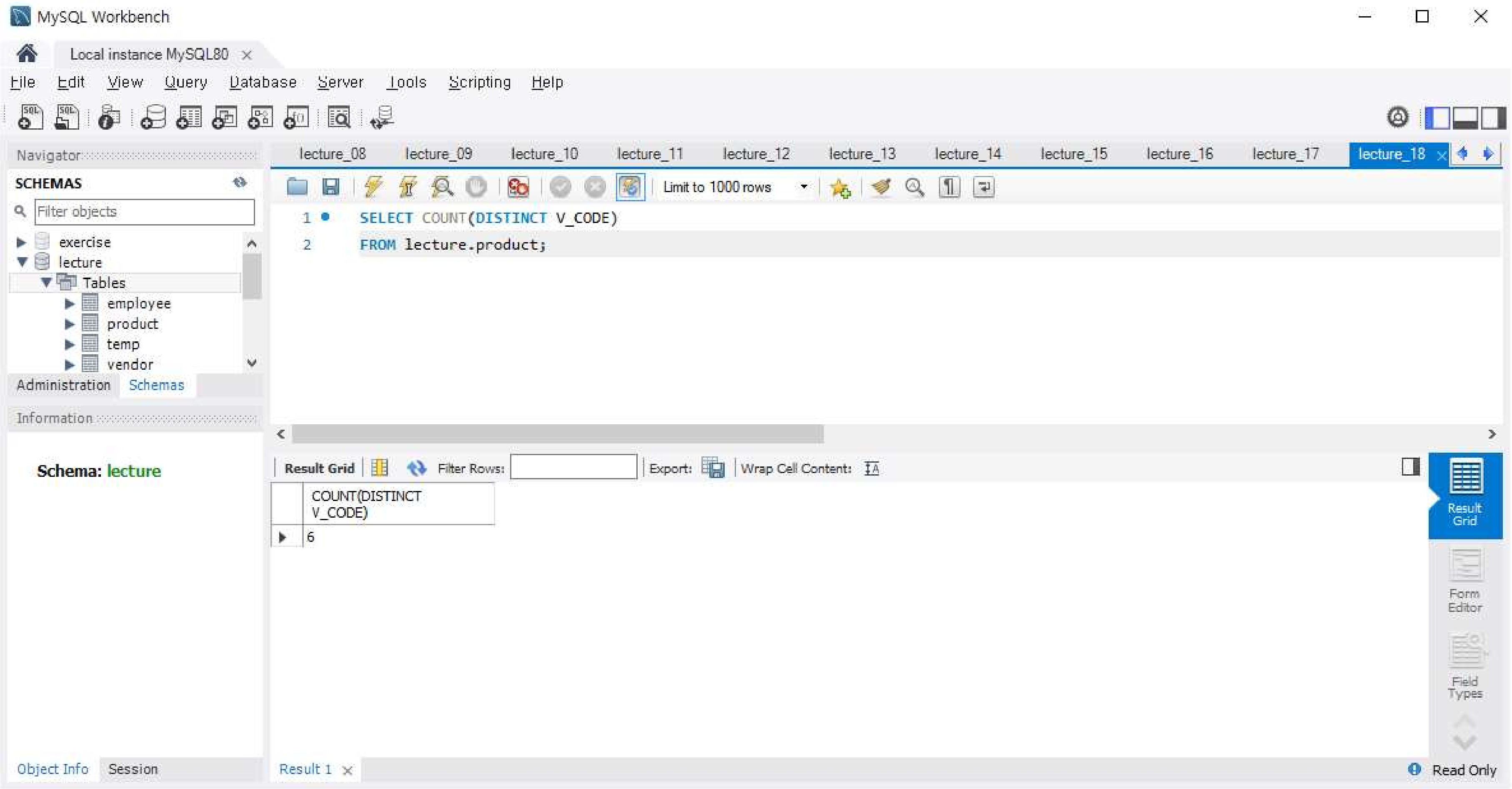
Some Basic SQL Aggregate Functions



FUNCTION	OUTPUT
COUNT	The number of rows containing “non null” values
MIN	The minimum attribute value encountered in a given column
MAX	The maximum attribute value encountered in a given column
SUM	The sum of all values for a given column
AVG	The arithmetic mean (average) for the specified column

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

COUNT Function (1)

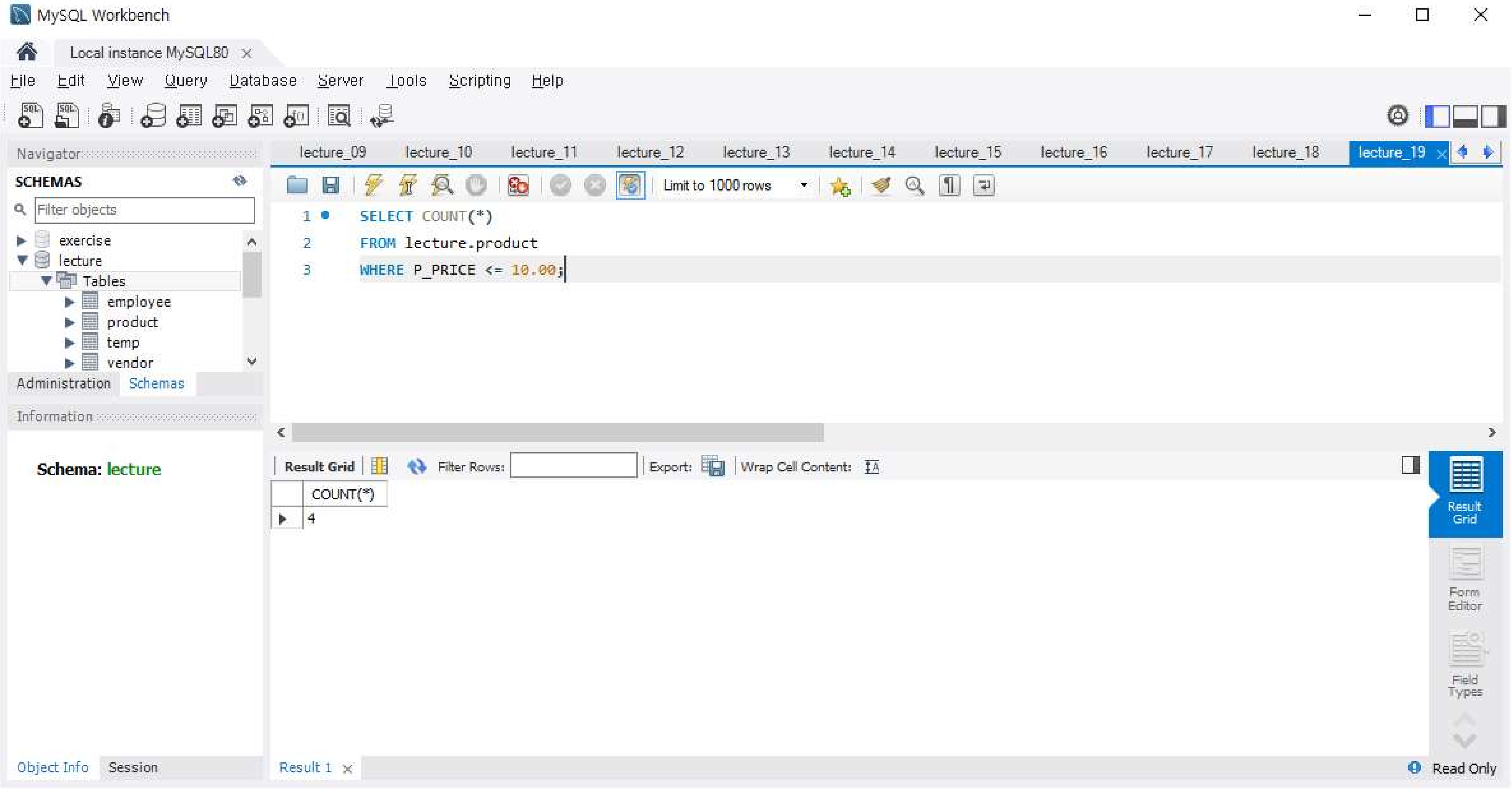


```
SELECT COUNT(DISTINCT V_CODE)
FROM lecture.product;
```

COUNT(DISTINCT V_CODE)
6

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

COUNT Function (2)



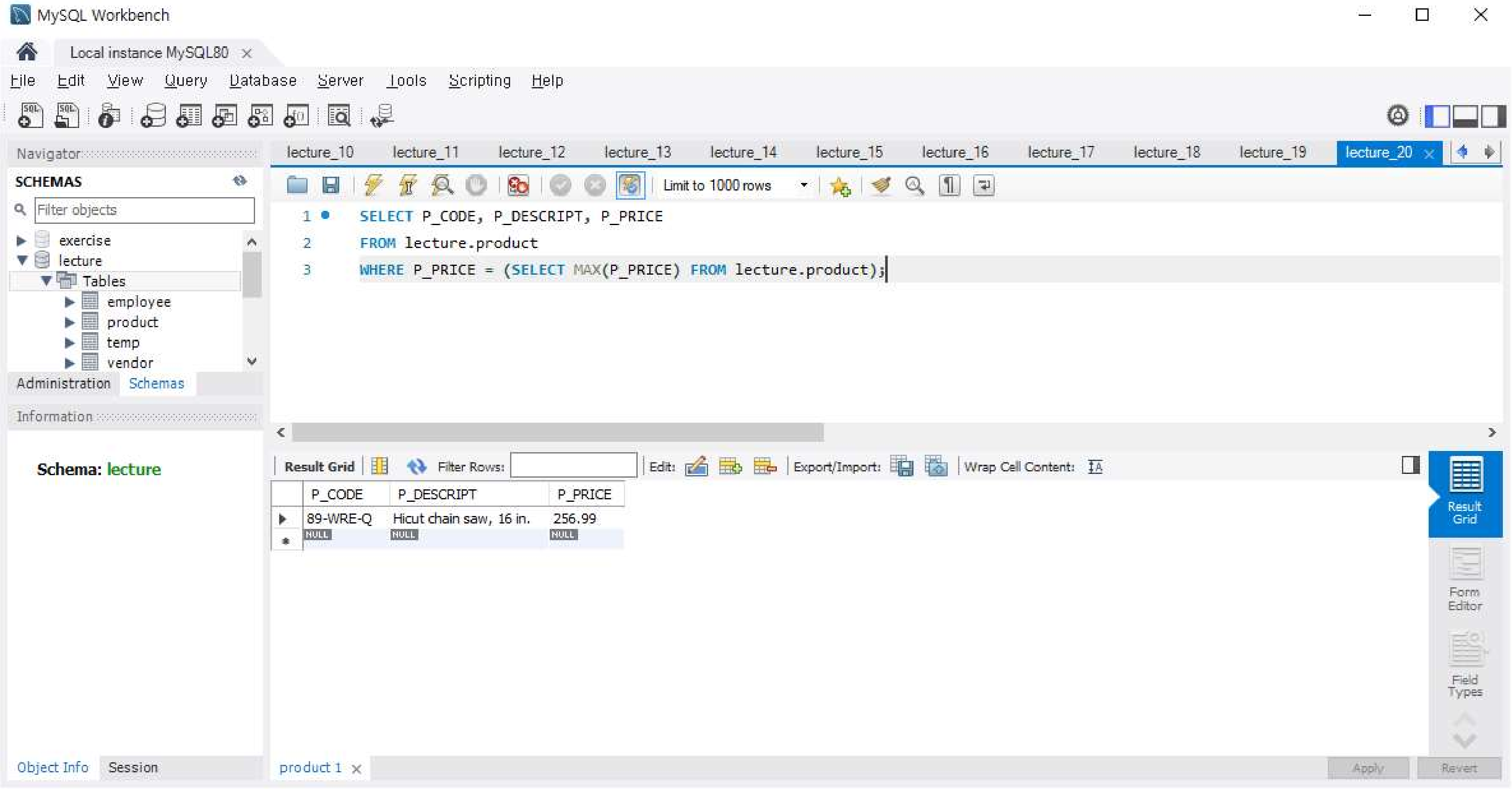
```
SELECT COUNT(*)
FROM lecture.product
WHERE P_PRICE <= 10.00;
```

Schema: lecture

```
COUNT(*)
4
```

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

MAX and MIN Function

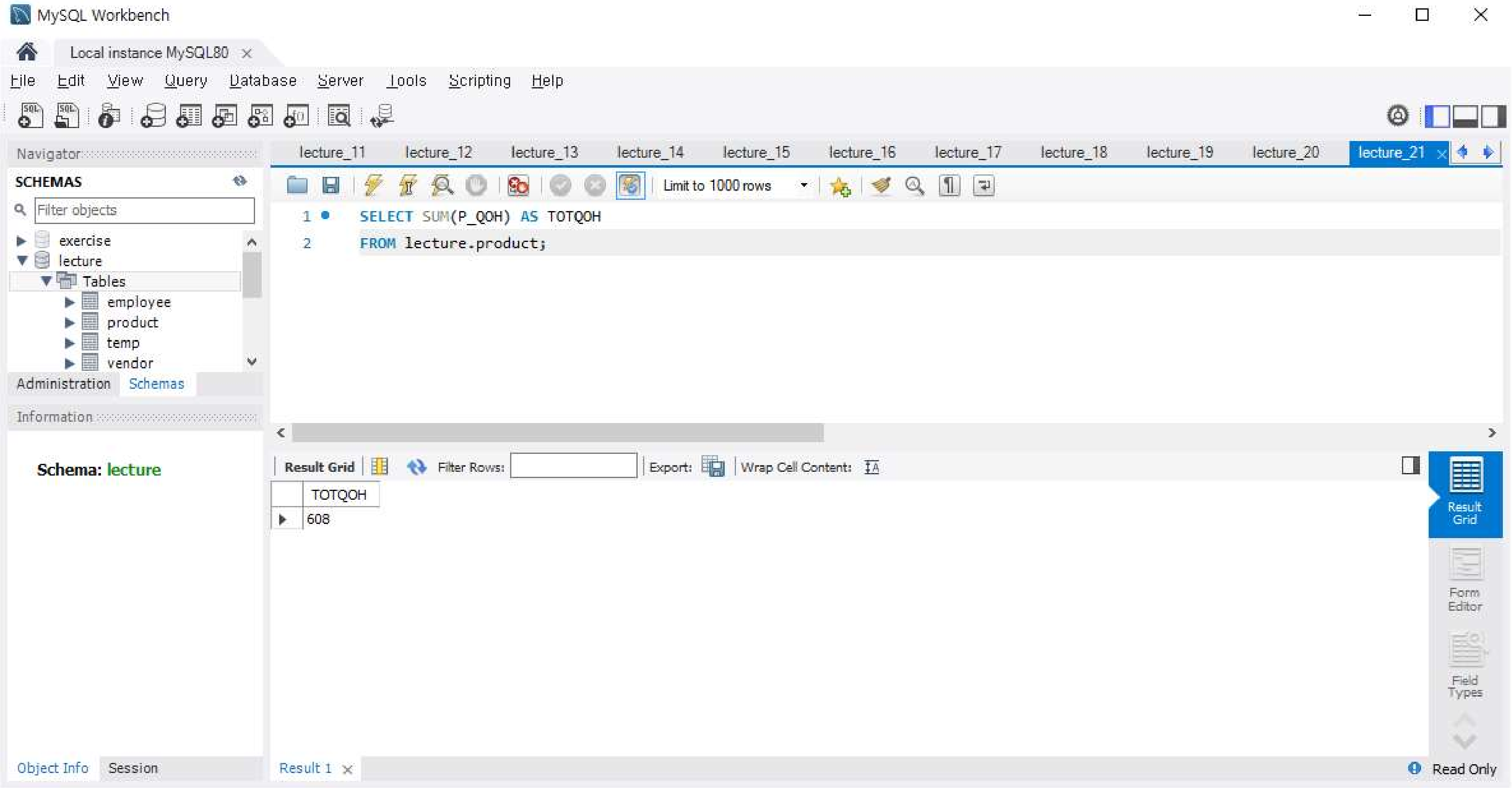


```
SELECT P_CODE, P_DESCRIPT, P_PRICE
FROM lecture.product
WHERE P_PRICE = (SELECT MAX(P_PRICE) FROM lecture.product);
```

P_CODE	P_DESCRIPT	P_PRICE
89-WRE-Q	Hicut chain saw, 16 in.	256.99
NULL	NULL	NULL

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

SUM Function (1)



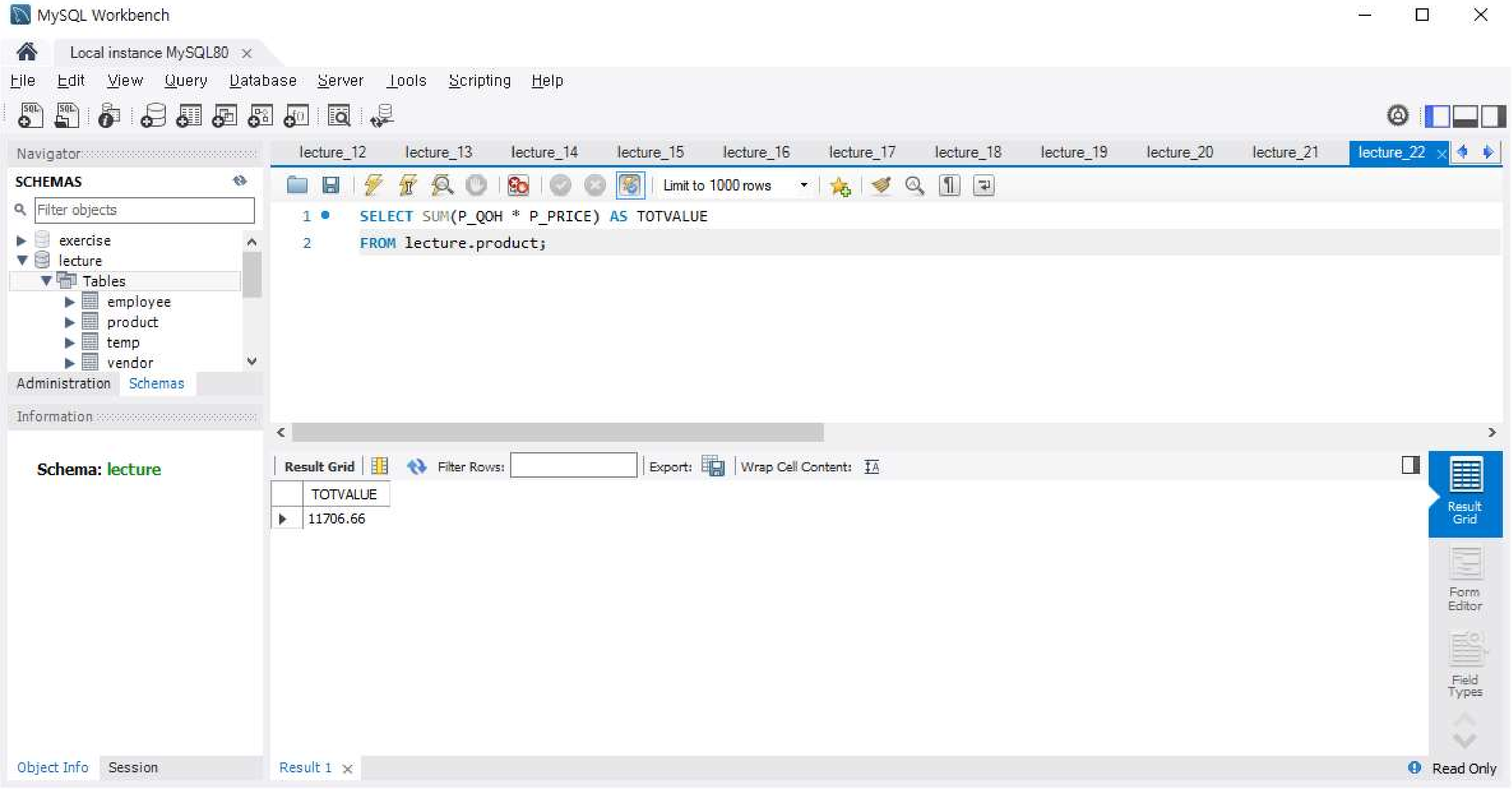
```
SELECT SUM(P_QOH) AS TOTQOH
FROM lecture.product;
```

Schema: lecture

TOTQOH
608

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

SUM Function (2)

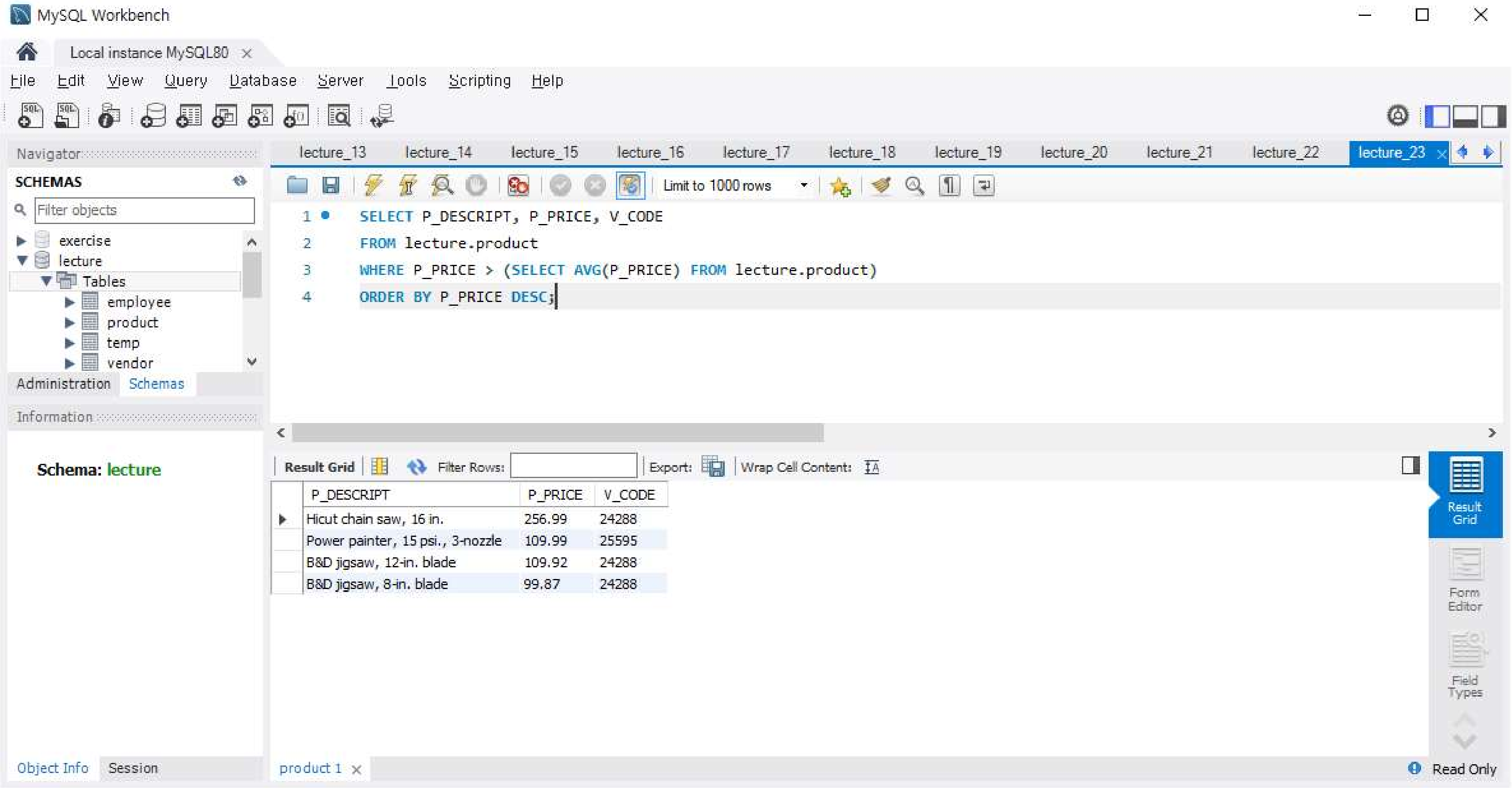


```
SELECT SUM(P_QOH * P_PRICE) AS TOTVALUE
FROM lecture.product;
```

TOTVALUE
11706.66

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

AVG Function



Schema: lecture

```sql
SELECT P_DESCRIPT, P_PRICE, V_CODE
FROM lecture.product
WHERE P_PRICE > (SELECT AVG(P_PRICE) FROM lecture.product)
ORDER BY P_PRICE DESC;
```

P_DESCRIPT	P_PRICE	V_CODE
Hicut chain saw, 16 in.	256.99	24288
Power painter, 15 psi., 3-nozzle	109.99	25595
B&D jigsaw, 12-in. blade	109.92	24288
B&D jigsaw, 8-in. blade	99.87	24288

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Joining Database Tables

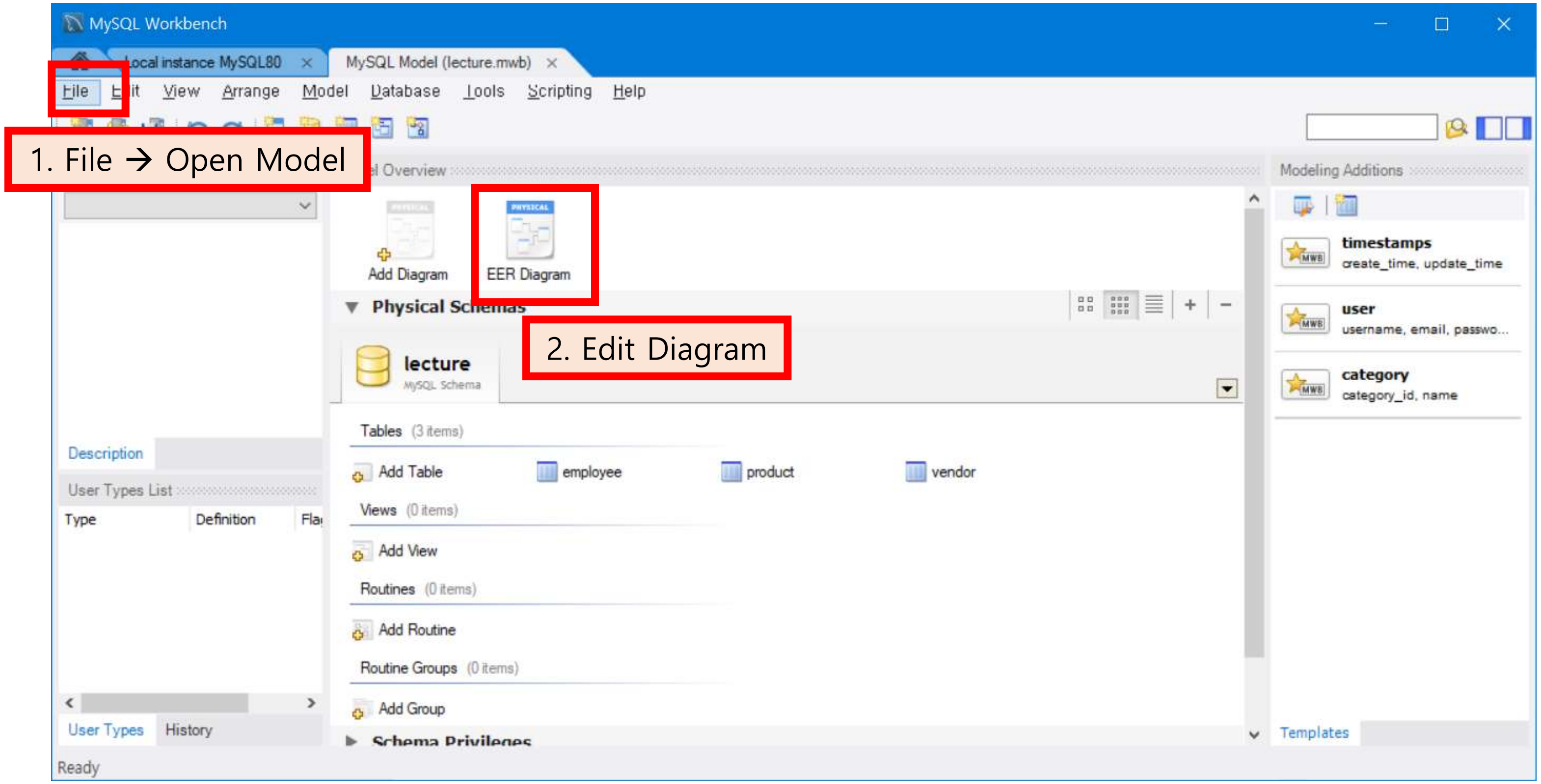
JOIN Function

- Ability to combine (join) tables on common attributes is most important distinction between a relational database and other databases
- Join is performed when data are retrieved from more than one table at a time
- Join is generally composed of an equality comparison between the foreign key and the primary key of related tables

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Creating Links Through Foreign Keys

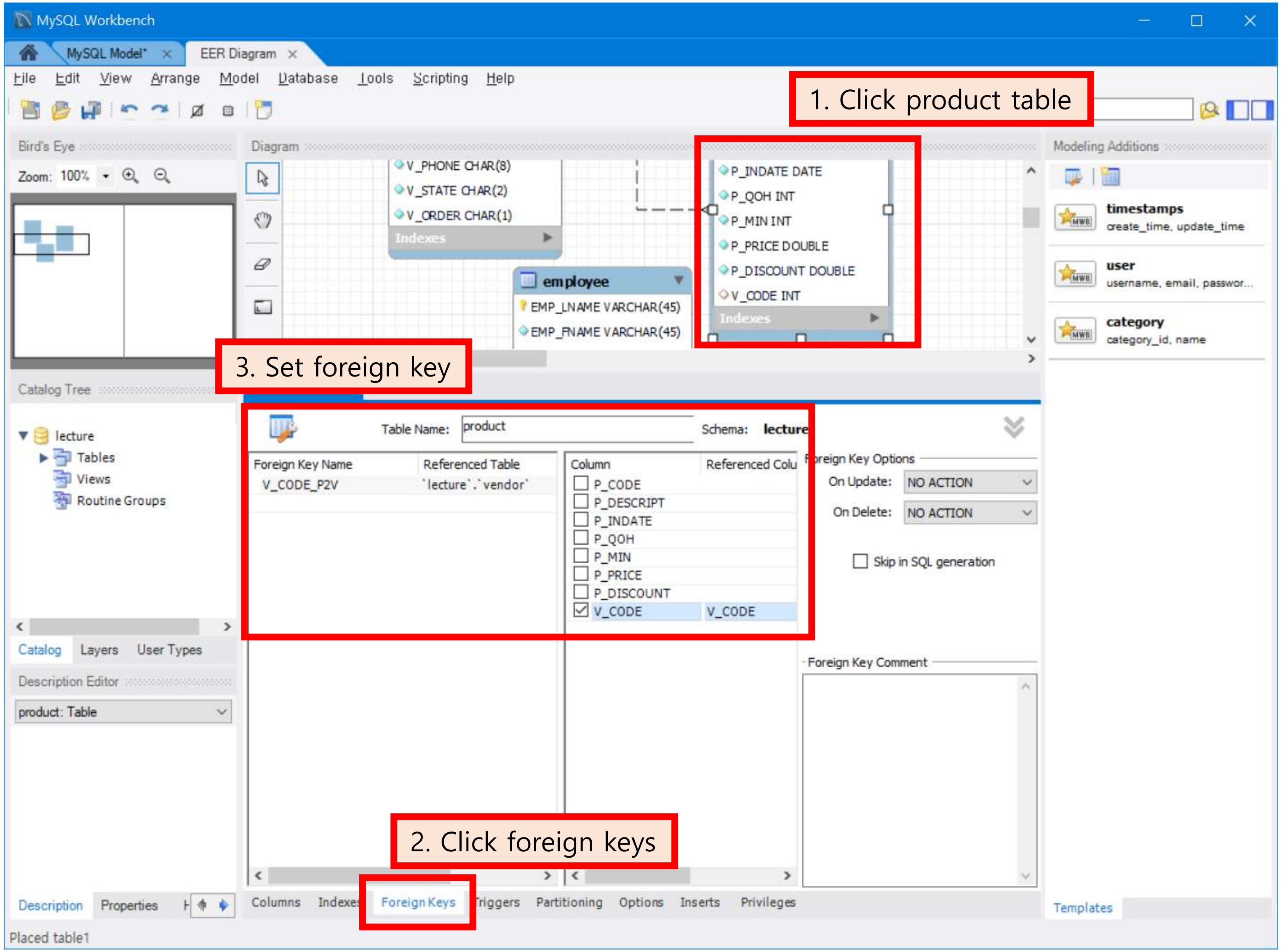
TABLE	ATTRIBUTES TO BE SHOWN	LINKING ATTRIBUTE
PRODUCT	P_DESCRIPT, P_PRICE	V_CODE
VENDOR	V_COMPANY, V_PHONE	V_CODE



1. File → Open Model

2. Edit Diagram

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management



1. Click product table

3. Set foreign key

Table Name: product
Schema: **lecture**

Foreign Key Name	Referenced Table
V_CODE_P2V	`lecture`.`vendor`

Column	Referenced Colu
☐ P_CODE	
☐ P_DESCRIPT	
☐ P_INDATE	
☐ P_QOH	
☐ P_MIN	
☐ P_PRICE	
☐ P_DISCOUNT	
☑ V_CODE	V_CODE

Foreign Key Options

On Update: NO ACTION

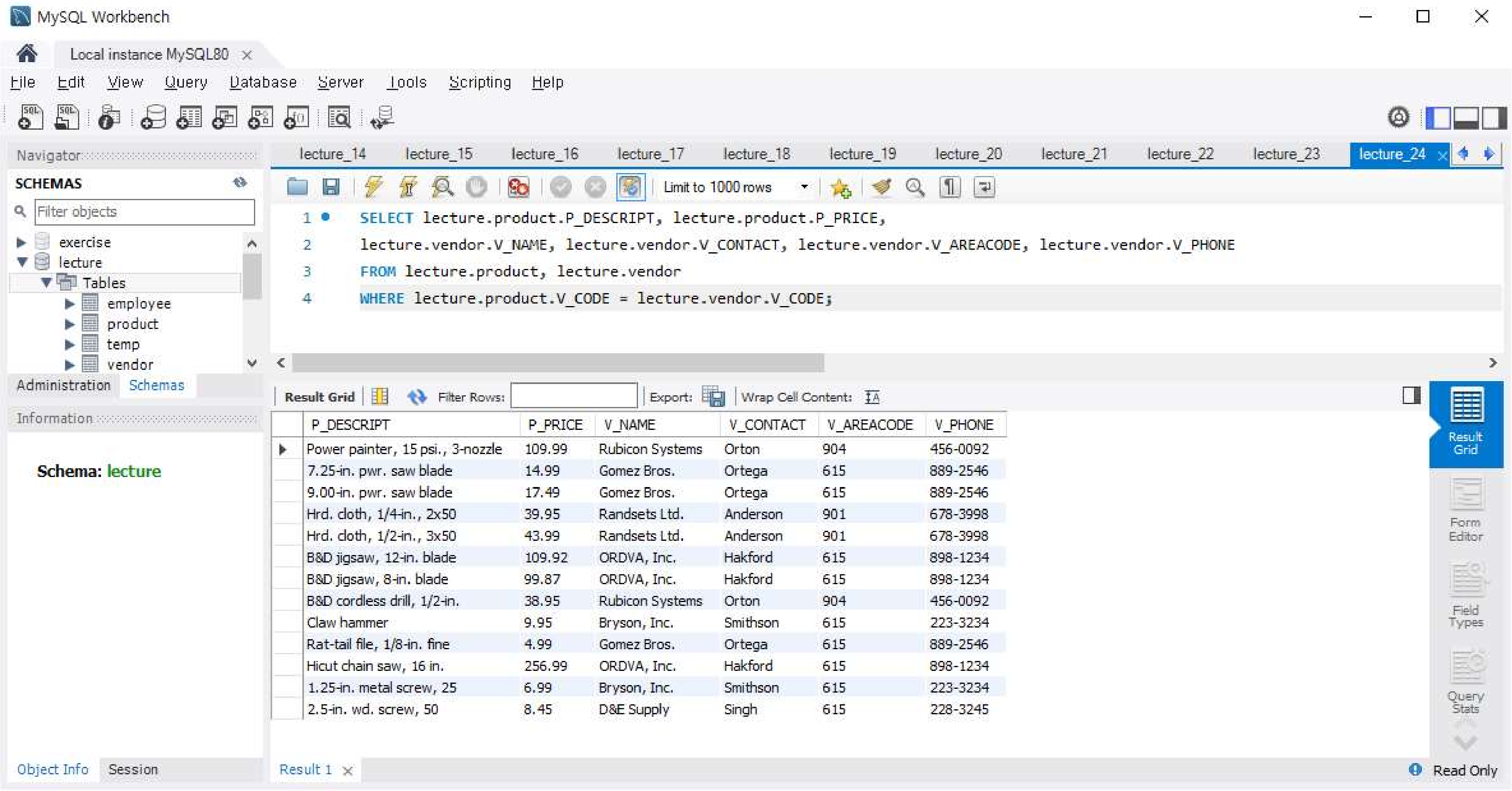
On Delete: NO ACTION

☐ Skip in SQL generation

2. Click foreign keys

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Joining Tables

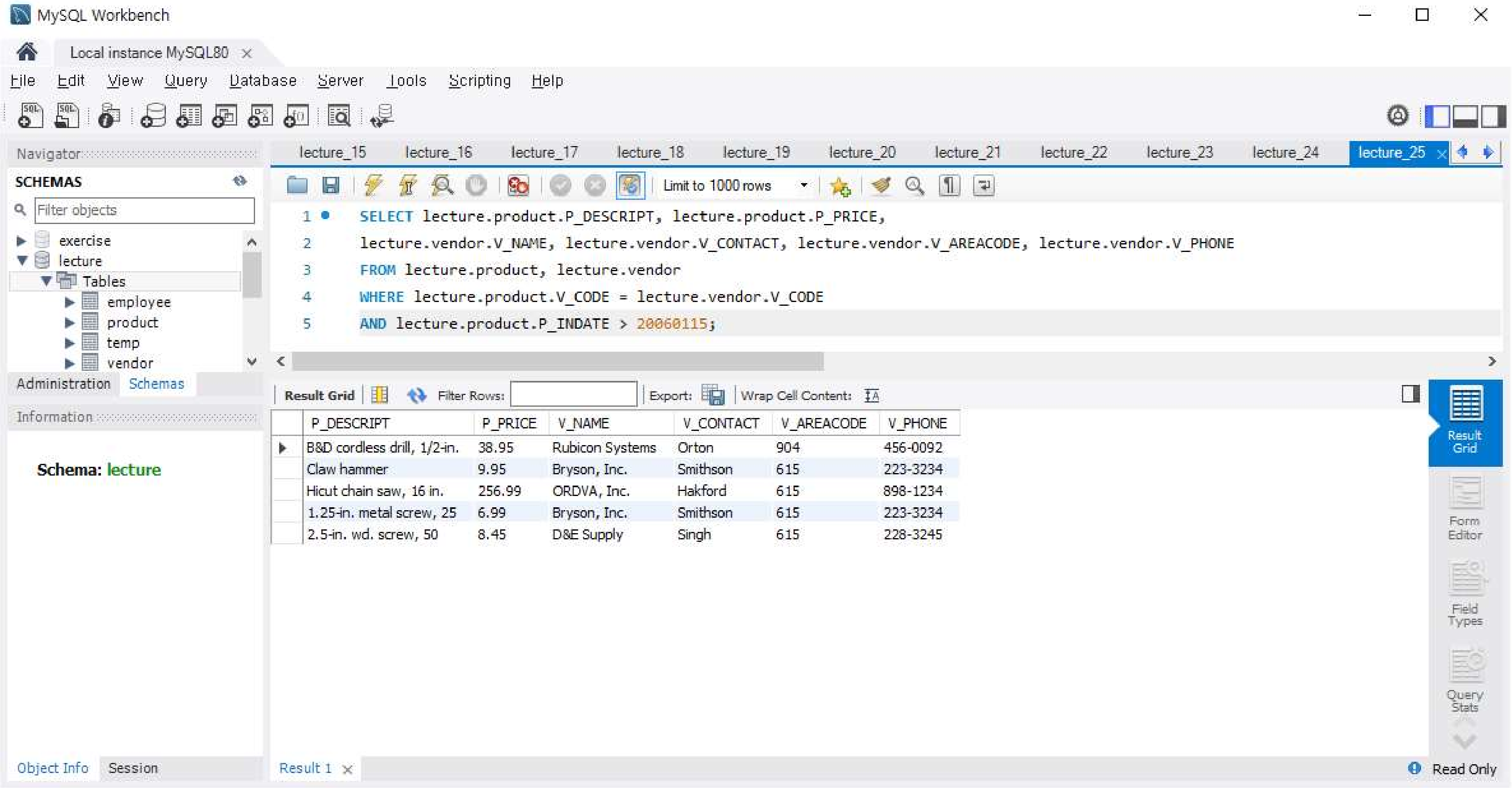


```sql
SELECT lecture.product.P_DESCRIPT, lecture.product.P_PRICE,
lecture.vendor.V_NAME, lecture.vendor.V_CONTACT, lecture.vendor.V_AREACODE, lecture.vendor.V_PHONE
FROM lecture.product, lecture.vendor
WHERE lecture.product.V_CODE = lecture.vendor.V_CODE;
```

P_DESCRIPT	P_PRICE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE
Power painter, 15 psi., 3-nozzle	109.99	Rubicon Systems	Orton	904	456-0092
7.25-in. pwr. saw blade	14.99	Gomez Bros.	Ortega	615	889-2546
9.00-in. pwr. saw blade	17.49	Gomez Bros.	Ortega	615	889-2546
Hrd. cloth, 1/4-in., 2x50	39.95	Randsets Ltd.	Anderson	901	678-3998
Hrd. cloth, 1/2-in., 3x50	43.99	Randsets Ltd.	Anderson	901	678-3998
B&D jigsaw, 12-in. blade	109.92	ORDVA, Inc.	Hakford	615	898-1234
B&D jigsaw, 8-in. blade	99.87	ORDVA, Inc.	Hakford	615	898-1234
B&D cordless drill, 1/2-in.	38.95	Rubicon Systems	Orton	904	456-0092
Claw hammer	9.95	Bryson, Inc.	Smithson	615	223-3234
Rat-tail file, 1/8-in. fine	4.99	Gomez Bros.	Ortega	615	889-2546
Hicut chain saw, 16 in.	256.99	ORDVA, Inc.	Hakford	615	898-1234
1.25-in. metal screw, 25	6.99	Bryson, Inc.	Smithson	615	223-3234
2.5-in. wd. screw, 50	8.45	D&E Supply	Singh	615	228-3245

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

An Ordered and Limited Listing After a JOIN



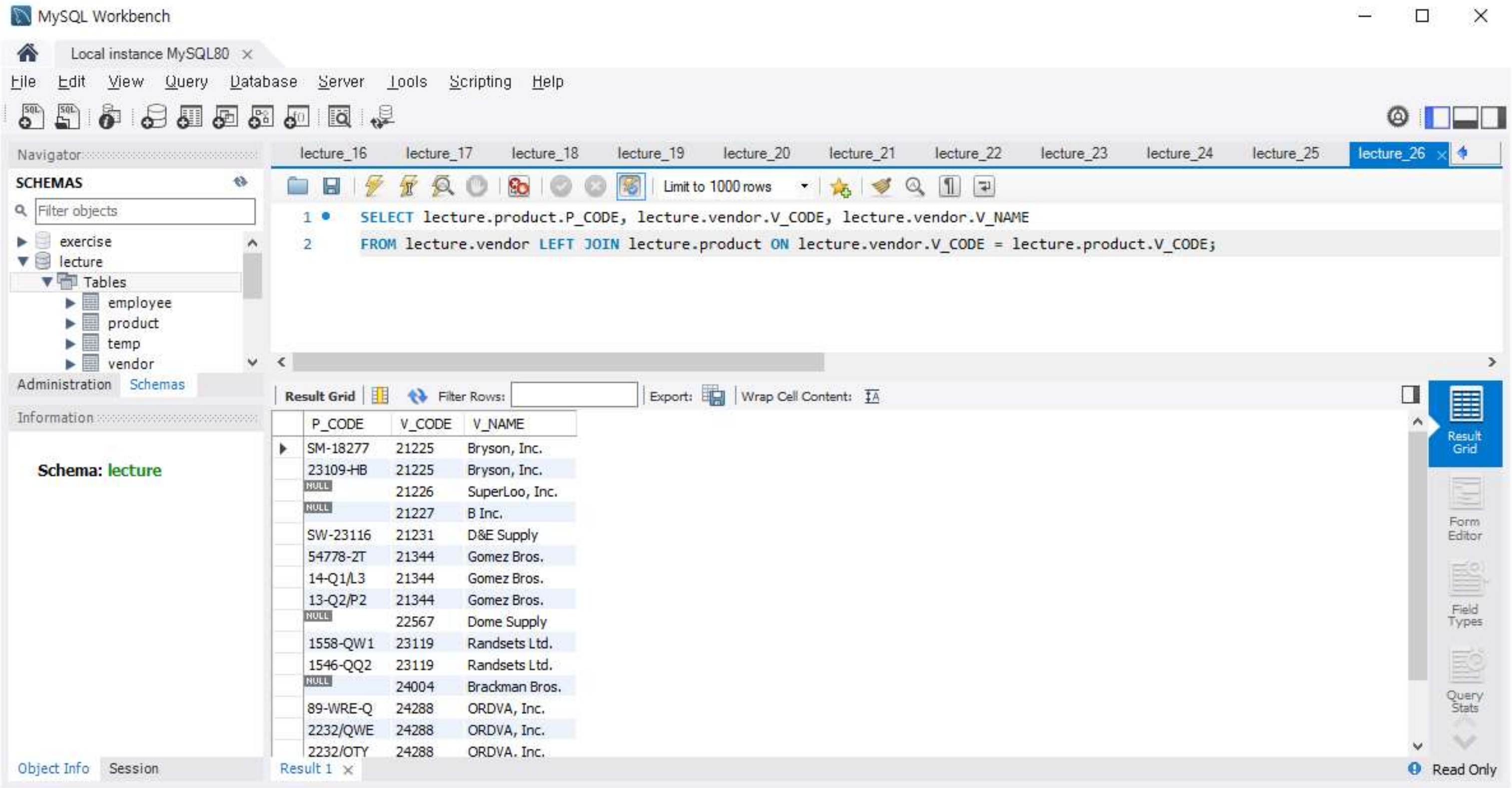
```sql
SELECT lecture.product.P_DESCRIPT, lecture.product.P_PRICE,
lecture.vendor.V_NAME, lecture.vendor.V_CONTACT, lecture.vendor.V_AREACODE, lecture.vendor.V_PHONE
FROM lecture.product, lecture.vendor
WHERE lecture.product.V_CODE = lecture.vendor.V_CODE
AND lecture.product.P_INDATE > 20060115;
```

P_DESCRIPT	P_PRICE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE
B&D cordless drill, 1/2-in.	38.95	Rubicon Systems	Orton	904	456-0092
Claw hammer	9.95	Bryson, Inc.	Smithson	615	223-3234
Hicut chain saw, 16 in.	256.99	ORDVA, Inc.	Hakford	615	898-1234
1.25-in. metal screw, 25	6.99	Bryson, Inc.	Smithson	615	223-3234
2.5-in. wd. screw, 50	8.45	D&E Supply	Singh	615	228-3245

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Left Outer Join

- With just JOIN operation, null values are ignored (e.g., no matching V_CODE between PRODUCT & VENDOR)
- In this LEFT JOIN, show all VENDOR rows and all matching PRODUCT rows



Schema: lecture

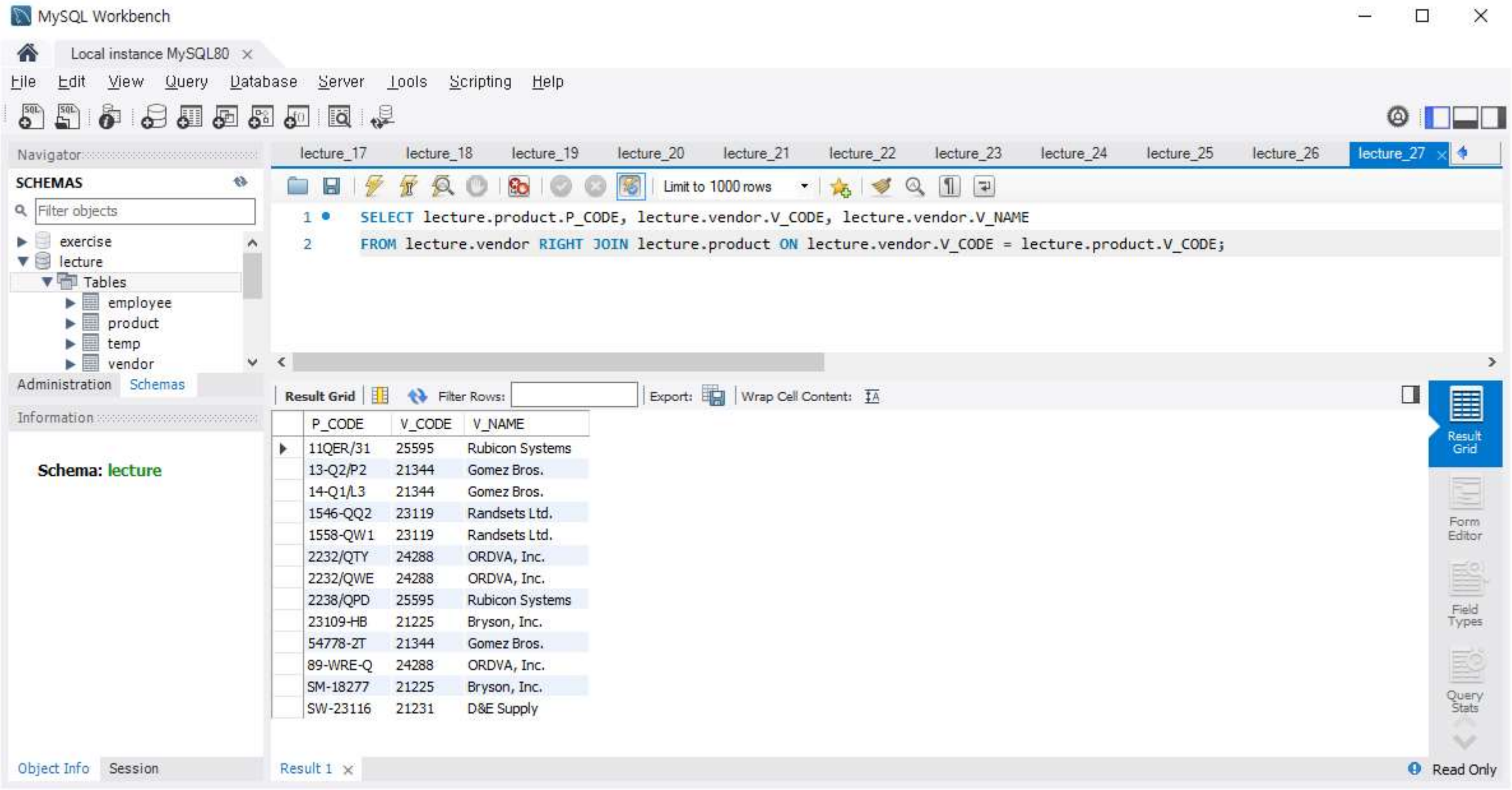
```sql
SELECT lecture.product.P_CODE, lecture.vendor.V_CODE, lecture.vendor.V_NAME
FROM lecture.vendor LEFT JOIN lecture.product ON lecture.vendor.V_CODE = lecture.product.V_CODE;
```

P_CODE	V_CODE	V_NAME
SM-18277	21225	Bryson, Inc.
23109-HB	21225	Bryson, Inc.
NULL	21226	SuperLoo, Inc.
NULL	21227	B Inc.
SW-23116	21231	D&E Supply
54778-2T	21344	Gomez Bros.
14-Q1/L3	21344	Gomez Bros.
13-Q2/P2	21344	Gomez Bros.
NULL	22567	Dome Supply
1558-QW1	23119	Randsets Ltd.
1546-QQ2	23119	Randsets Ltd.
NULL	24004	Brackman Bros.
89-WRE-Q	24288	ORDVA, Inc.
2232/QWE	24288	ORDVA, Inc.
2232/QTY	24288	ORDVA, Inc.

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

Right Outer Join

- In this RIGHT JOIN, show all PRODUCT rows and all matching VENDOR rows



```sql
SELECT lecture.product.P_CODE, lecture.vendor.V_CODE, lecture.vendor.V_NAME
FROM lecture.vendor RIGHT JOIN lecture.product ON lecture.vendor.V_CODE = lecture.product.V_CODE;
```

P_CODE	V_CODE	V_NAME
11QER/31	25595	Rubicon Systems
13-Q2/P2	21344	Gomez Bros.
14-Q1/L3	21344	Gomez Bros.
1546-QQ2	23119	Randsets Ltd.
1558-QW1	23119	Randsets Ltd.
2232/QTY	24288	ORDVA, Inc.
2232/QWE	24288	ORDVA, Inc.
2238/QPD	25595	Rubicon Systems
23109-HB	21225	Bryson, Inc.
54778-2T	21344	Gomez Bros.
89-WRE-Q	24288	ORDVA, Inc.
SM-18277	21225	Bryson, Inc.
SW-23116	21231	D&E Supply

From Rob and Coronel (2004), Database Systems: Design, Implementation, and Management

MySQL Installation

1 Introduction

MySQL

- https://www.mysql.com/

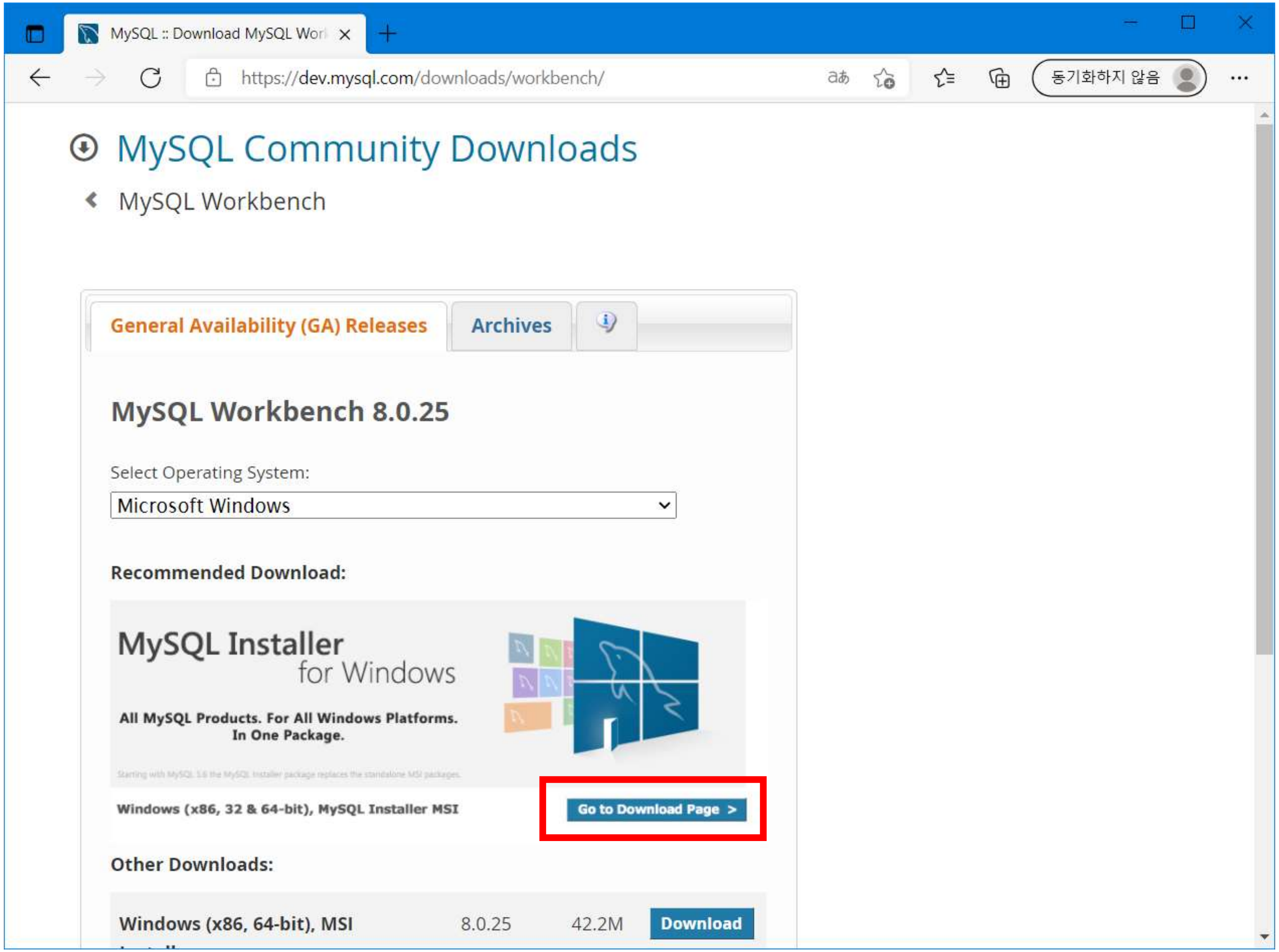
- An open-sourced Relational DataBase Management System (RDBMS) that is the most commonly used over the world
- Operated by Oracle
- Supports various programming languages including C/C++/C#/Java/Python/Ruby/PHP/…

MySQL Workbench

- https://dev.mysql.com/downloads/workbench/

- A visual tool for the user to manage the MySQL database

2 Installation



MySQL :: Download MySQL Wor

https://dev.mysql.com/downloads/workbench/

동기화하지 않음

MySQL Community Downloads

MySQL Workbench

General Availability (GA) Releases | Archives

MySQL Workbench 8.0.25

Select Operating System:

Microsoft Windows

Recommended Download:

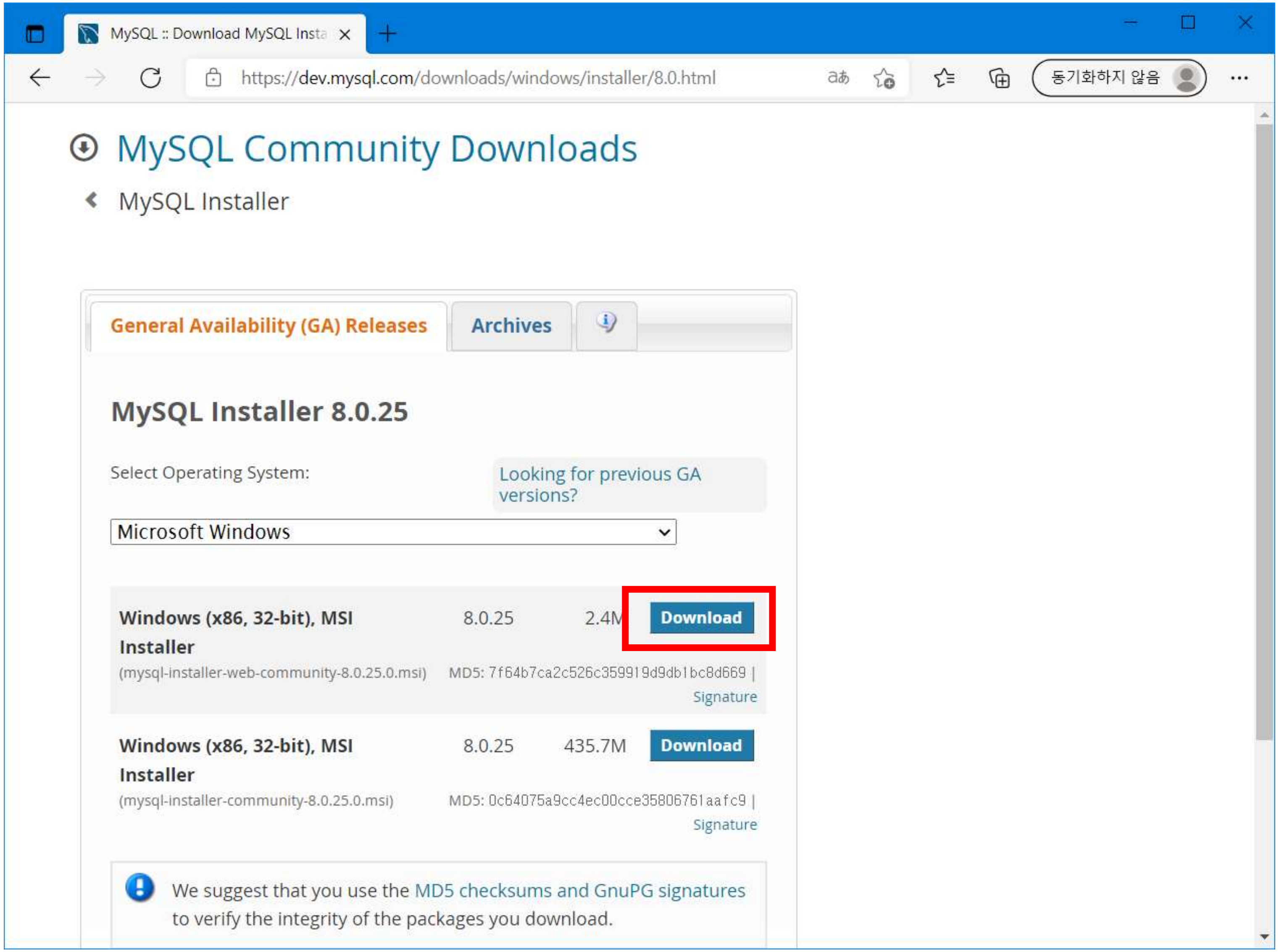
MySQL Installer
for Windows
All MySQL Products. For All Windows Platforms.
In One Package.
Starting with MySQL 5.6 the MySQL Installer package replaces the standalone MSI packages.

Windows (x86, 32 & 64-bit), MySQL Installer MSI

Go to Download Page >

Other Downloads:

Windows (x86, 64-bit), MSI Installer	8.0.25	42.2M	Download



MySQL :: Download MySQL Insta
https://dev.mysql.com/downloads/windows/installer/8.0.html
동기화하지 않음

MySQL Community Downloads

MySQL Installer

General Availability (GA) Releases
Archives

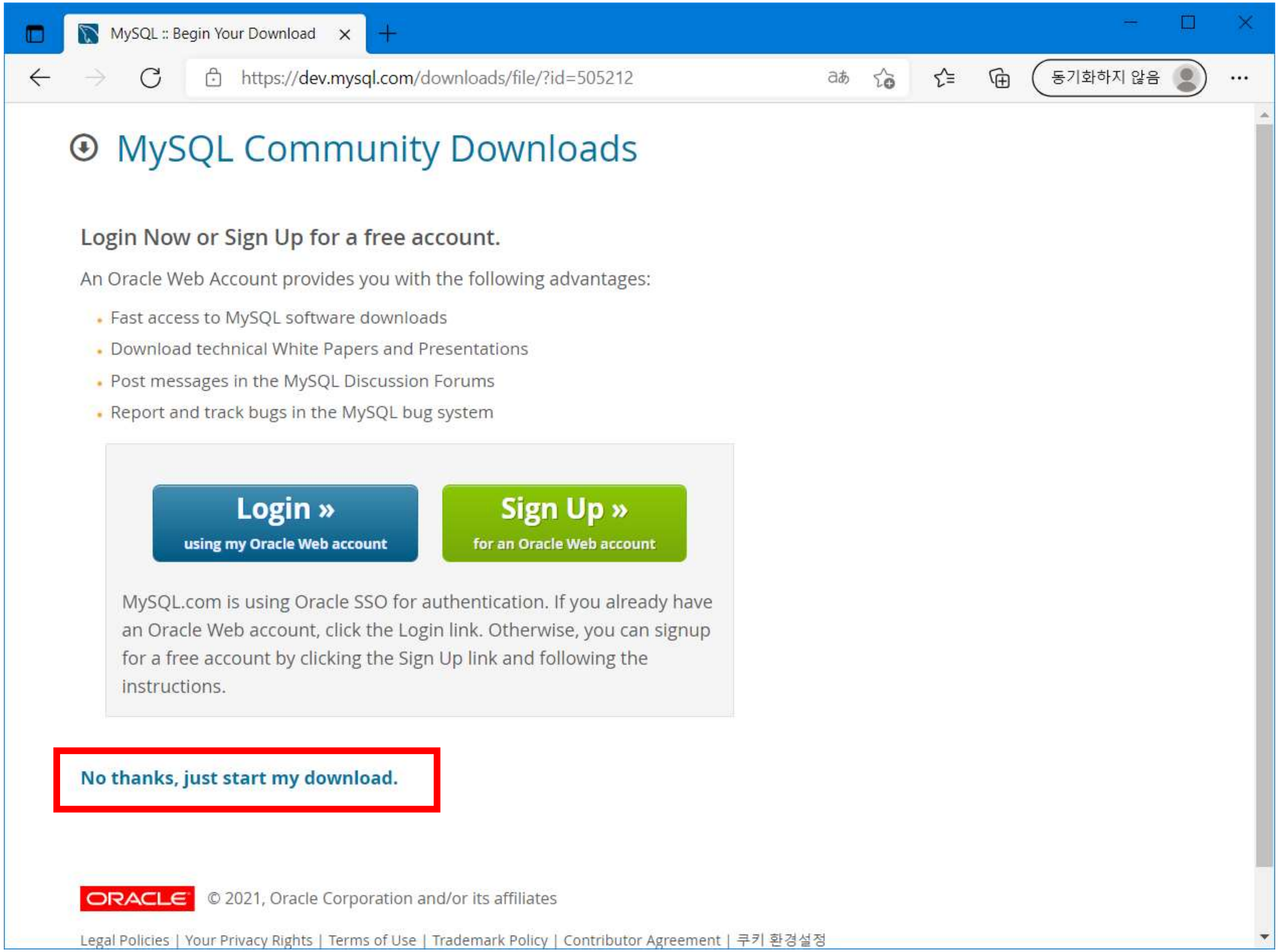
MySQL Installer 8.0.25

Select Operating System: Microsoft Windows

Looking for previous GA versions?

Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-8.0.25.0.msi)	8.0.25	2.4M	Download MD5: 7f64b7ca2c526c359919d9db1bc8d669 | Signature
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-8.0.25.0.msi)	8.0.25	435.7M	Download MD5: 0c64075a9cc4ec00cce35806761aafc9 | Signature

We suggest that you use the MD5 checksums and GnuPG signatures to verify the integrity of the packages you download.



MySQL :: Begin Your Download
https://dev.mysql.com/downloads/file/?id=505212

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

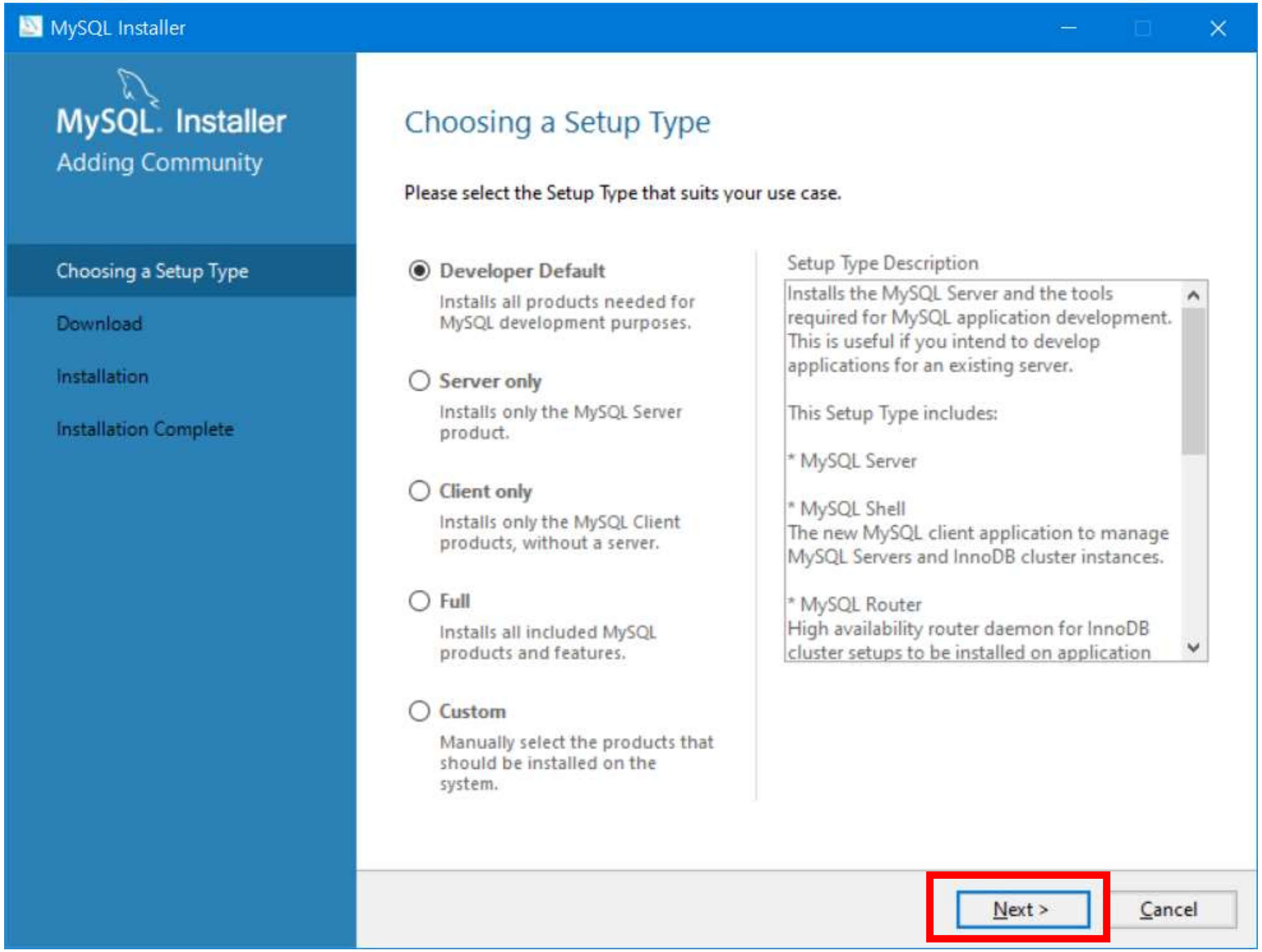
- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

Login »
using my Oracle Web account
Sign Up »
for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions.

No thanks, just start my download.

Legal Policies | Your Privacy Rights | Terms of Use | Trademark Policy | Contributor Agreement | 쿠키 환경설정



Choosing a Setup Type

Please select the Setup Type that suits your use case.

● Developer Default
Installs all products needed for MySQL development purposes.

○ Server only
Installs only the MySQL Server product.

○ Client only
Installs only the MySQL Client products, without a server.

○ Full
Installs all included MySQL products and features.

○ Custom
Manually select the products that should be installed on the system.

Setup Type Description

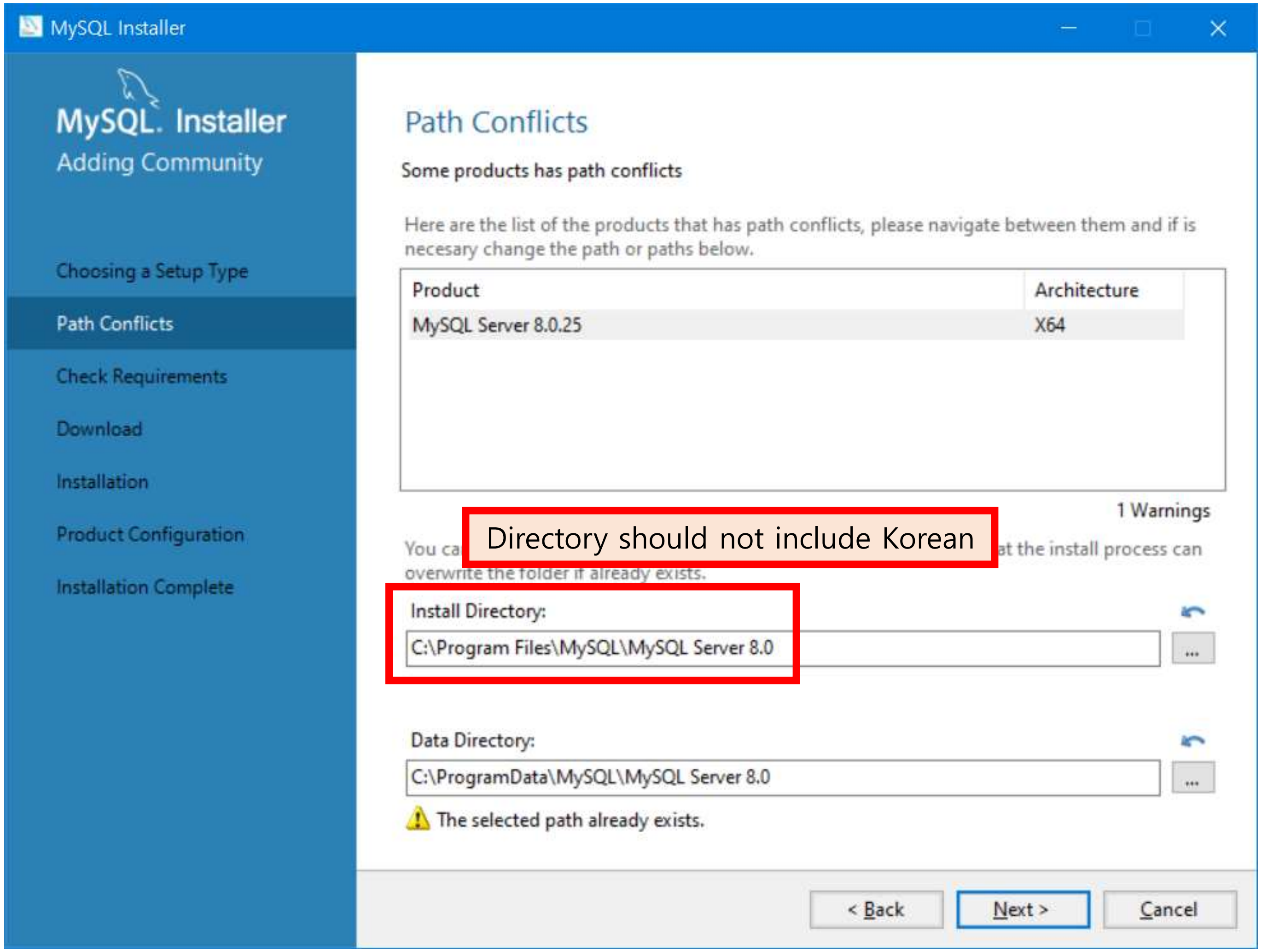
Installs the MySQL Server and the tools required for MySQL application development. This is useful if you intend to develop applications for an existing server.

This Setup Type includes:

* MySQL Server

* MySQL Shell
The new MySQL client application to manage MySQL Servers and InnoDB cluster instances.

* MySQL Router
High availability router daemon for InnoDB cluster setups to be installed on application



Path Conflicts

Some products has path conflicts

Here are the list of the products that has path conflicts, please navigate between them and if is necesary change the path or paths below.

Product	Architecture
MySQL Server 8.0.25	X64

1 Warnings

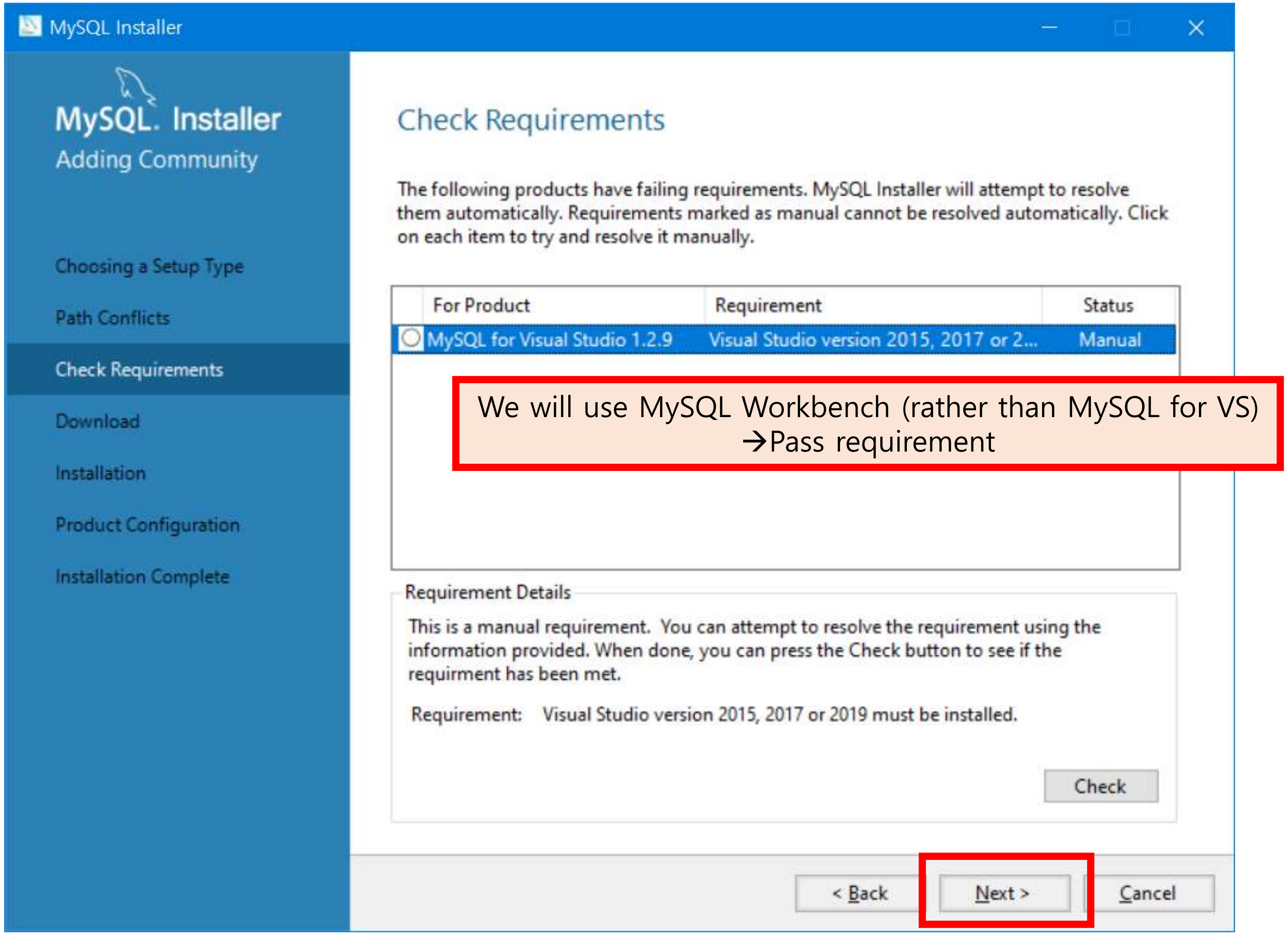
You ca... at the install process can overwrite the folder if already exists.

Install Directory: C:\Program Files\MySQL\MySQL Server 8.0

Data Directory: C:\ProgramData\MySQL\MySQL Server 8.0

⚠ The selected path already exists.

Directory should not include Korean



Check Requirements

The following products have failing requirements. MySQL Installer will attempt to resolve them automatically. Requirements marked as manual cannot be resolved automatically. Click on each item to try and resolve it manually.

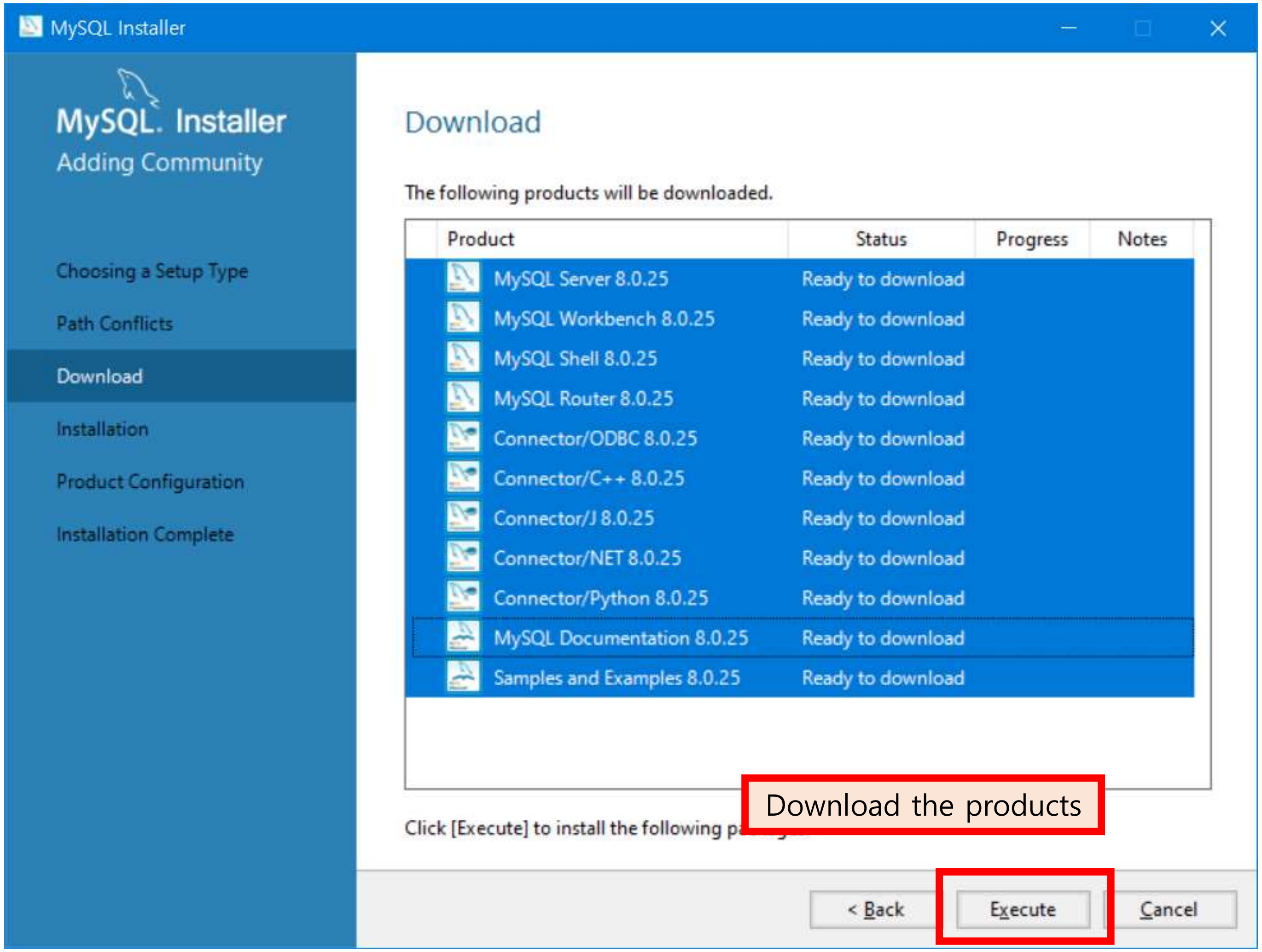
For Product	Requirement	Status
○ MySQL for Visual Studio 1.2.9	Visual Studio version 2015, 2017 or 2...	Manual

Requirement Details

This is a manual requirement. You can attempt to resolve the requirement using the information provided. When done, you can press the Check button to see if the requirment has been met.

Requirement: Visual Studio version 2015, 2017 or 2019 must be installed.

We will use MySQL Workbench (rather than MySQL for VS)
→Pass requirement



MySQL Installer

MySQL. Installer
Adding Community

- Choosing a Setup Type
- Path Conflicts
- Download
- Installation
- Product Configuration
- Installation Complete

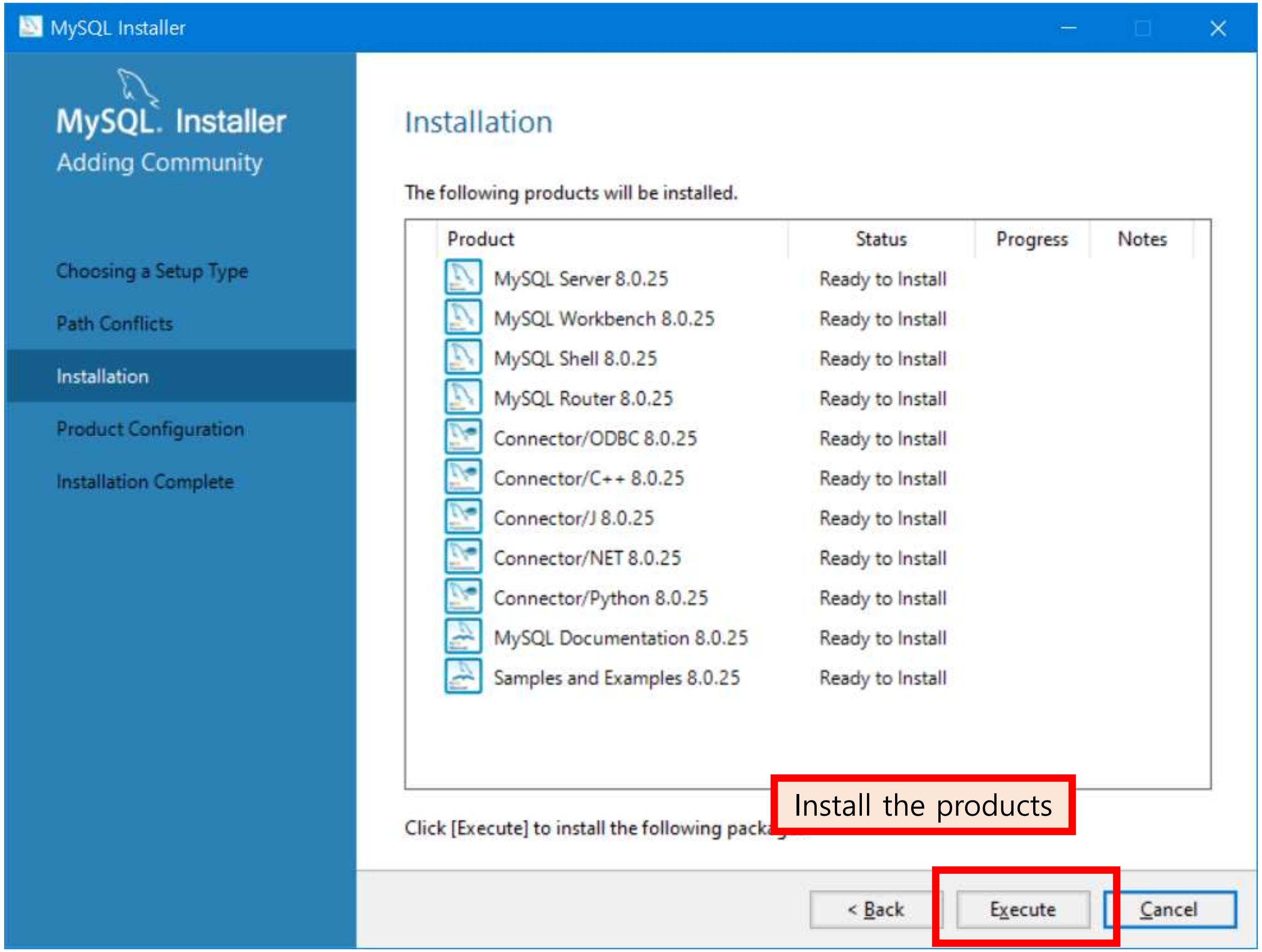
Download

The following products will be downloaded.

Product	Status	Progress	Notes
MySQL Server 8.0.25	Ready to download		
MySQL Workbench 8.0.25	Ready to download		
MySQL Shell 8.0.25	Ready to download		
MySQL Router 8.0.25	Ready to download		
Connector/ODBC 8.0.25	Ready to download		
Connector/C++ 8.0.25	Ready to download		
Connector/J 8.0.25	Ready to download		
Connector/NET 8.0.25	Ready to download		
Connector/Python 8.0.25	Ready to download		
MySQL Documentation 8.0.25	Ready to download		
Samples and Examples 8.0.25	Ready to download		

Click [Execute] to install the following pa

Download the products



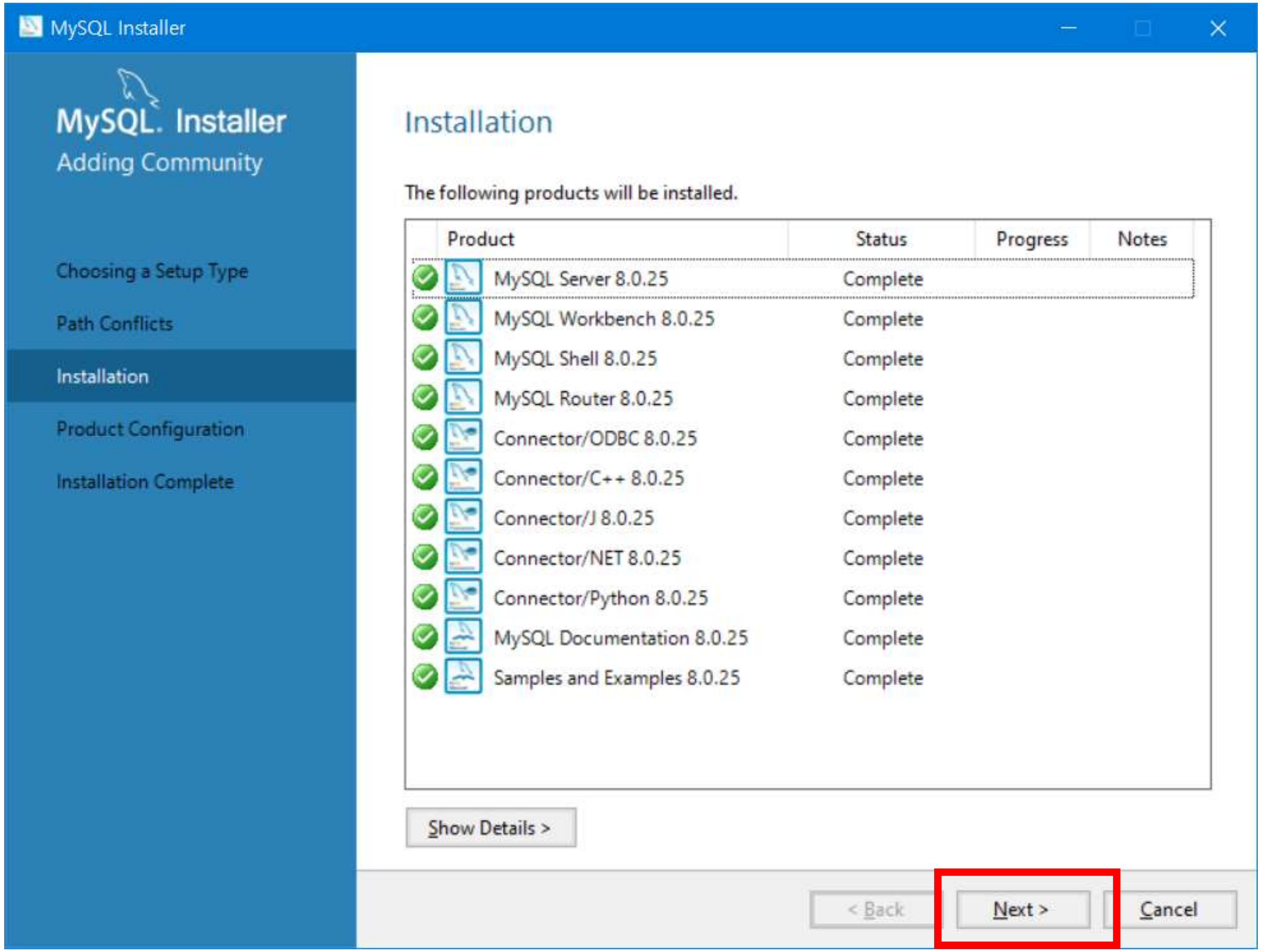
Installation

The following products will be installed.

Product	Status	Progress	Notes
MySQL Server 8.0.25	Ready to Install		
MySQL Workbench 8.0.25	Ready to Install		
MySQL Shell 8.0.25	Ready to Install		
MySQL Router 8.0.25	Ready to Install		
Connector/ODBC 8.0.25	Ready to Install		
Connector/C++ 8.0.25	Ready to Install		
Connector/J 8.0.25	Ready to Install		
Connector/NET 8.0.25	Ready to Install		
Connector/Python 8.0.25	Ready to Install		
MySQL Documentation 8.0.25	Ready to Install		
Samples and Examples 8.0.25	Ready to Install		

Click [Execute] to install the following packa

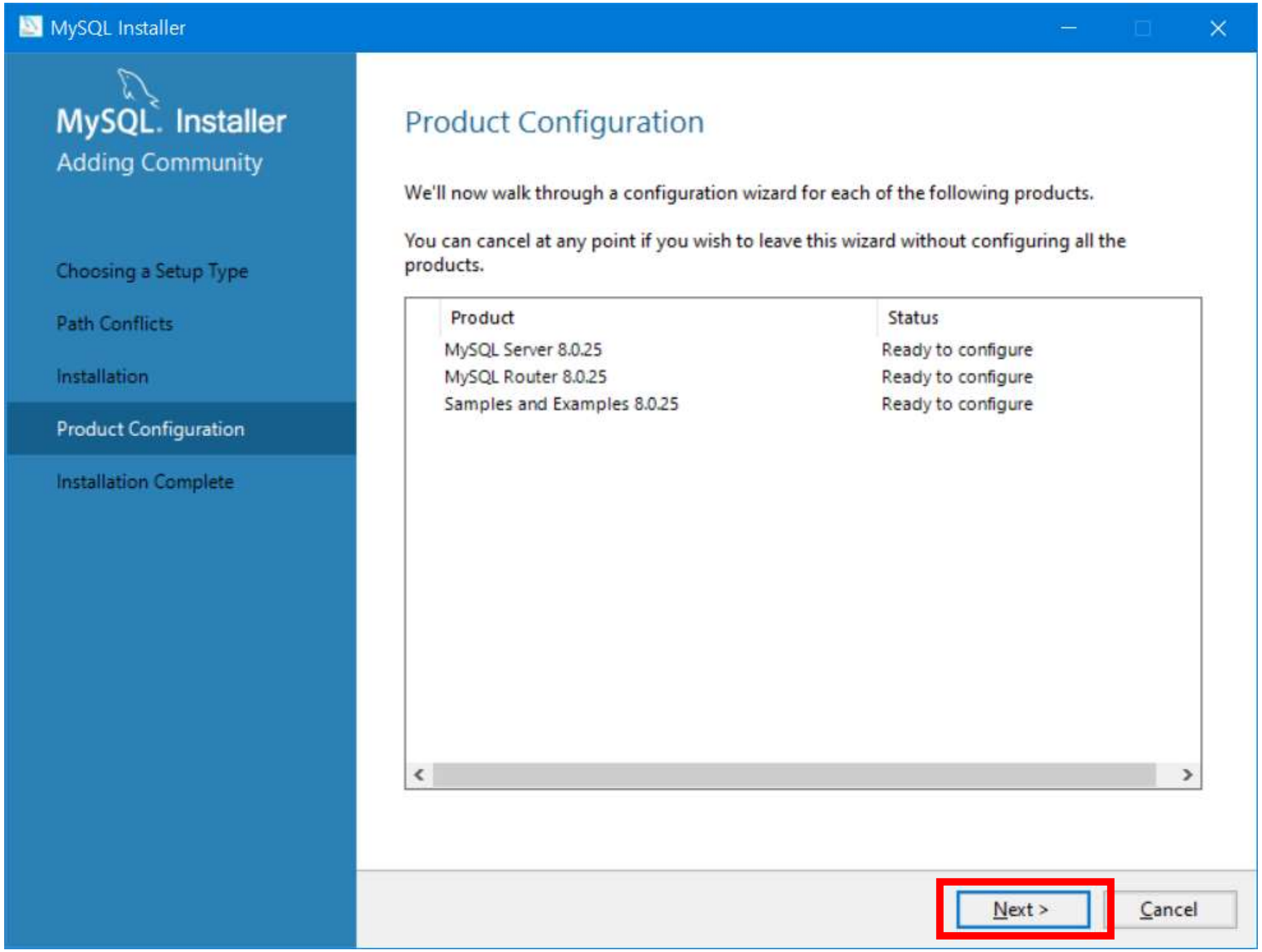
Install the products



Installation

The following products will be installed.

Product	Status	Progress	Notes
MySQL Server 8.0.25	Complete		
MySQL Workbench 8.0.25	Complete		
MySQL Shell 8.0.25	Complete		
MySQL Router 8.0.25	Complete		
Connector/ODBC 8.0.25	Complete		
Connector/C++ 8.0.25	Complete		
Connector/J 8.0.25	Complete		
Connector/NET 8.0.25	Complete		
Connector/Python 8.0.25	Complete		
MySQL Documentation 8.0.25	Complete		
Samples and Examples 8.0.25	Complete		

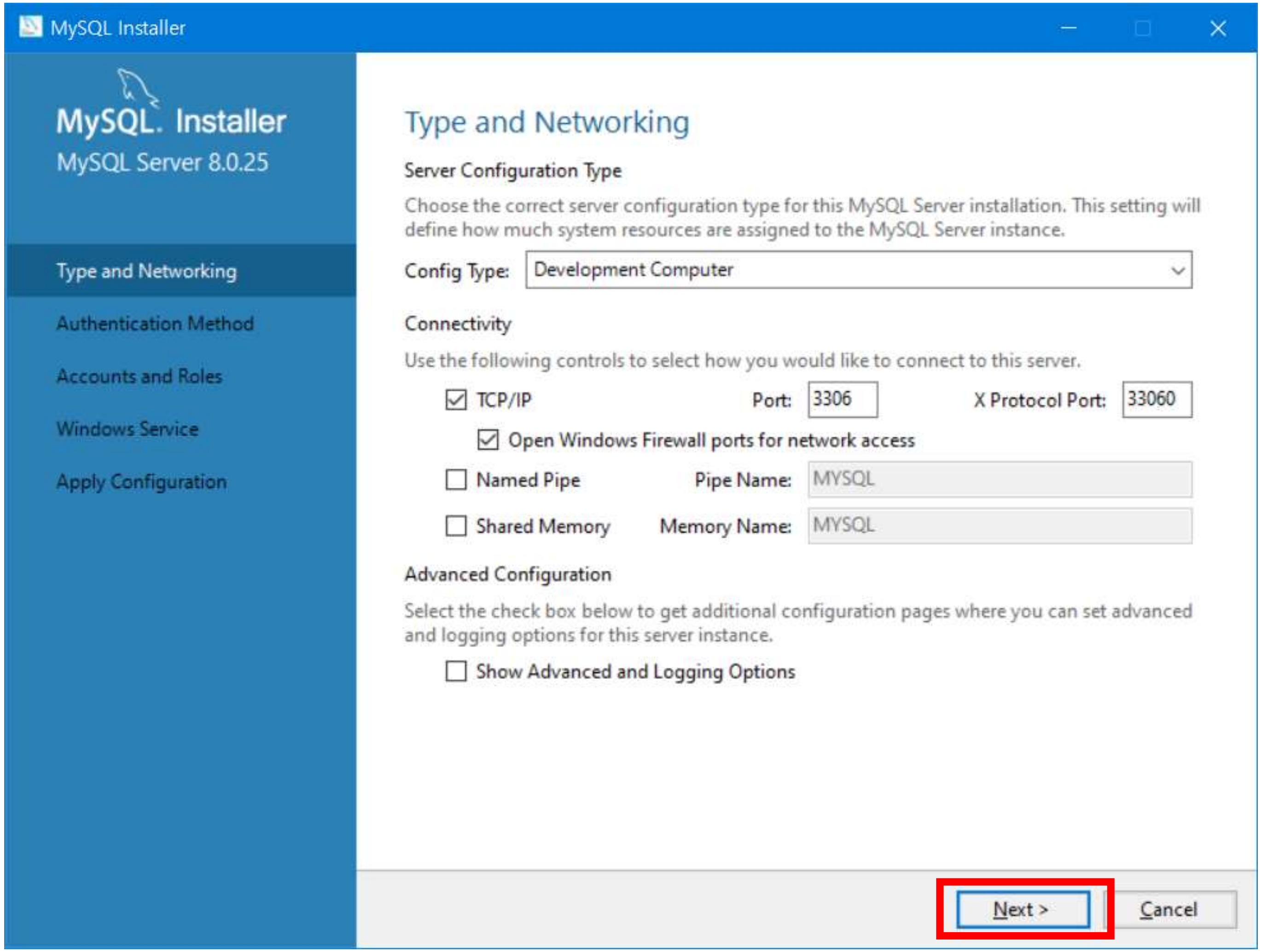


Product Configuration

We'll now walk through a configuration wizard for each of the following products.

You can cancel at any point if you wish to leave this wizard without configuring all the products.

Product	Status
MySQL Server 8.0.25	Ready to configure
MySQL Router 8.0.25	Ready to configure
Samples and Examples 8.0.25	Ready to configure



MySQL Installer

MySQL. Installer
MySQL Server 8.0.25

Type and Networking

Server Configuration Type

Choose the correct server configuration type for this MySQL Server installation. This setting will define how much system resources are assigned to the MySQL Server instance.

Config Type: Development Computer

Connectivity

Use the following controls to select how you would like to connect to this server.

☑ TCP/IP Port: 3306 X Protocol Port: 33060

☑ Open Windows Firewall ports for network access

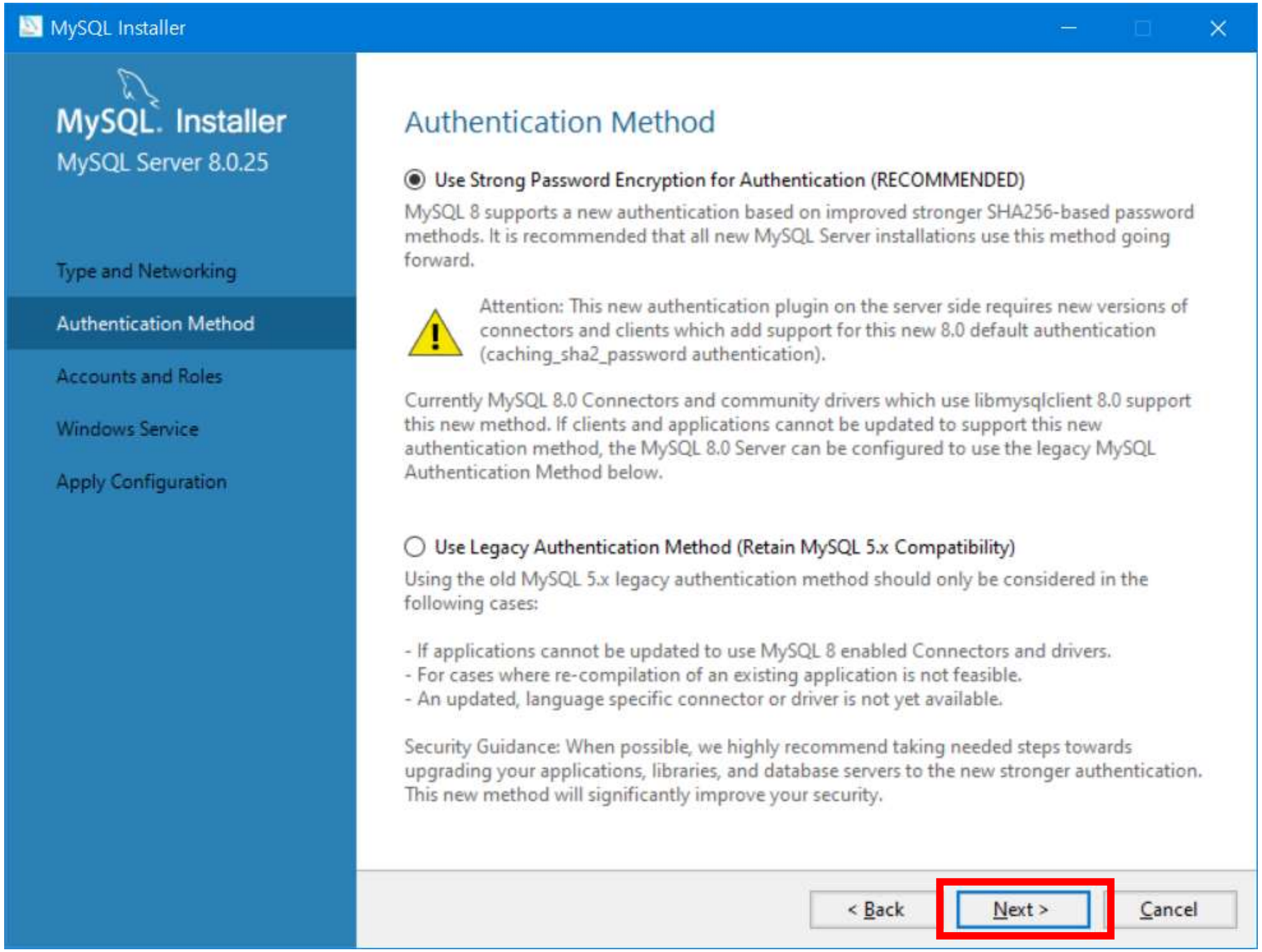
□ Named Pipe Pipe Name: MYSQL

□ Shared Memory Memory Name: MYSQL

Advanced Configuration

Select the check box below to get additional configuration pages where you can set advanced and logging options for this server instance.

□ Show Advanced and Logging Options



Authentication Method

● Use Strong Password Encryption for Authentication (RECOMMENDED)

MySQL 8 supports a new authentication based on improved stronger SHA256-based password methods. It is recommended that all new MySQL Server installations use this method going forward.

Attention: This new authentication plugin on the server side requires new versions of connectors and clients which add support for this new 8.0 default authentication (caching_sha2_password authentication).

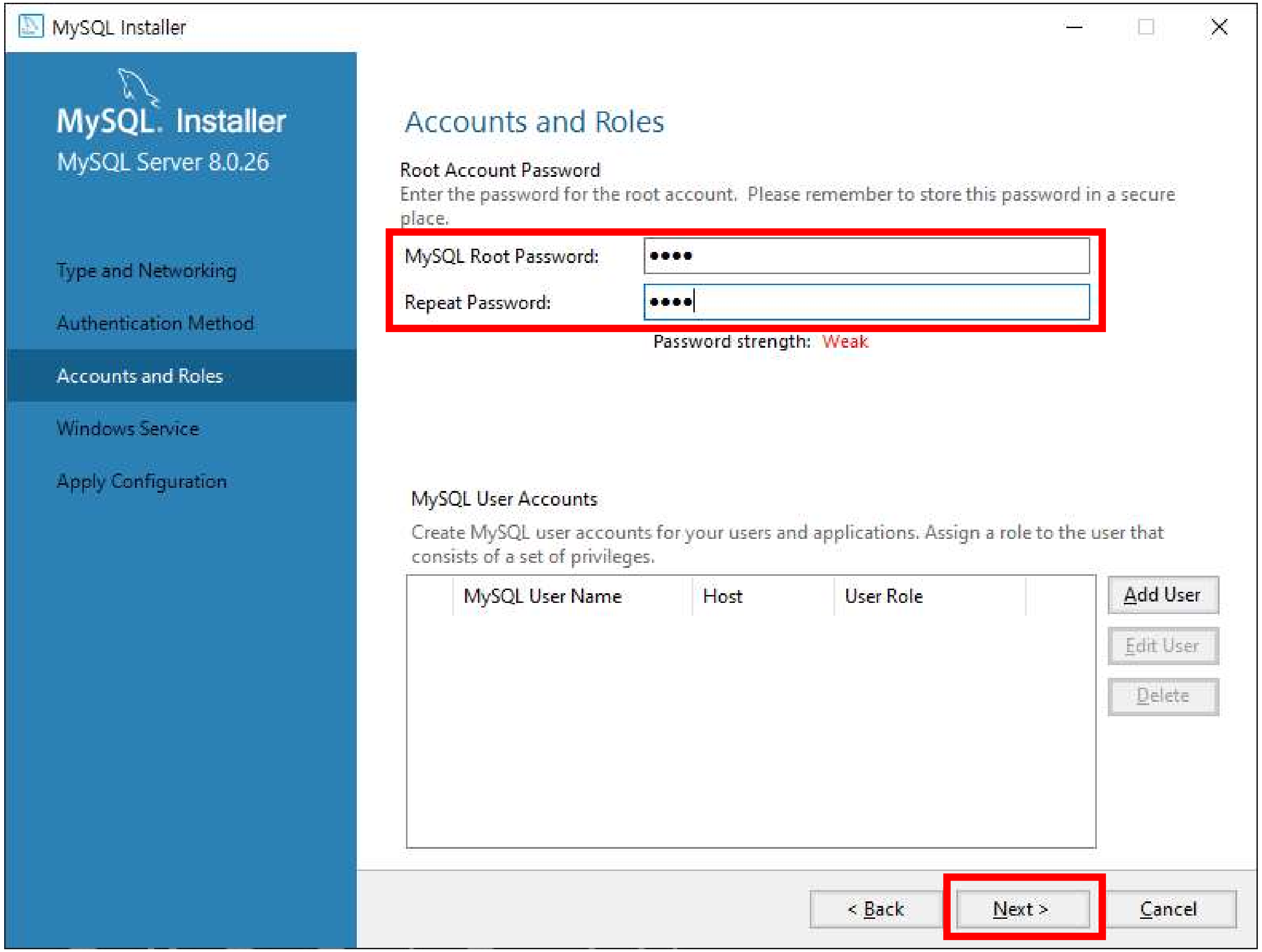
Currently MySQL 8.0 Connectors and community drivers which use libmysqlclient 8.0 support this new method. If clients and applications cannot be updated to support this new authentication method, the MySQL 8.0 Server can be configured to use the legacy MySQL Authentication Method below.

○ Use Legacy Authentication Method (Retain MySQL 5.x Compatibility)

Using the old MySQL 5.x legacy authentication method should only be considered in the following cases:

- If applications cannot be updated to use MySQL 8 enabled Connectors and drivers.
- For cases where re-compilation of an existing application is not feasible.
- An updated, language specific connector or driver is not yet available.

Security Guidance: When possible, we highly recommend taking needed steps towards upgrading your applications, libraries, and database servers to the new stronger authentication. This new method will significantly improve your security.



Accounts and Roles

Root Account Password

Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password: ●●●●

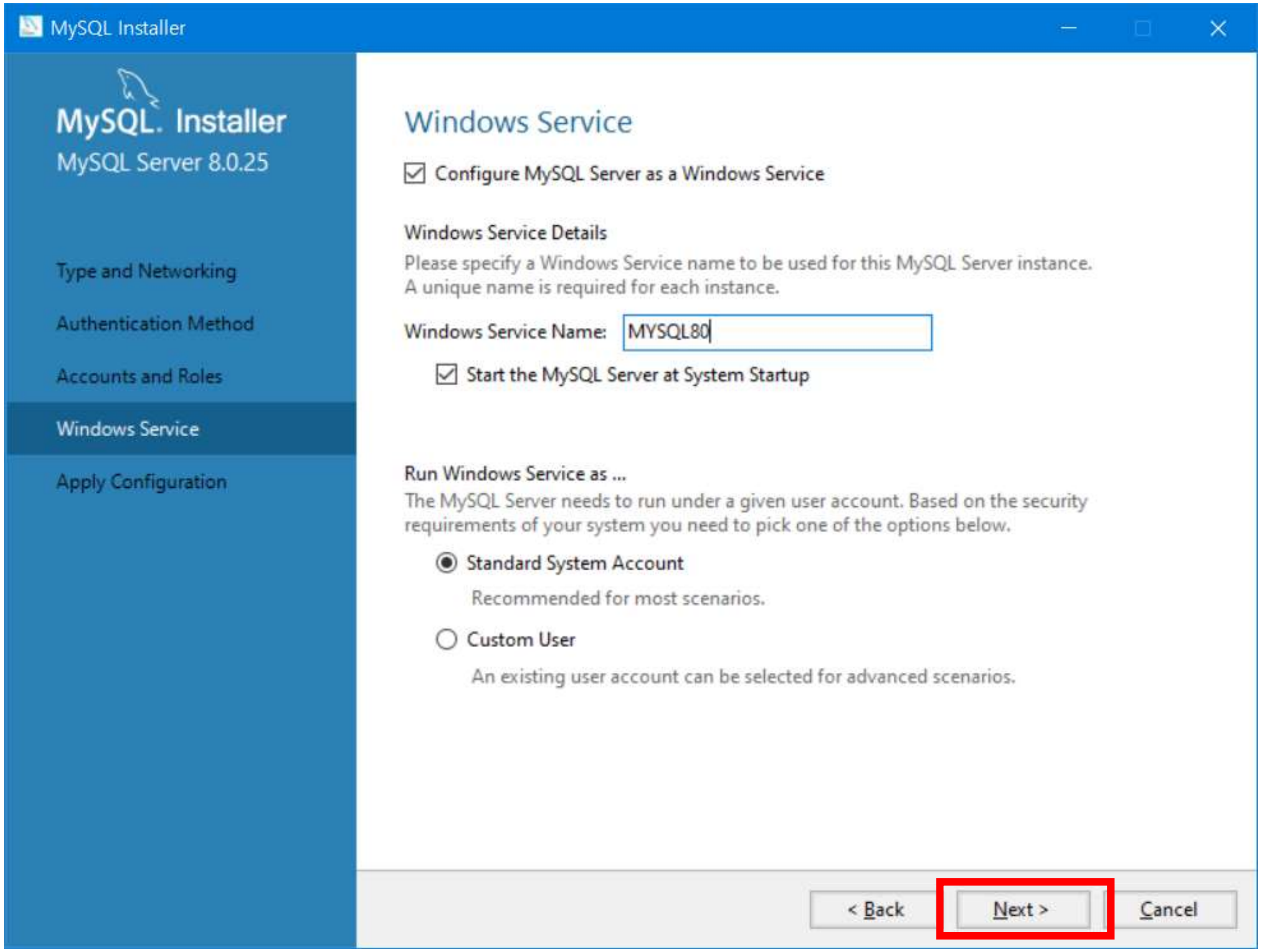
Repeat Password: ●●●●

Password strength: Weak

MySQL User Accounts

Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL User Name	Host	User Role



Windows Service

☑ Configure MySQL Server as a Windows Service

Windows Service Details

Please specify a Windows Service name to be used for this MySQL Server instance.
A unique name is required for each instance.

Windows Service Name: MYSQL80

☑ Start the MySQL Server at System Startup

Run Windows Service as ...

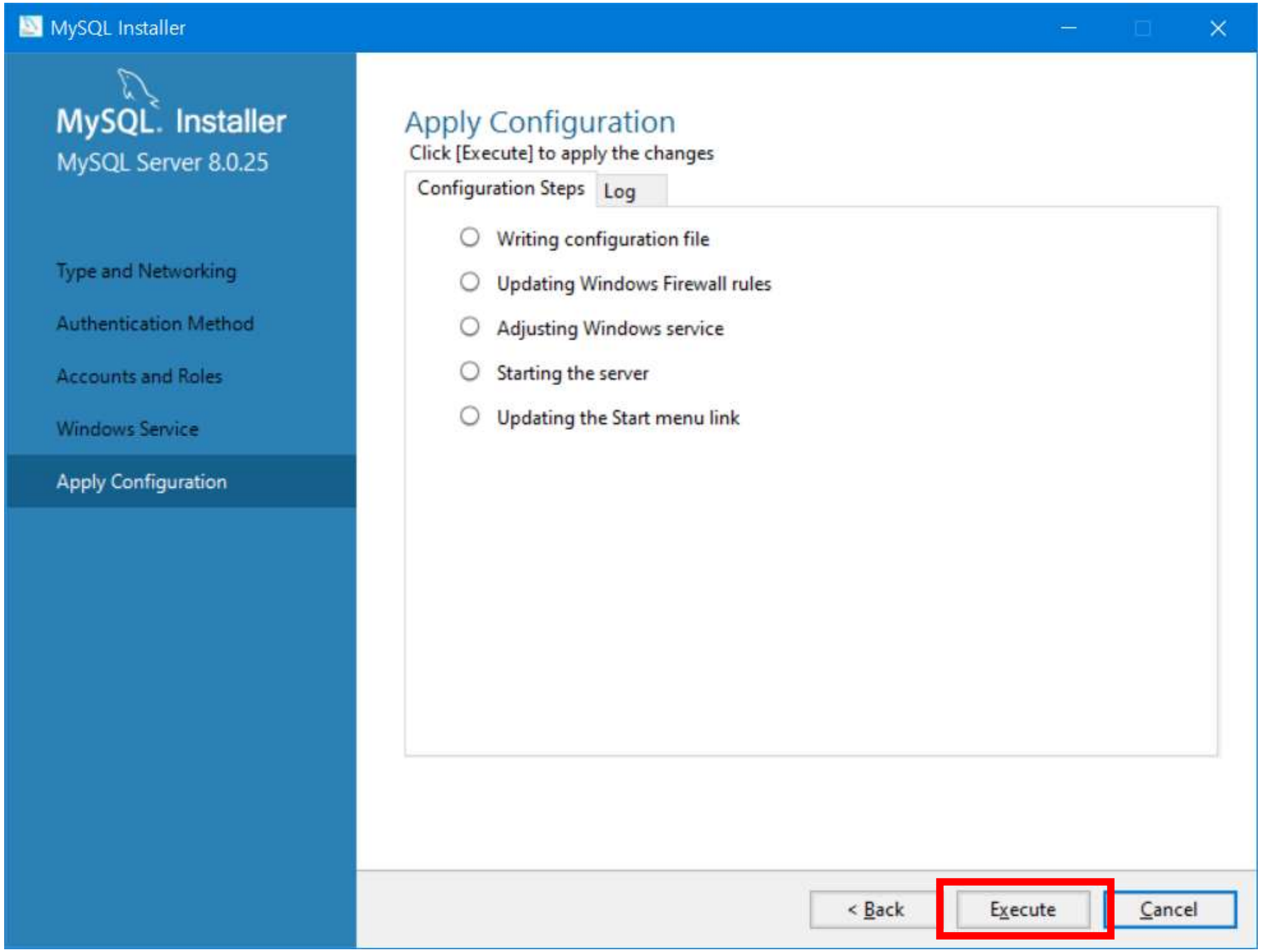
The MySQL Server needs to run under a given user account. Based on the security requirements of your system you need to pick one of the options below.

● Standard System Account

Recommended for most scenarios.

○ Custom User

An existing user account can be selected for advanced scenarios.



MySQL Installer

MySQL. Installer

MySQL Server 8.0.25

Apply Configuration

Click [Execute] to apply the changes

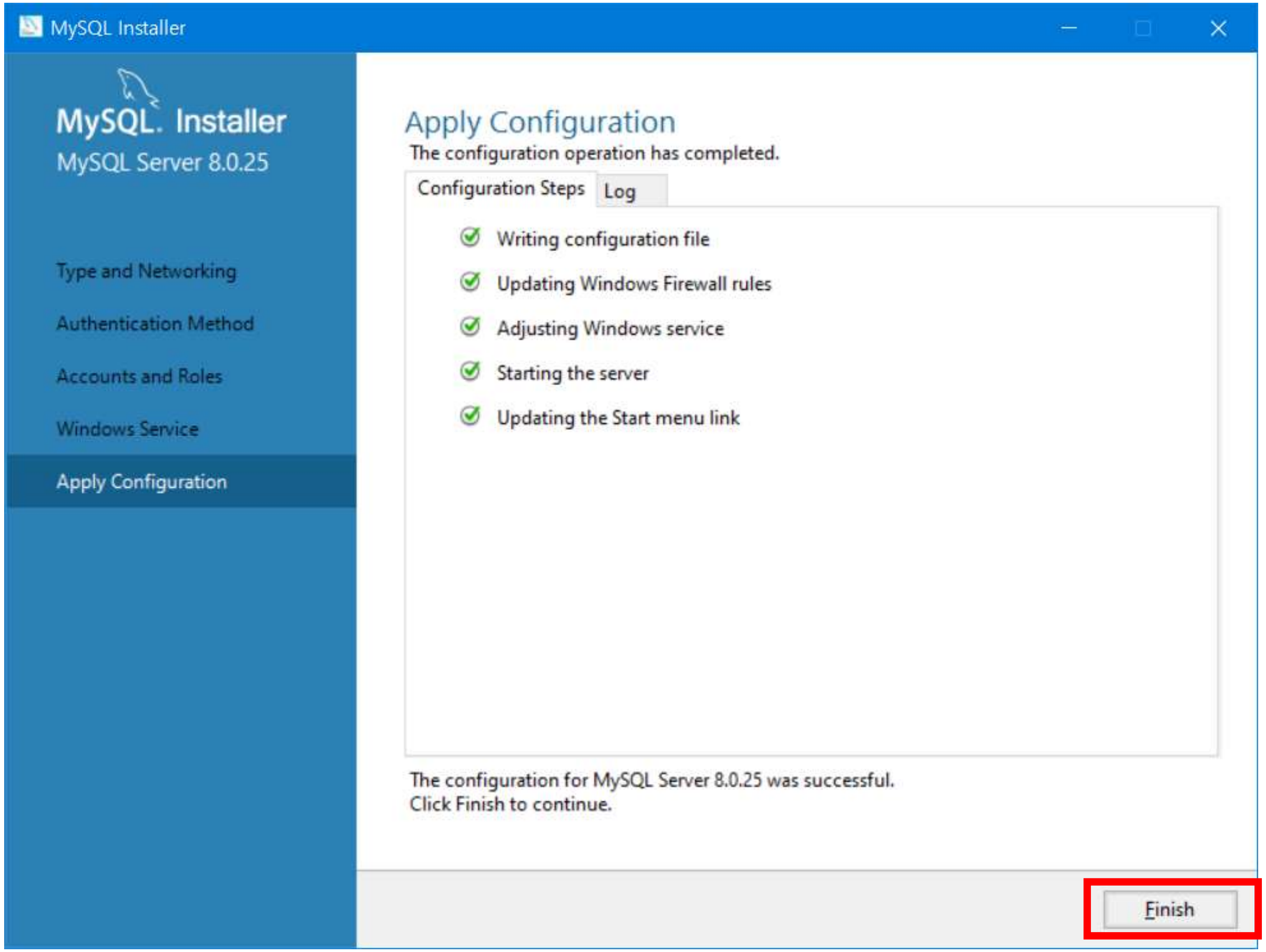
○ Writing configuration file

○ Updating Windows Firewall rules

○ Adjusting Windows service

○ Starting the server

○ Updating the Start menu link

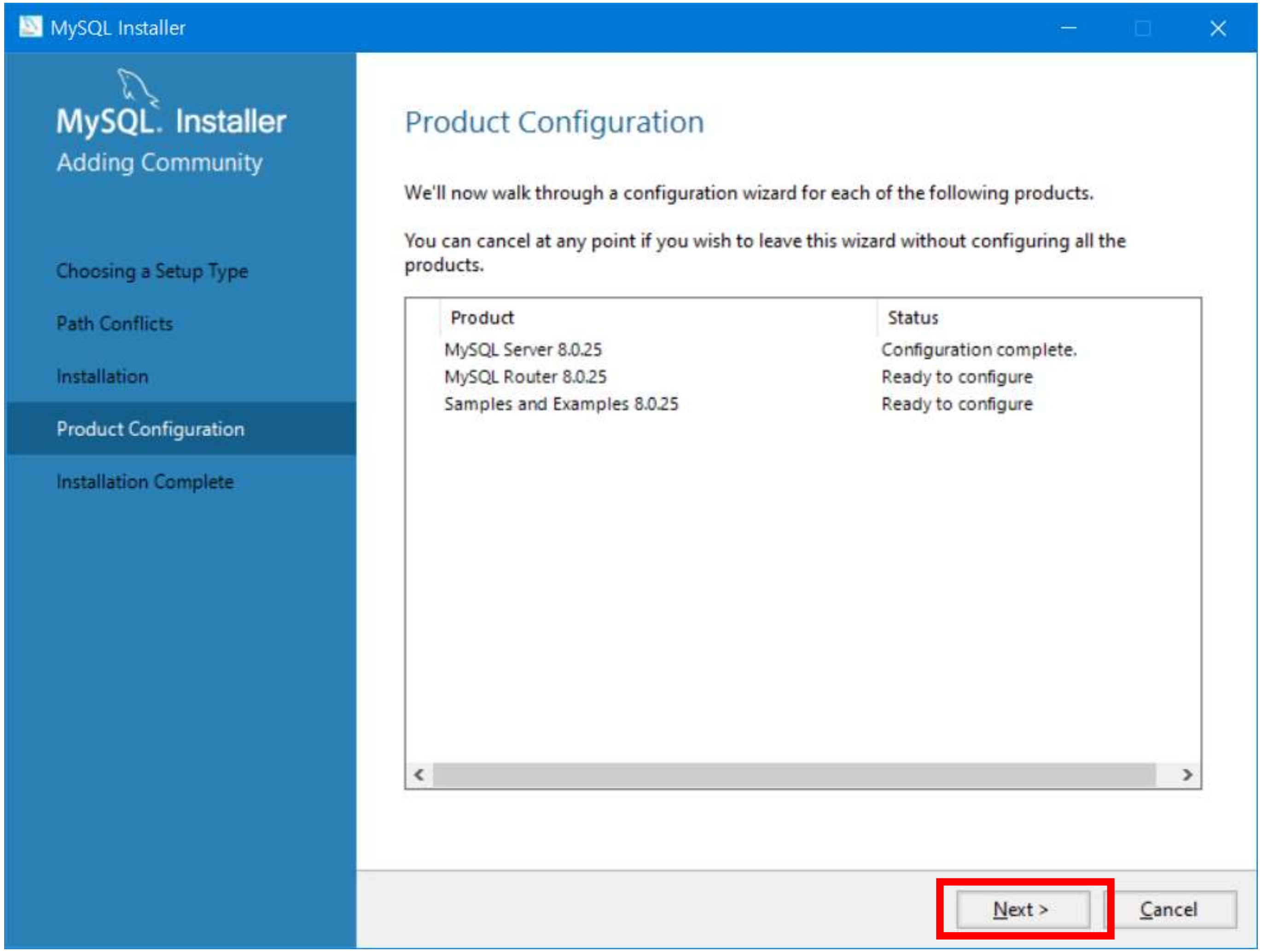


Apply Configuration

The configuration operation has completed.

- ✓ Writing configuration file
- ✓ Updating Windows Firewall rules
- ✓ Adjusting Windows service
- ✓ Starting the server
- ✓ Updating the Start menu link

The configuration for MySQL Server 8.0.25 was successful.
Click Finish to continue.

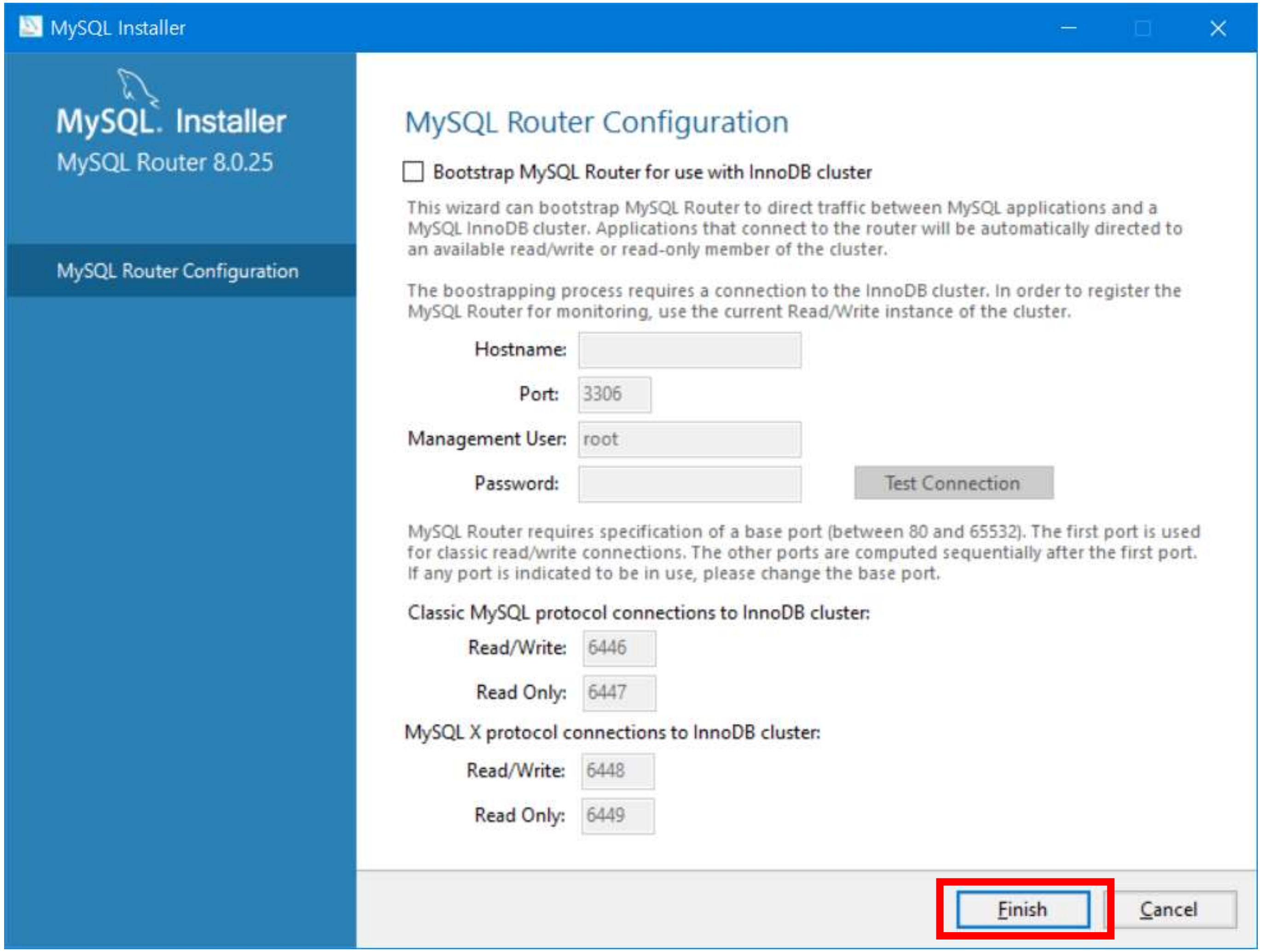


Product Configuration

We'll now walk through a configuration wizard for each of the following products.

You can cancel at any point if you wish to leave this wizard without configuring all the products.

Product	Status
MySQL Server 8.0.25	Configuration complete.
MySQL Router 8.0.25	Ready to configure
Samples and Examples 8.0.25	Ready to configure



MySQL Router Configuration

□ Bootstrap MySQL Router for use with InnoDB cluster

This wizard can bootstrap MySQL Router to direct traffic between MySQL applications and a MySQL InnoDB cluster. Applications that connect to the router will be automatically directed to an available read/write or read-only member of the cluster.

The boostrapping process requires a connection to the InnoDB cluster. In order to register the MySQL Router for monitoring, use the current Read/Write instance of the cluster.

Hostname:

Port: 3306

Management User: root

Password:

MySQL Router requires specification of a base port (between 80 and 65532). The first port is used for classic read/write connections. The other ports are computed sequentially after the first port. If any port is indicated to be in use, please change the base port.

Classic MySQL protocol connections to InnoDB cluster:

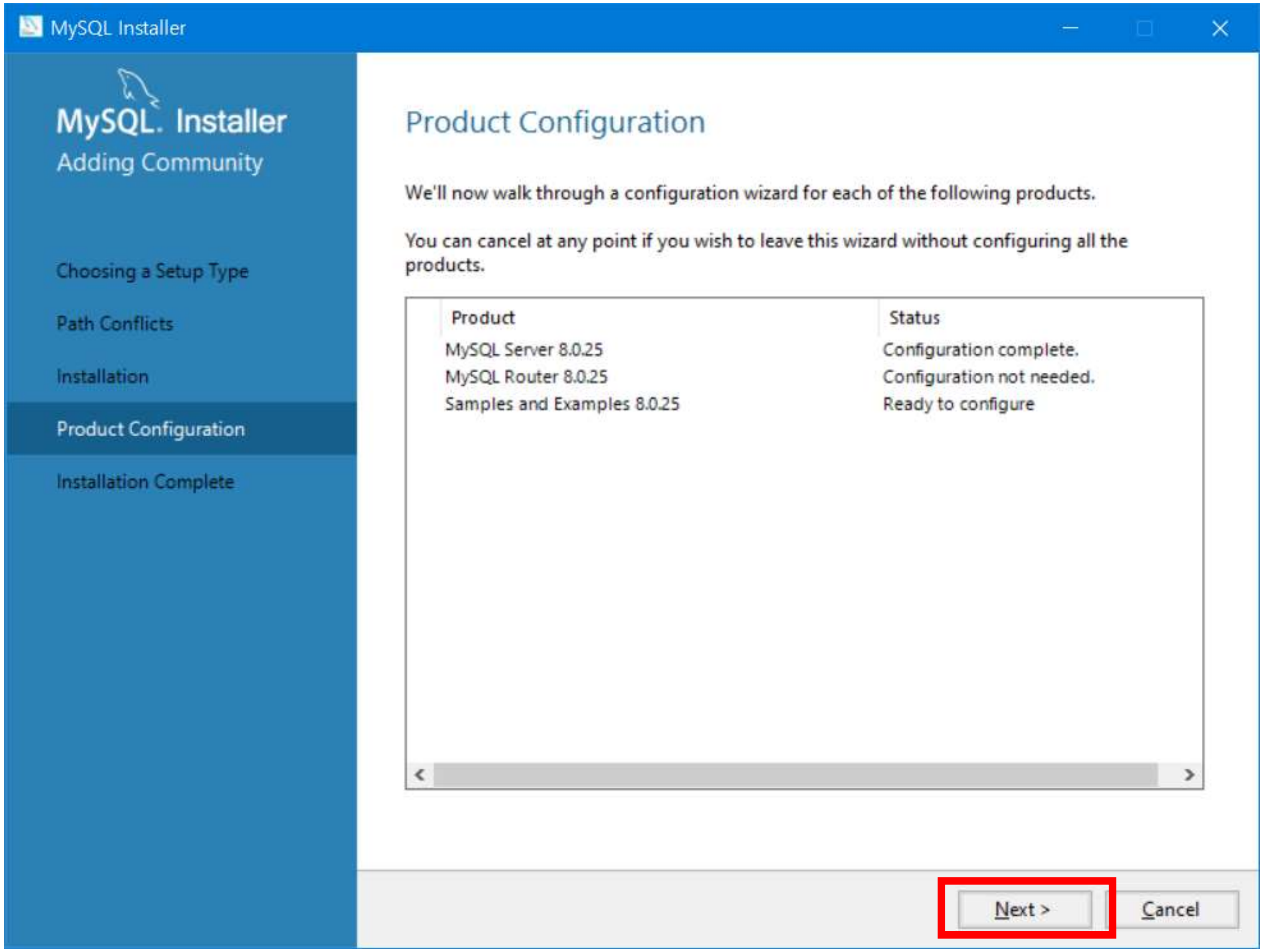
Read/Write: 6446

Read Only: 6447

MySQL X protocol connections to InnoDB cluster:

Read/Write: 6448

Read Only: 6449

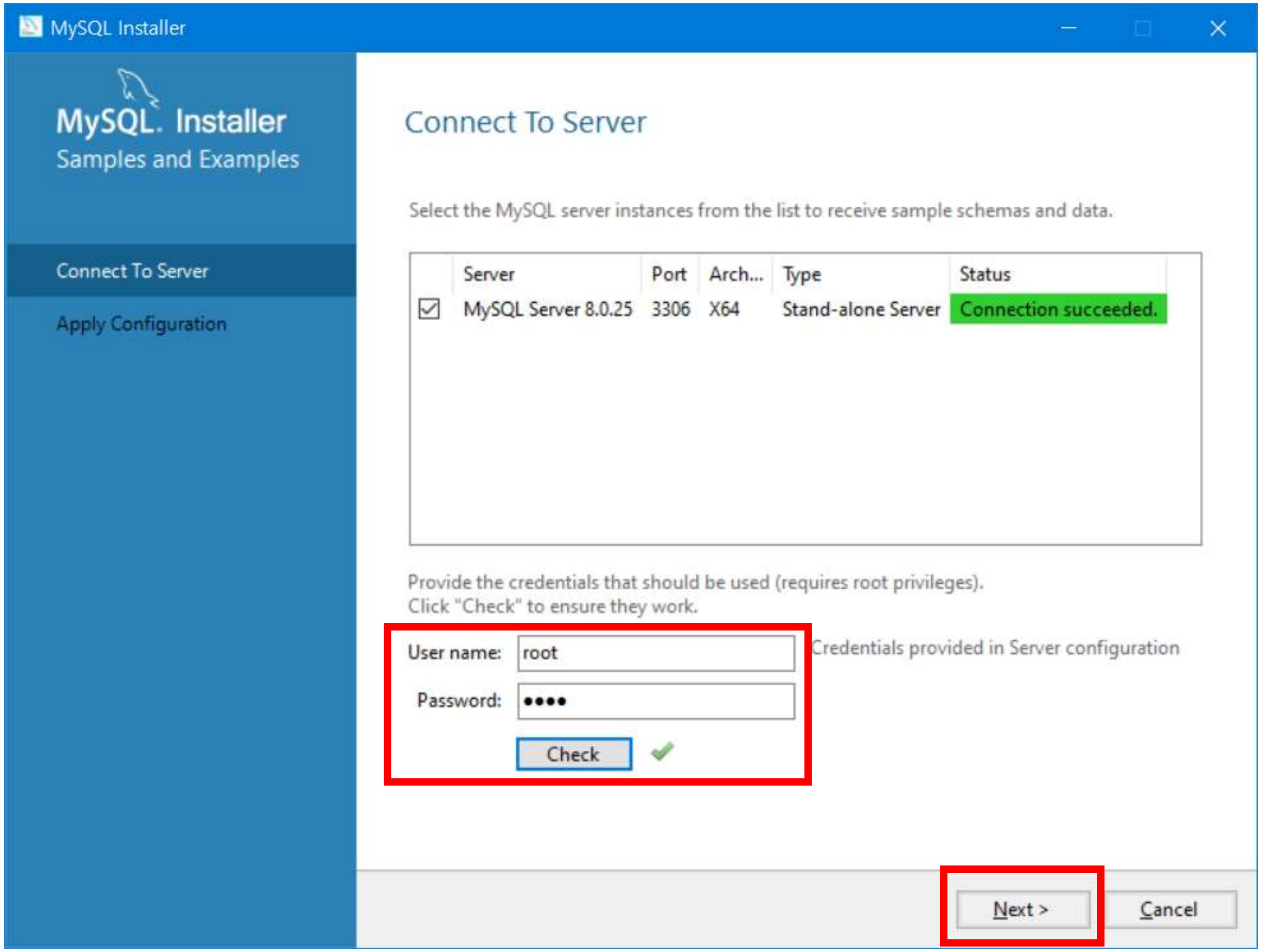


Product Configuration

We'll now walk through a configuration wizard for each of the following products.

You can cancel at any point if you wish to leave this wizard without configuring all the products.

Product	Status
MySQL Server 8.0.25	Configuration complete.
MySQL Router 8.0.25	Configuration not needed.
Samples and Examples 8.0.25	Ready to configure



Connect To Server

Select the MySQL server instances from the list to receive sample schemas and data.

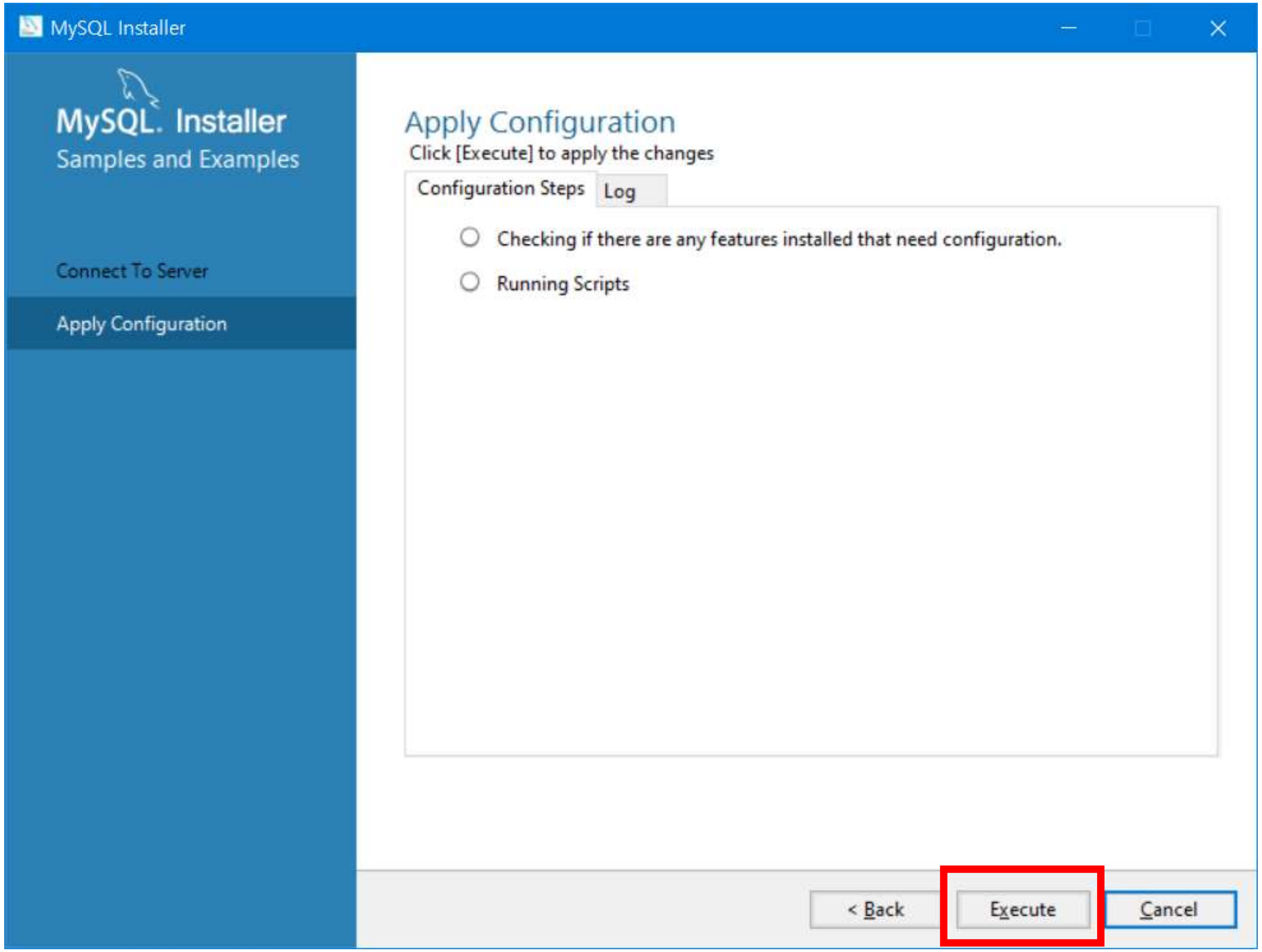
	Server	Port	Arch...	Type	Status
☑	MySQL Server 8.0.25	3306	X64	Stand-alone Server	Connection succeeded.

Provide the credentials that should be used (requires root privileges).
Click "Check" to ensure they work.

User name: root

Password: ●●●●

Credentials provided in Server configuration

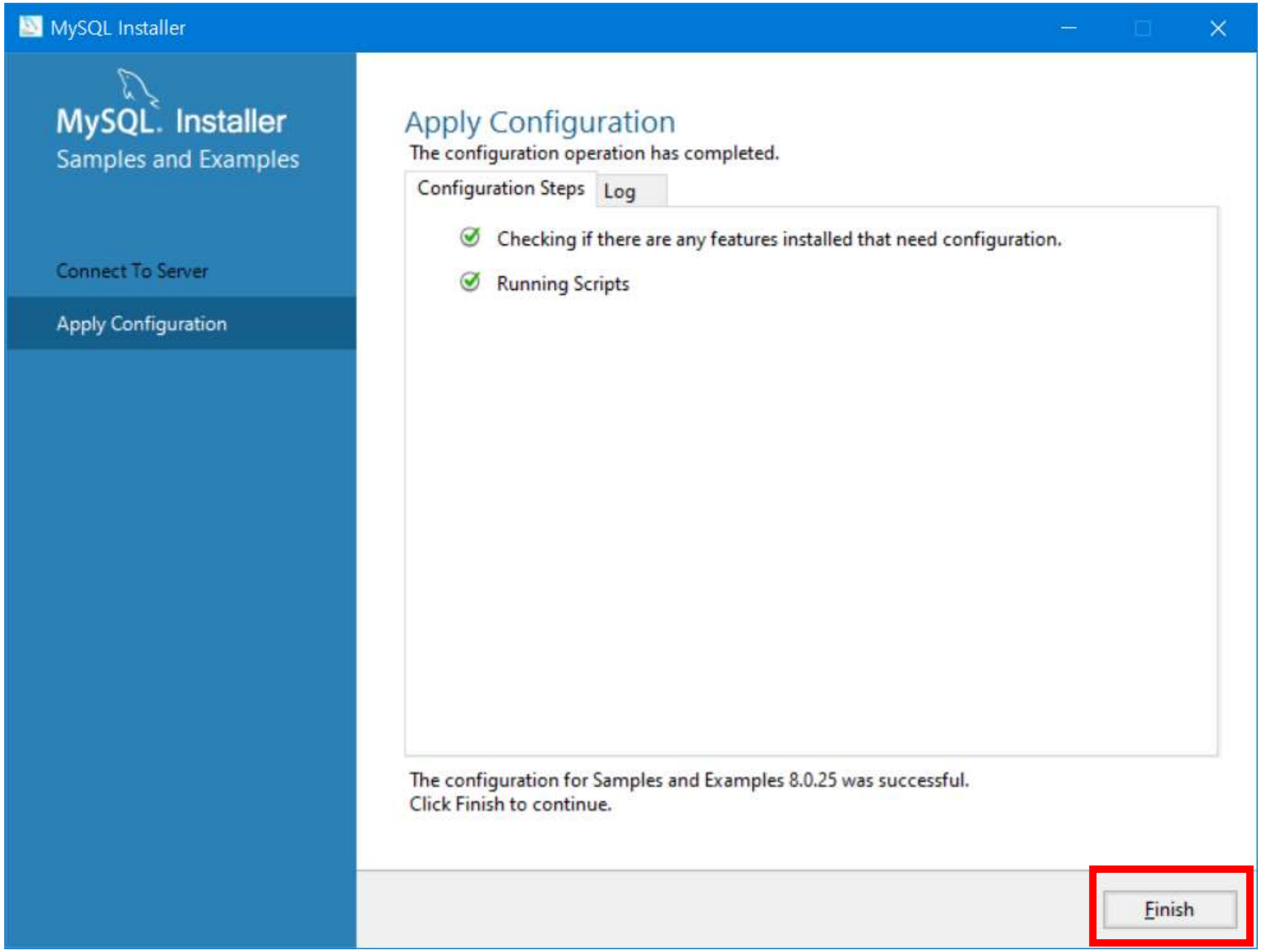


Apply Configuration

Click [Execute] to apply the changes

○ Checking if there are any features installed that need configuration.

○ Running Scripts



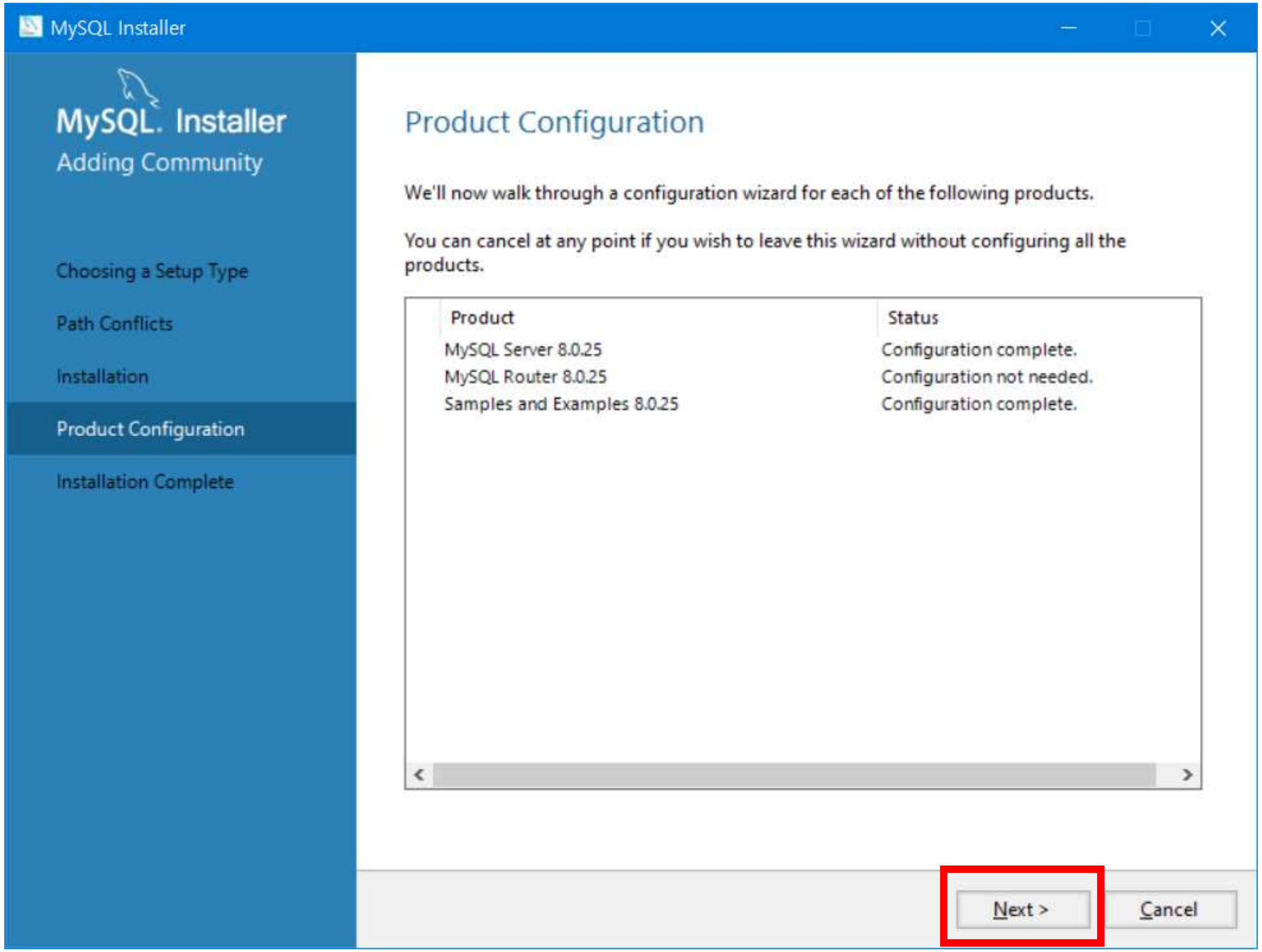
Apply Configuration

The configuration operation has completed.

✓ Checking if there are any features installed that need configuration.

✓ Running Scripts

The configuration for Samples and Examples 8.0.25 was successful.
Click Finish to continue.

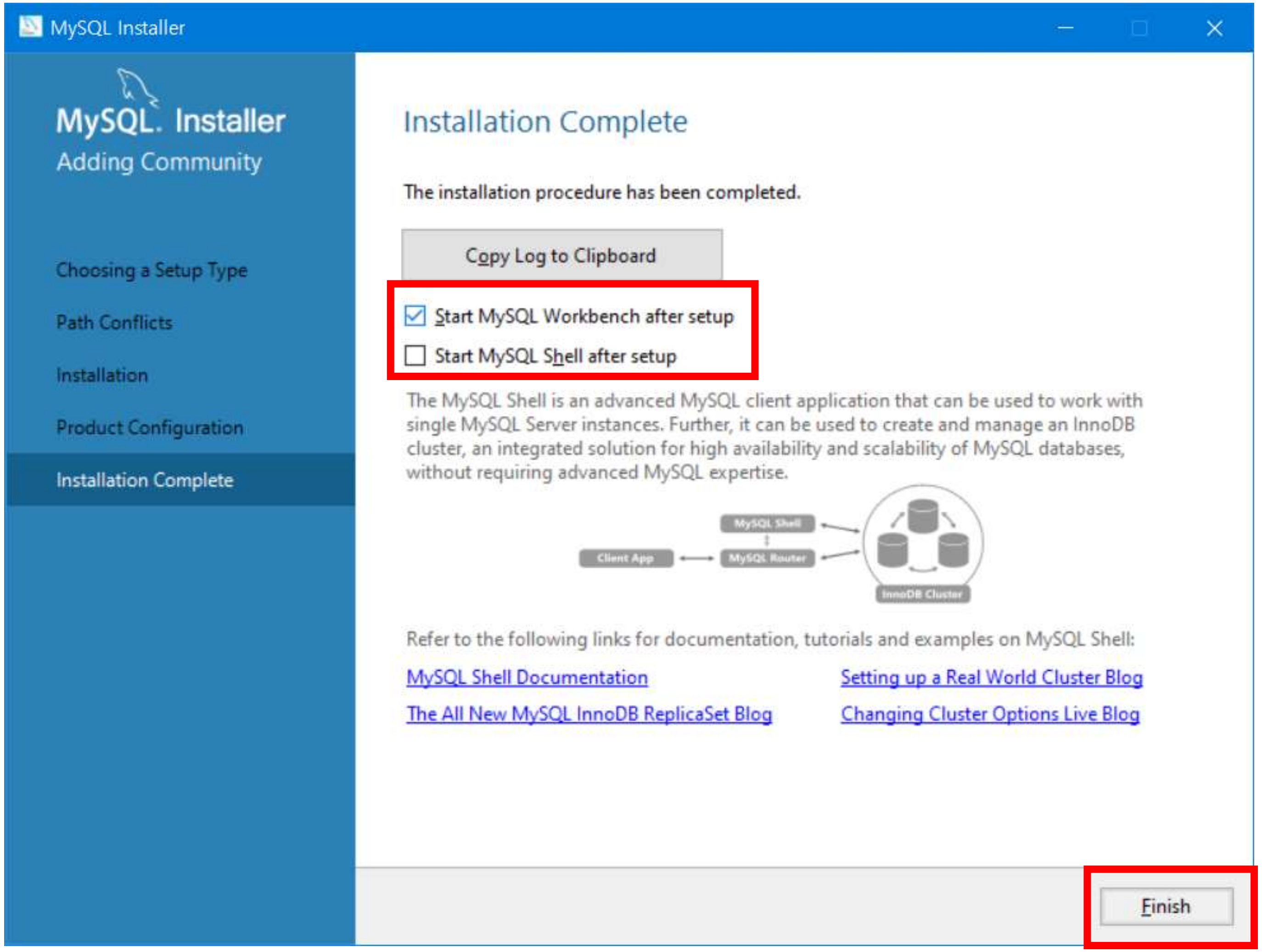


Product Configuration

We'll now walk through a configuration wizard for each of the following products.

You can cancel at any point if you wish to leave this wizard without configuring all the products.

Product	Status
MySQL Server 8.0.25	Configuration complete.
MySQL Router 8.0.25	Configuration not needed.
Samples and Examples 8.0.25	Configuration complete.



MySQL Installer

MySQL. Installer
Adding Community

- Choosing a Setup Type
- Path Conflicts
- Installation
- Product Configuration
- Installation Complete

Installation Complete

The installation procedure has been completed.

Copy Log to Clipboard

☑ Start MySQL Workbench after setup

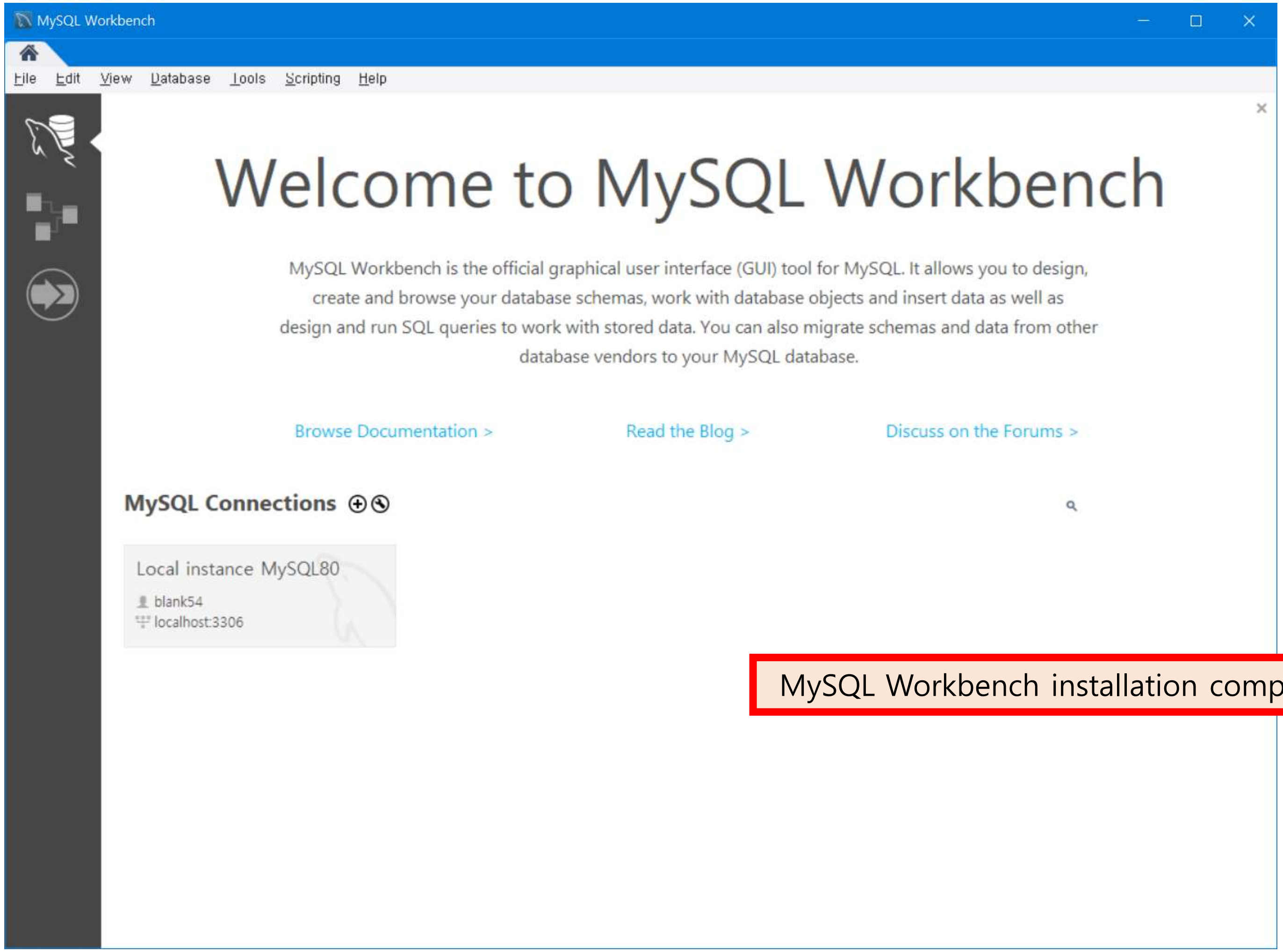
□ Start MySQL Shell after setup

The MySQL Shell is an advanced MySQL client application that can be used to work with single MySQL Server instances. Further, it can be used to create and manage an InnoDB cluster, an integrated solution for high availability and scalability of MySQL databases, without requiring advanced MySQL expertise.

Refer to the following links for documentation, tutorials and examples on MySQL Shell:

MySQL Shell Documentation
Setting up a Real World Cluster Blog
The All New MySQL InnoDB ReplicaSet Blog
Changing Cluster Options Live Blog

Finish



Welcome to MySQL Workbench

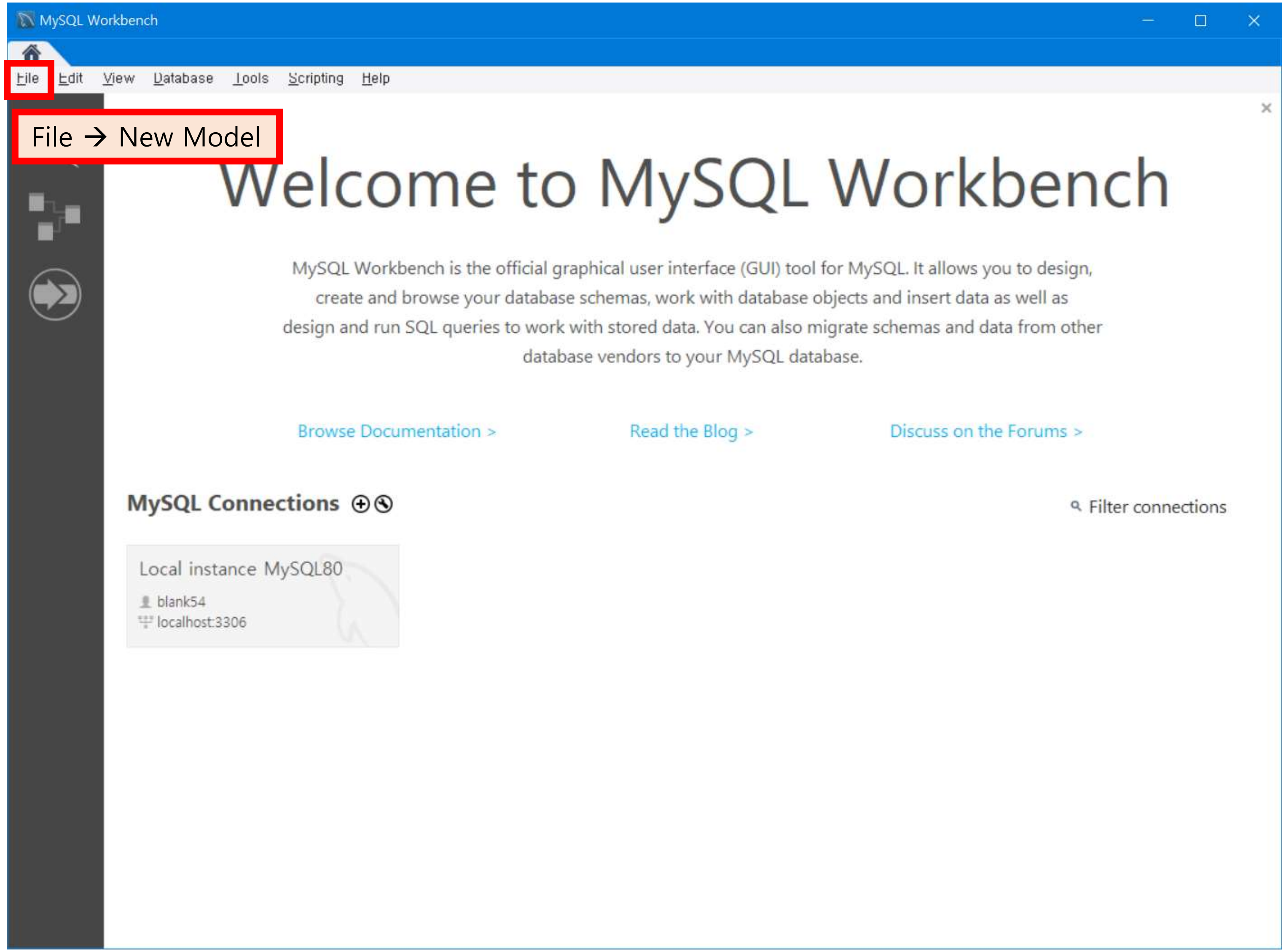
MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

MySQL Connections ⊕ ⊗

Local instance MySQL80
blank54
localhost:3306

MySQL Workbench installation complete

3 Create Model



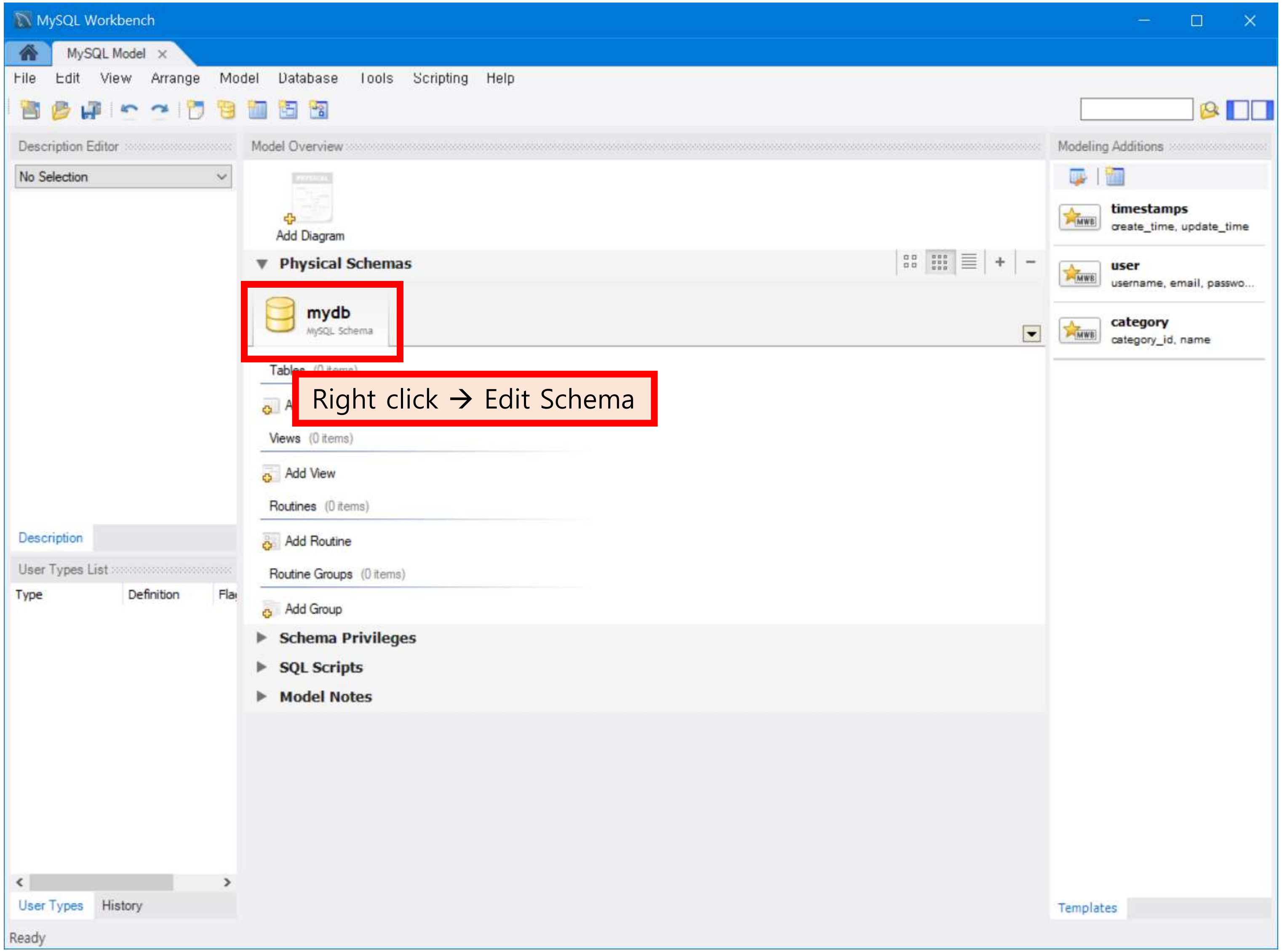
File → New Model

Welcome to MySQL Workbench

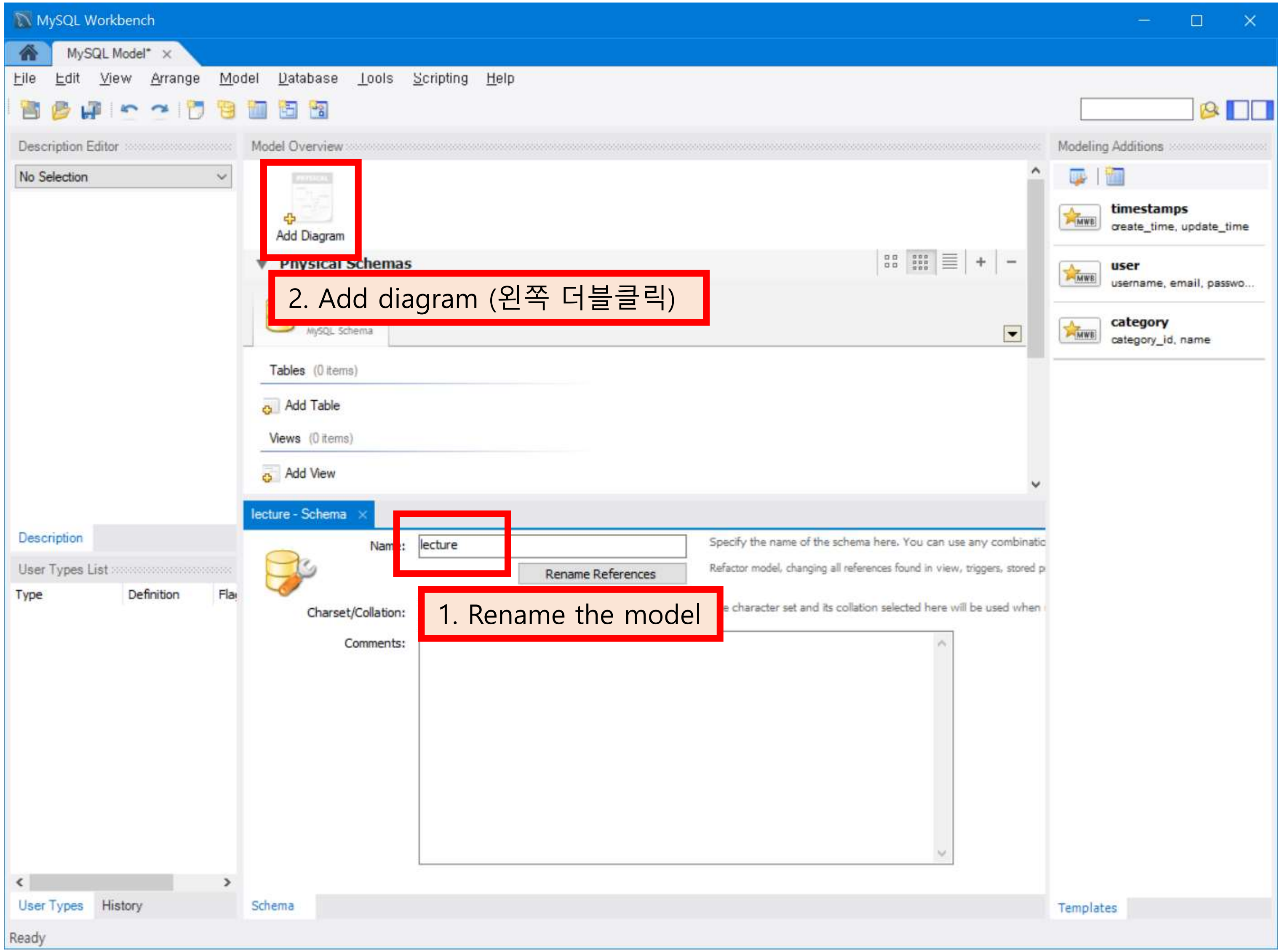
MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

MySQL Connections

Local instance MySQL80
blank54
localhost:3306



Right click → Edit Schema



Name: lecture

Rename References

Specify the name of the schema here. You can use any combinatio

Refactor model, changing all references found in view, triggers, stored p

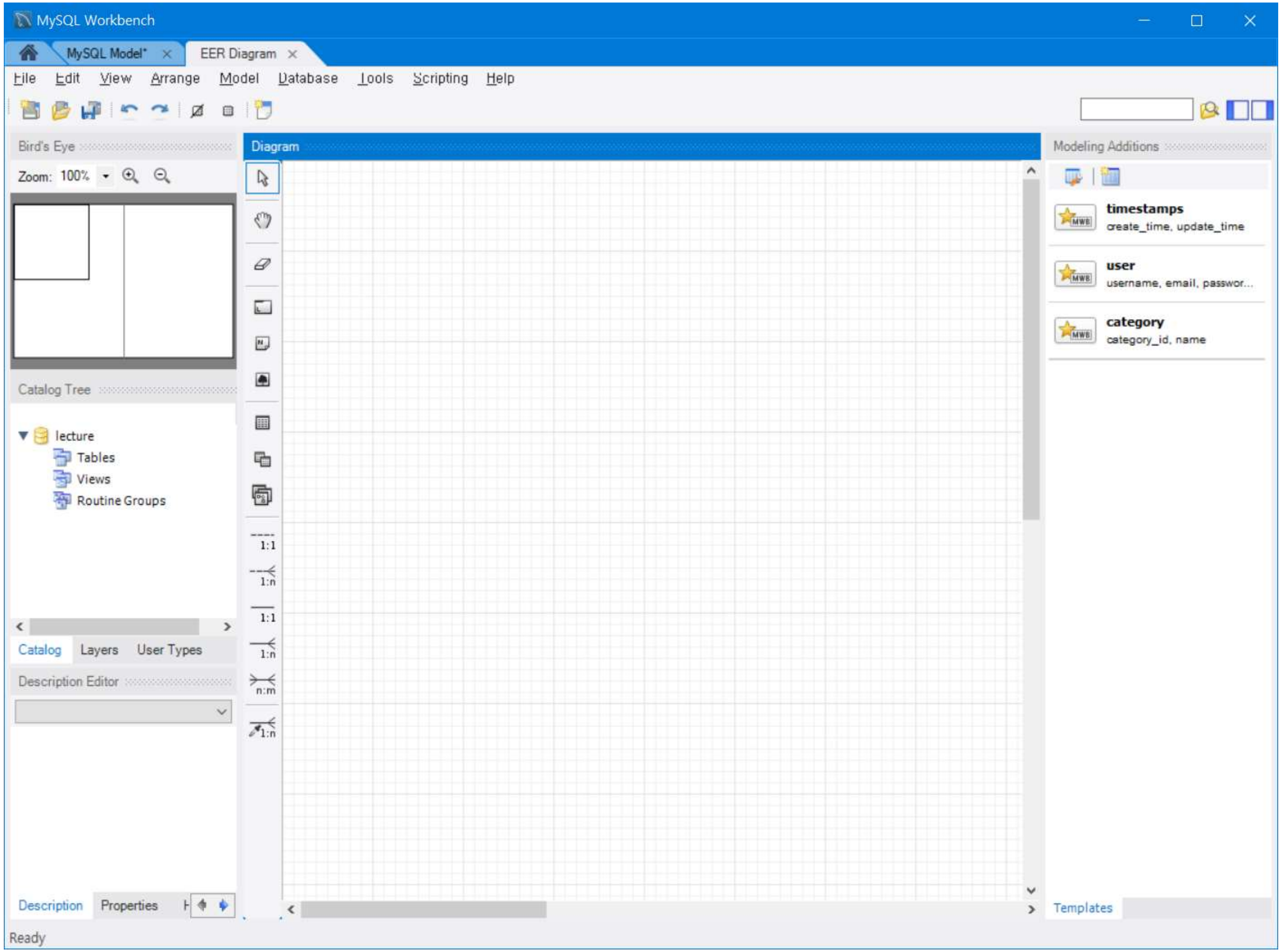
Charset/Collation:

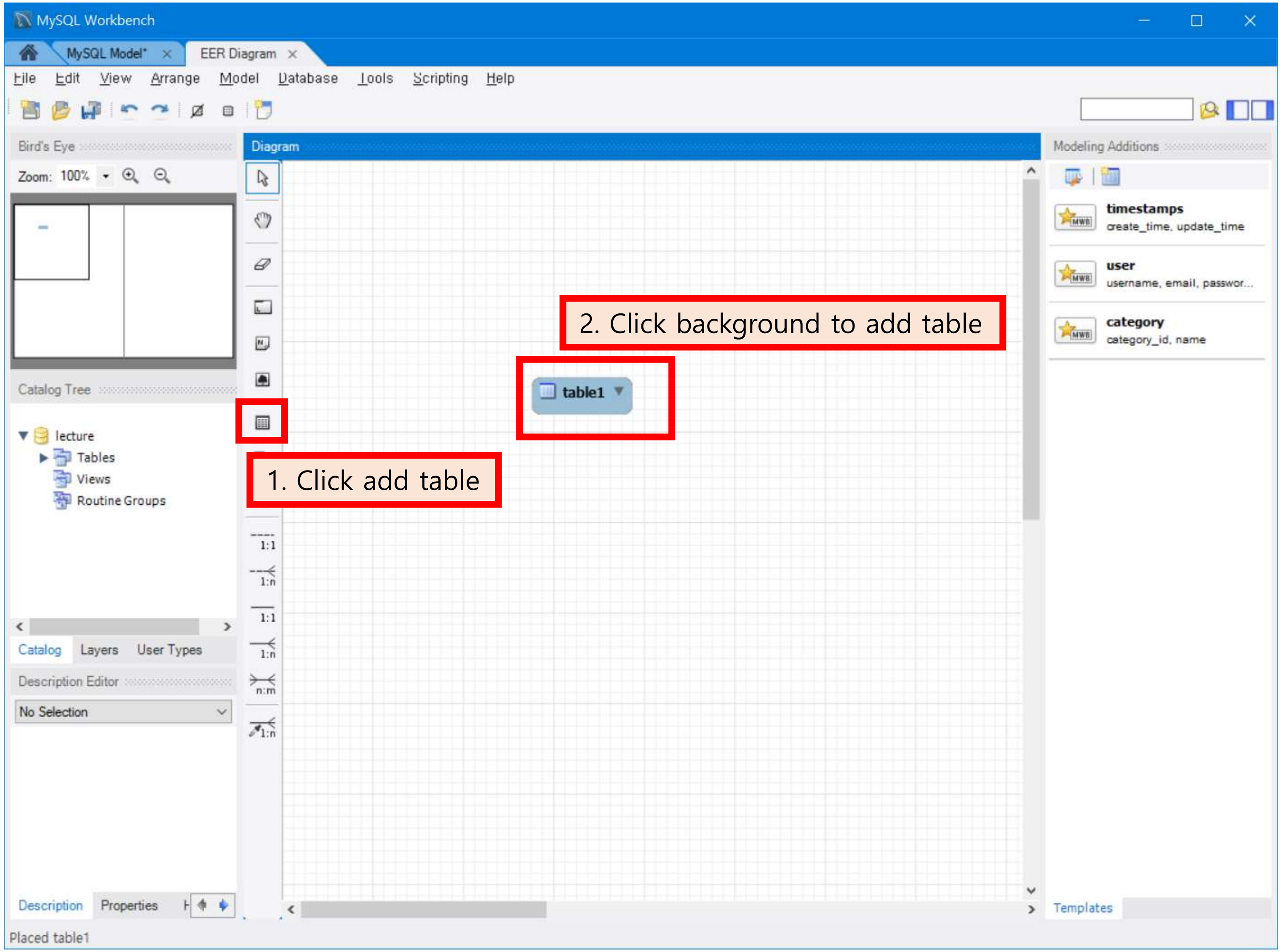
e character set and its collation selected here will be used when

Comments:

2. Add diagram (왼쪽 더블클릭)

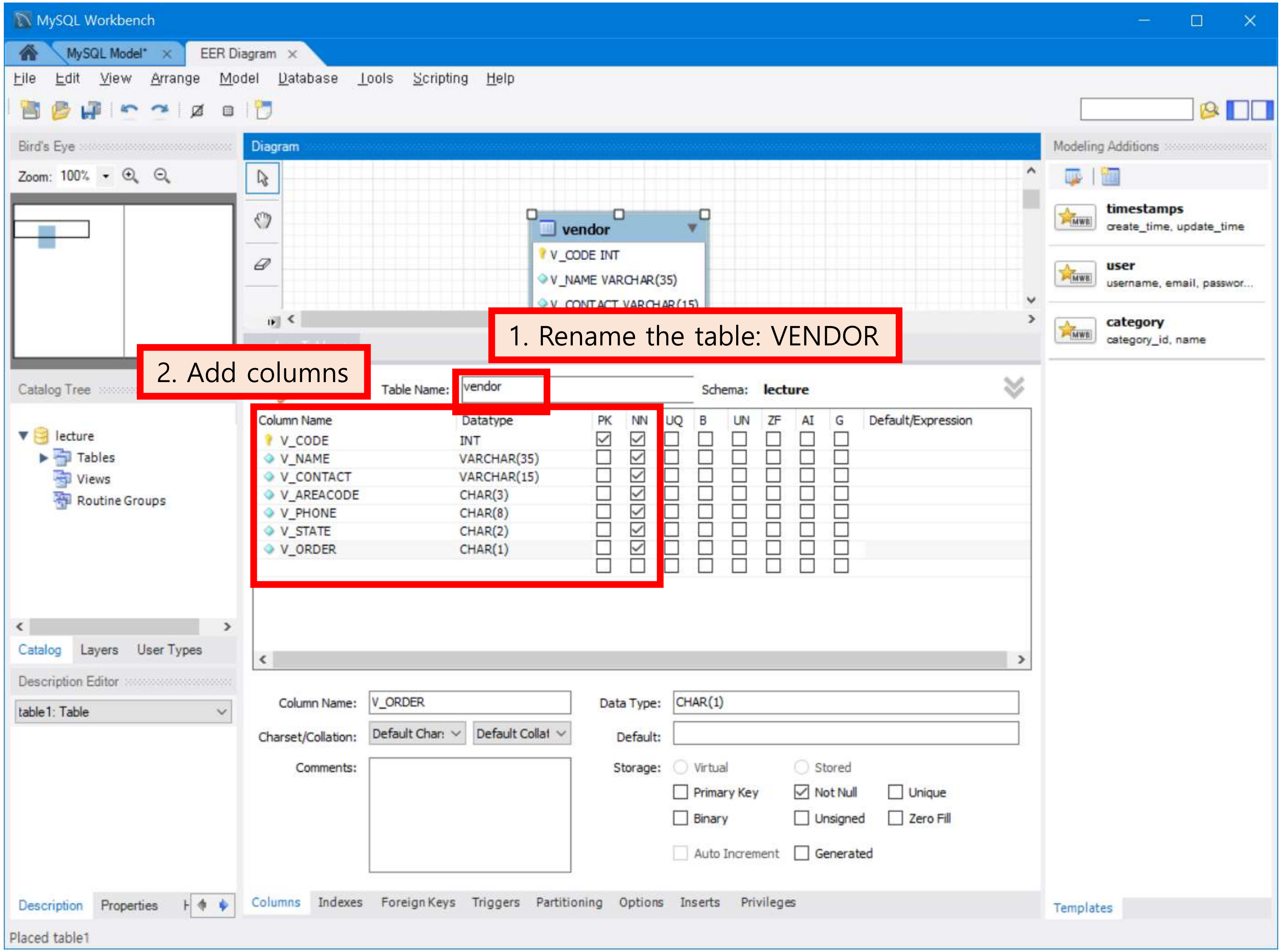
1. Rename the model


MySQL Workbench
MySQL Model*
EER Diagram
File Edit View Arrange Model Database Tools Scripting Help
Bird's Eye
Zoom: 100%
Diagram
Modeling Additions
timestamps
create_time, update_time
user
username, email, passwor...
category
category_id, name
Catalog Tree
lecture
Tables
Views
Routine Groups
1:1
1:n
1:1
1:n
n:m
1:n
Catalog
Layers
User Types
Description Editor
Description
Properties
Templates
Ready



2. Click background to add table

1. Click add table



1. Rename the table: VENDOR

2. Add columns

Table Name: vendor   Schema: lecture

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
V_CODE	INT	☑	☑	□	□	□	□	□	□	
V_NAME	VARCHAR(35)	□	☑	□	□	□	□	□	□	
V_CONTACT	VARCHAR(15)	□	☑	□	□	□	□	□	□	
V_AREACODE	CHAR(3)	□	☑	□	□	□	□	□	□	
V_PHONE	CHAR(8)	□	☑	□	□	□	□	□	□	
V_STATE	CHAR(2)	□	☑	□	□	□	□	□	□	
V_ORDER	CHAR(1)	□	☑	□	□	□	□	□	□	
		□	□	□	□	□	□	□	□	

Column Name: V_ORDER   Data Type: CHAR(1)

Charset/Collation: Default Char: Default Collat   Default:

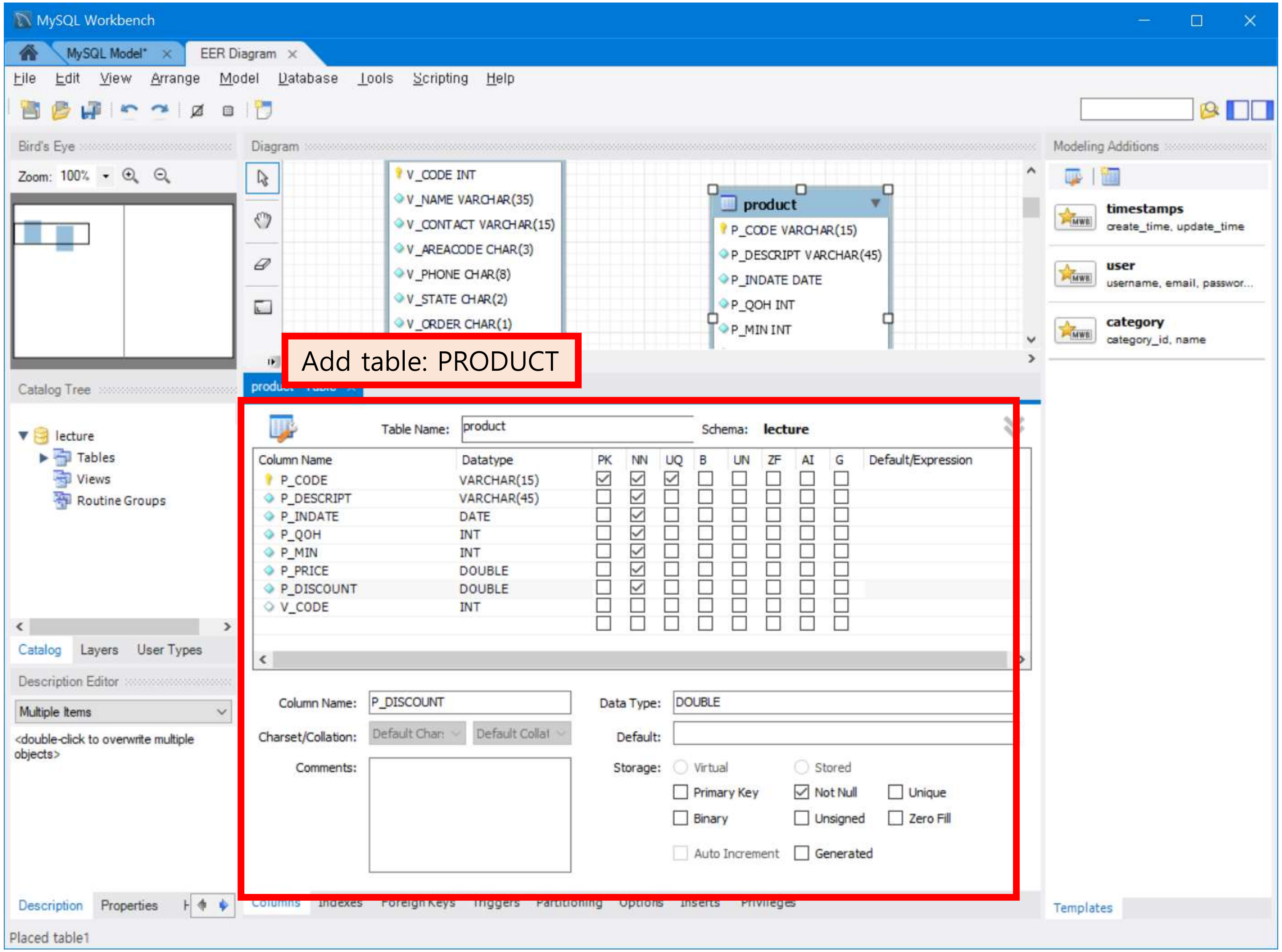
Comments:

Storage: ○ Virtual ○ Stored

□ Primary Key ☑ Not Null □ Unique

□ Binary □ Unsigned □ Zero Fill

□ Auto Increment □ Generated



V_CODE INT
V_NAME VARCHAR(35)
V_CONTACT VARCHAR(15)
V_AREACODE CHAR(3)
V_PHONE CHAR(8)
V_STATE CHAR(2)
V_ORDER CHAR(1)

product
P_CODE VARCHAR(15)
P_DESCRIPT VARCHAR(45)
P_INDATE DATE
P_QOH INT
P_MIN INT

Add table: PRODUCT

Table Name: product Schema: **lecture**

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
P_CODE	VARCHAR(15)	☑	☑	☑	□	□	□	□	□	
P_DESCRIPT	VARCHAR(45)	□	☑	□	□	□	□	□	□	
P_INDATE	DATE	□	☑	□	□	□	□	□	□	
P_QOH	INT	□	☑	□	□	□	□	□	□	
P_MIN	INT	□	☑	□	□	□	□	□	□	
P_PRICE	DOUBLE	□	☑	□	□	□	□	□	□	
P_DISCOUNT	DOUBLE	□	☑	□	□	□	□	□	□	
V_CODE	INT	□	□	□	□	□	□	□	□	
		□	□	□	□	□	□	□	□	

Column Name: P_DISCOUNT

Data Type: DOUBLE

Charset/Collation: Default Char: Default Collat

Default:

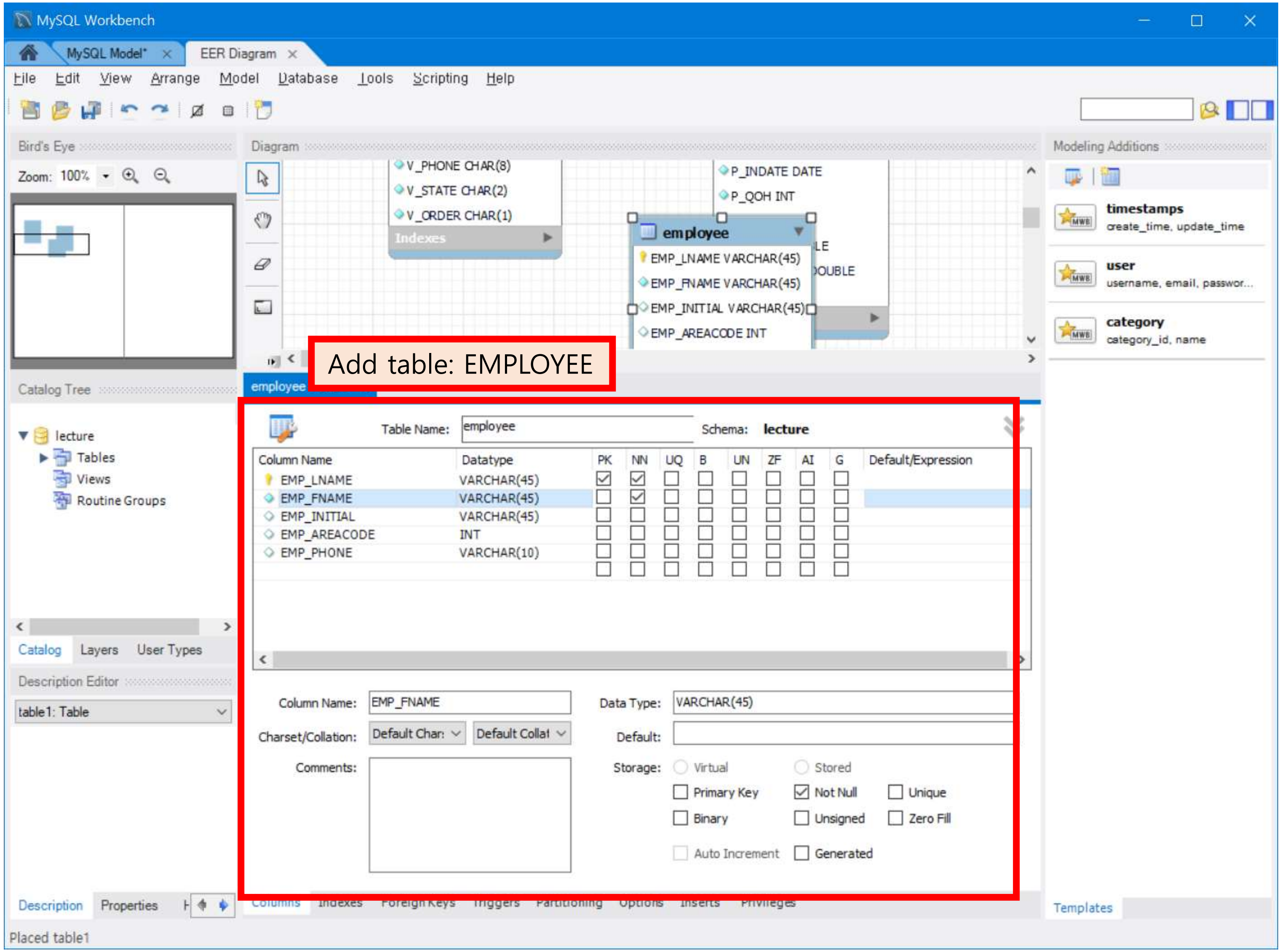
Comments:

Storage: ○ Virtual ○ Stored

□ Primary Key ☑ Not Null □ Unique

□ Binary □ Unsigned □ Zero Fill

□ Auto Increment □ Generated



Add table: EMPLOYEE

employee

Table Name: employee Schema: lecture

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
EMP_LNAME	VARCHAR(45)	☑	☑	□	□	□	□	□	□	
EMP_FNAME	VARCHAR(45)	□	☑	□	□	□	□	□	□	
EMP_INITIAL	VARCHAR(45)	□	□	□	□	□	□	□	□	
EMP_AREACODE	INT	□	□	□	□	□	□	□	□	
EMP_PHONE	VARCHAR(10)	□	□	□	□	□	□	□	□	
		□	□	□	□	□	□	□	□	

Column Name: EMP_FNAME

Data Type: VARCHAR(45)

Charset/Collation: Default Char: Default Collat

Default:

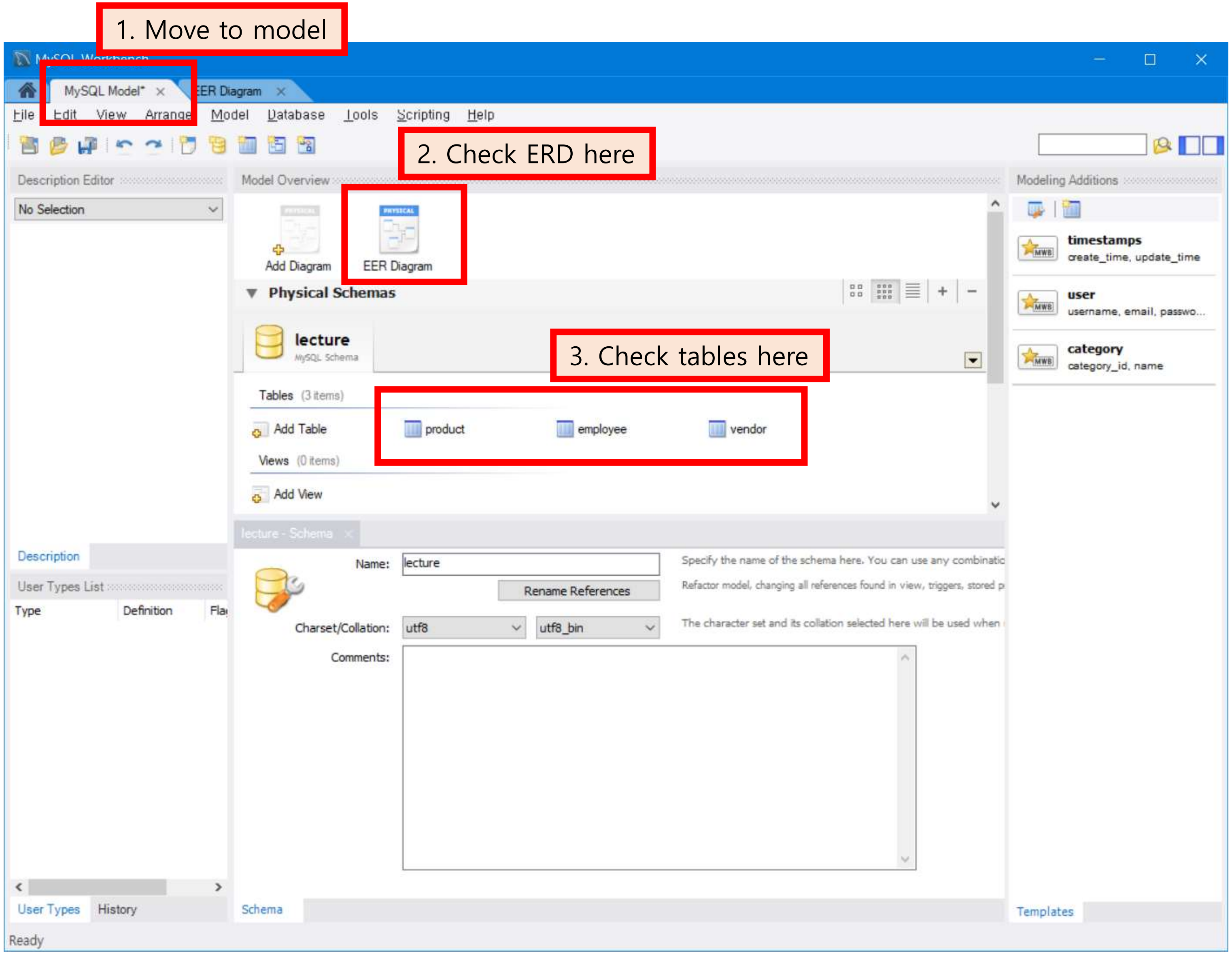
Comments:

Storage: ○ Virtual ○ Stored

□ Primary Key ☑ Not Null □ Unique

□ Binary □ Unsigned □ Zero Fill

□ Auto Increment □ Generated



1. Move to model

2. Check ERD here

3. Check tables here

lecture - Schema

Name: lecture

Rename References

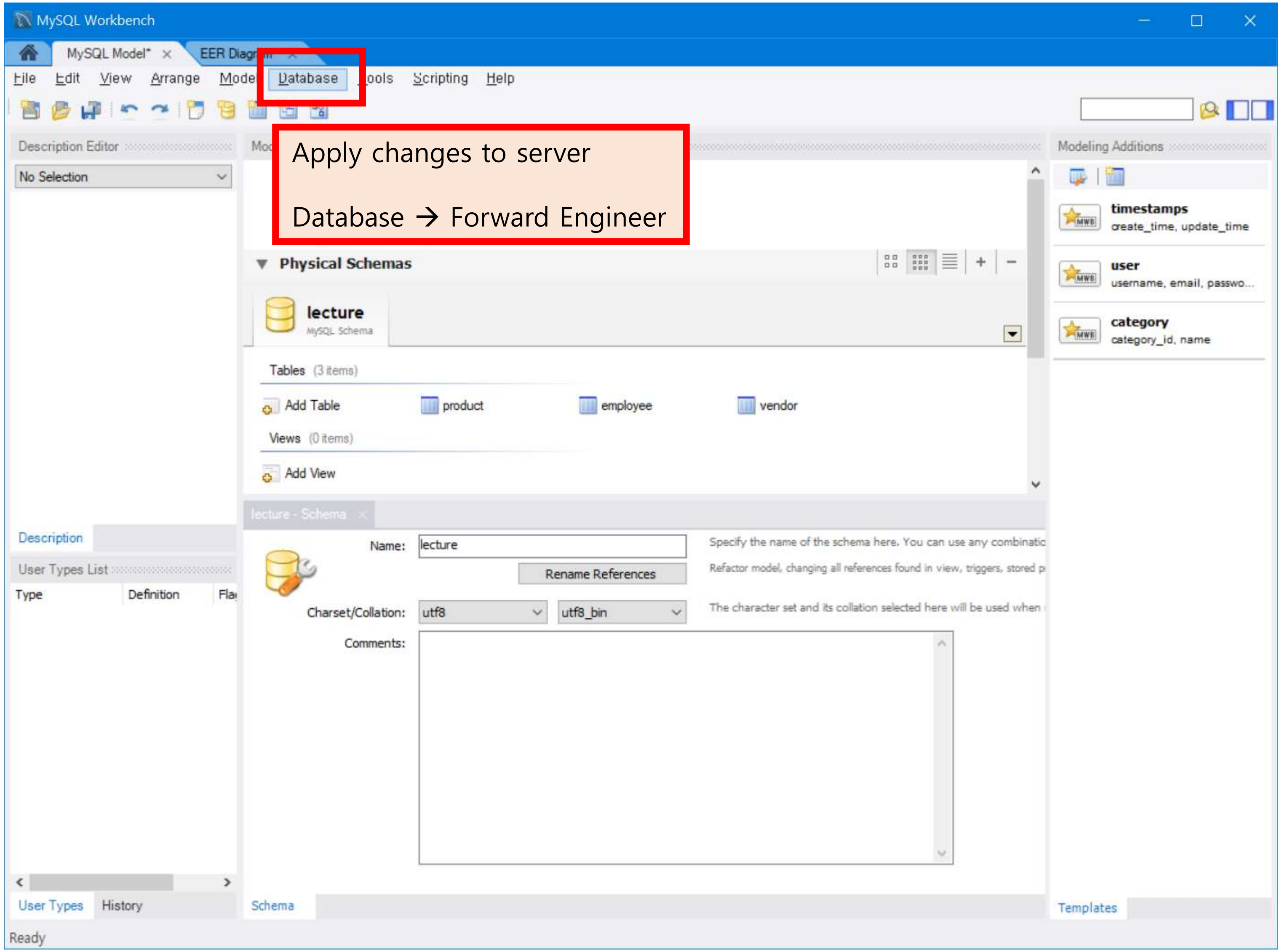
Specify the name of the schema here. You can use any combinatio

Refactor model, changing all references found in view, triggers, stored p

Charset/Collation: utf8 utf8_bin

The character set and its collation selected here will be used when

Comments:



Apply changes to server

Database → Forward Engineer

Name: lecture

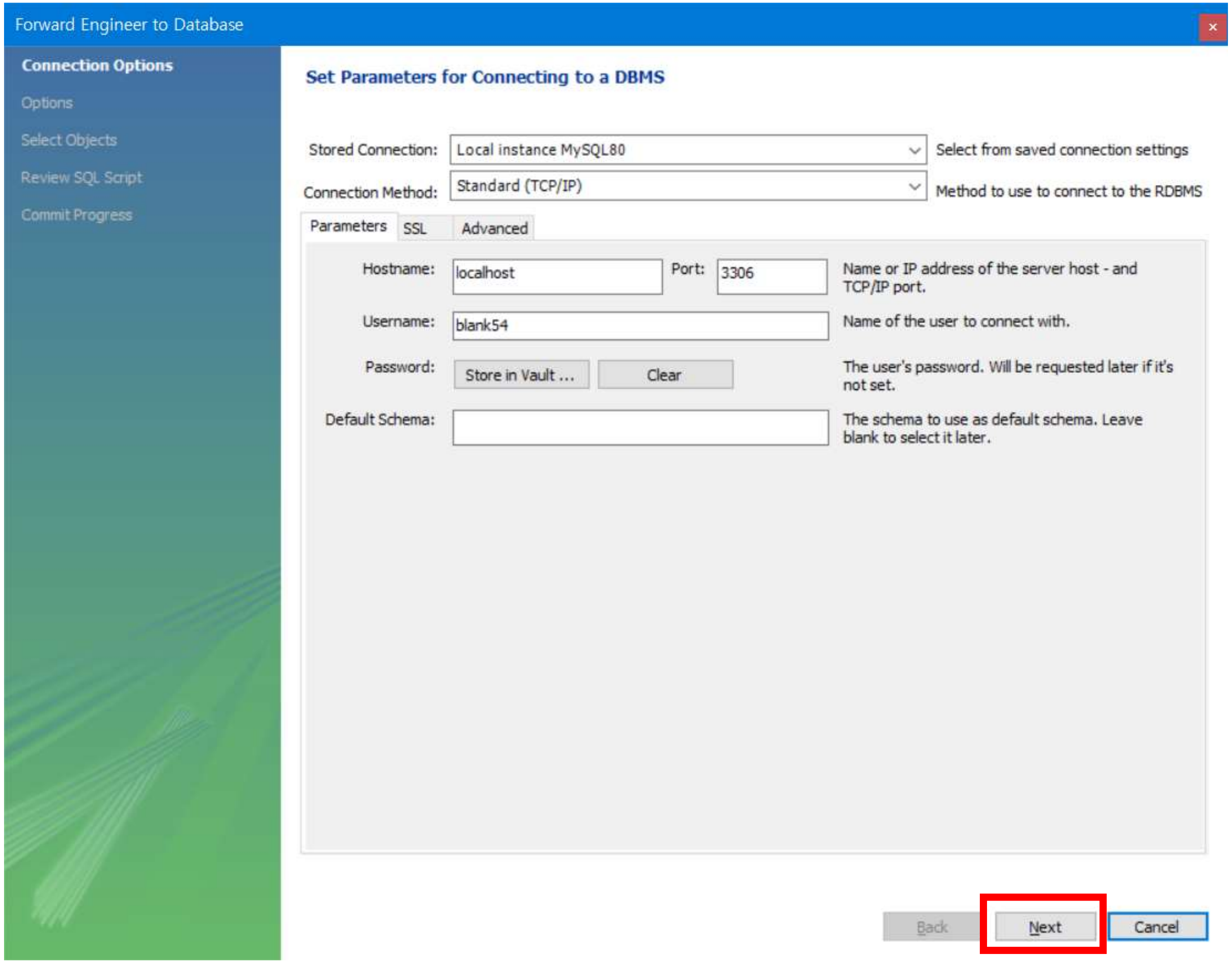
Charset/Collation: utf8 utf8_bin

Comments:

Specify the name of the schema here. You can use any combinatio

Refactor model, changing all references found in view, triggers, stored p

The character set and its collation selected here will be used when



Forward Engineer to Database

Set Parameters for Connecting to a DBMS

Stored Connection: Local instance MySQL80 — Select from saved connection settings

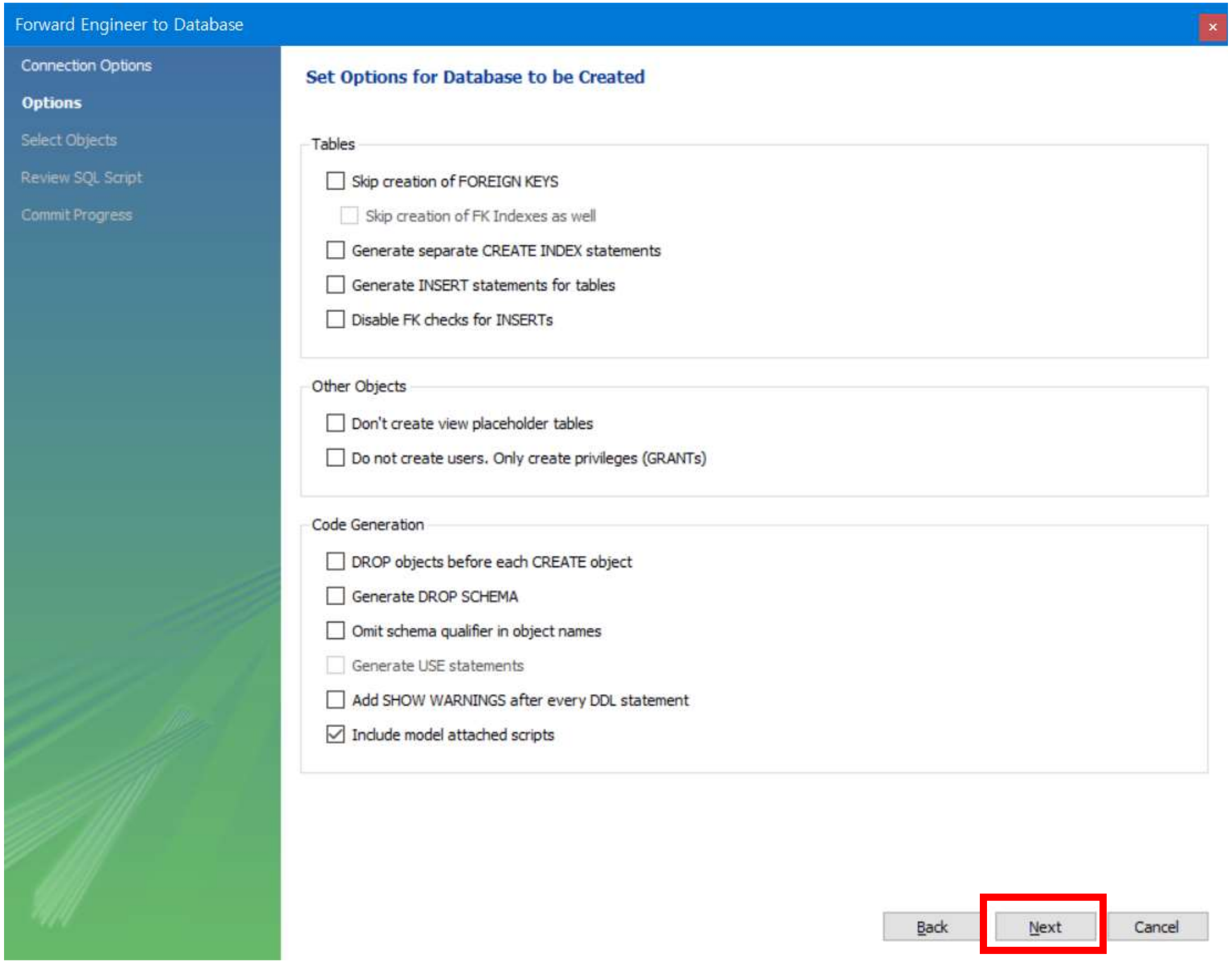
Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

Hostname: localhost Port: 3306 — Name or IP address of the server host - and TCP/IP port.

Username: blank54 — Name of the user to connect with.

Password: Store in Vault ... Clear — The user's password. Will be requested later if it's not set.

Default Schema: — The schema to use as default schema. Leave blank to select it later.



Set Options for Database to be Created

Tables

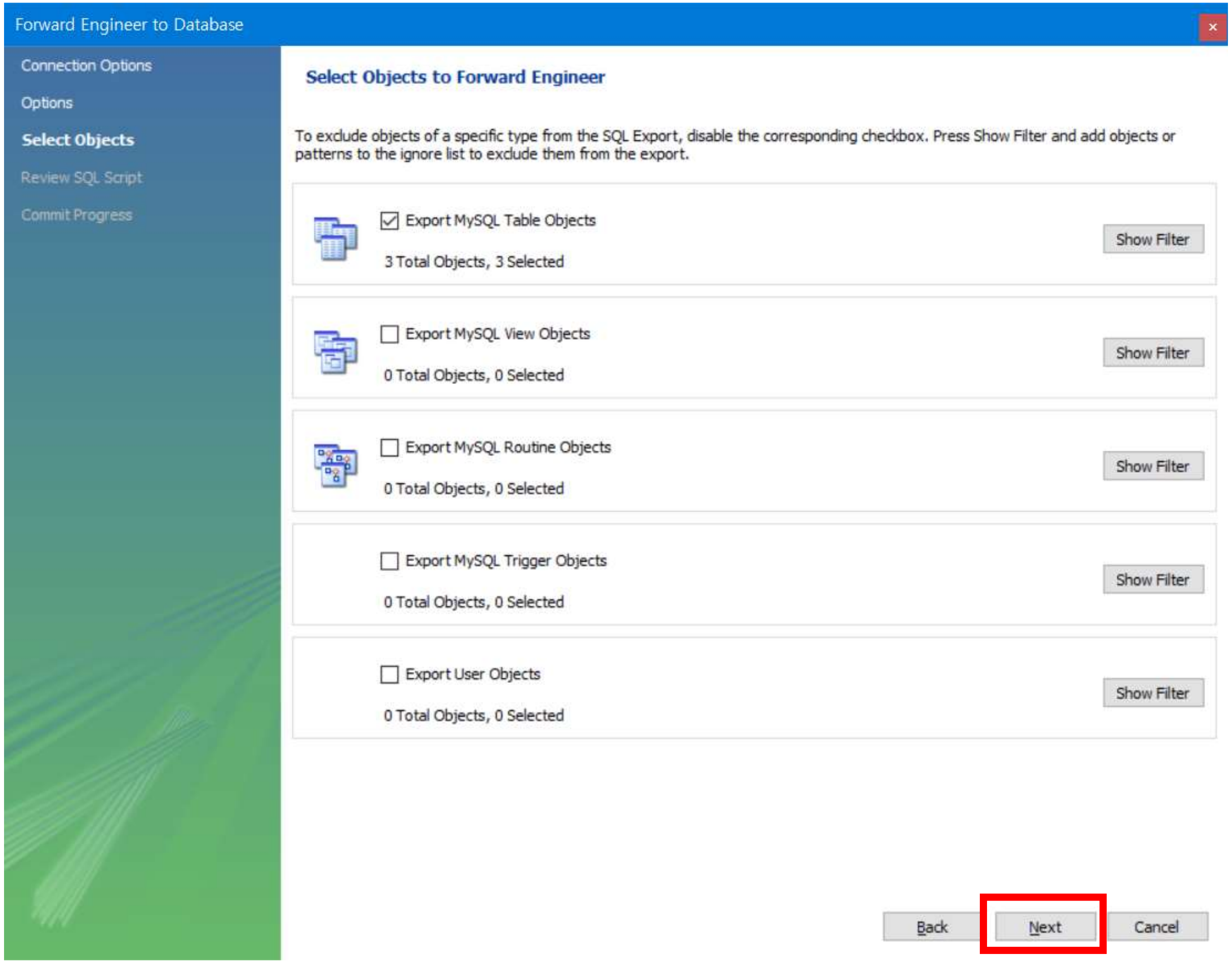
- □ Skip creation of FOREIGN KEYS
- □ Skip creation of FK Indexes as well
- □ Generate separate CREATE INDEX statements
- □ Generate INSERT statements for tables
- □ Disable FK checks for INSERTs

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



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

3 Total Objects, 3 Selected

☐ Export MySQL View Objects

0 Total Objects, 0 Selected

☐ Export MySQL Routine Objects

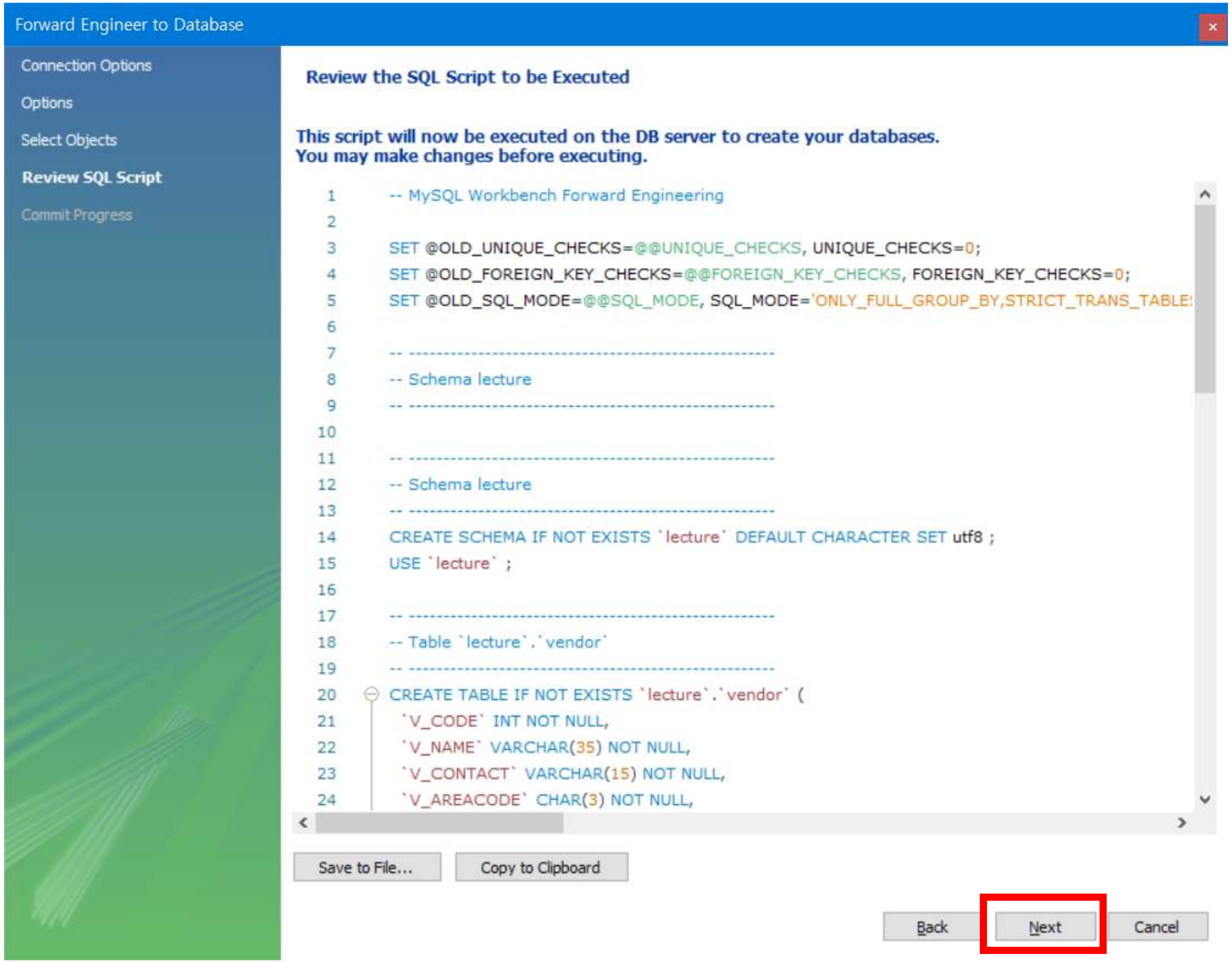
0 Total Objects, 0 Selected

☐ Export MySQL Trigger Objects

0 Total Objects, 0 Selected

☐ Export User Objects

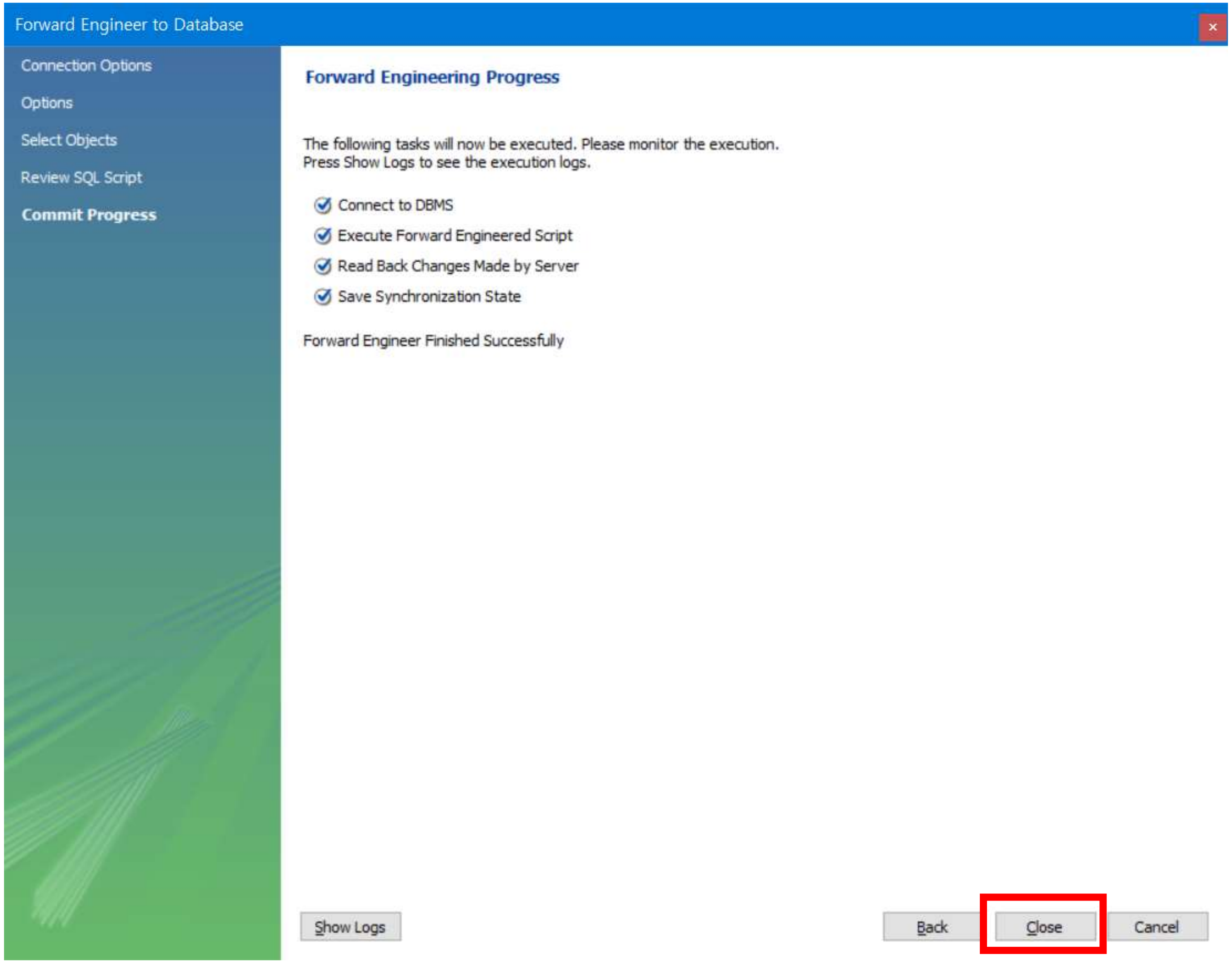
0 Total Objects, 0 Selected



Review the SQL Script to be Executed

This script will now be executed on the DB server to create your databases.
You may make changes before executing.

```
-- 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_TABLE

-- -----------------------------------------------------
-- Schema lecture
-- -----------------------------------------------------


-- -----------------------------------------------------
-- Schema lecture
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `lecture` DEFAULT CHARACTER SET utf8 ;
USE `lecture` ;

-- -----------------------------------------------------
-- Table `lecture`.`vendor`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `lecture`.`vendor` (
  `V_CODE` INT NOT NULL,
  `V_NAME` VARCHAR(35) NOT NULL,
  `V_CONTACT` VARCHAR(15) NOT NULL,
  `V_AREACODE` CHAR(3) NOT NULL,
```

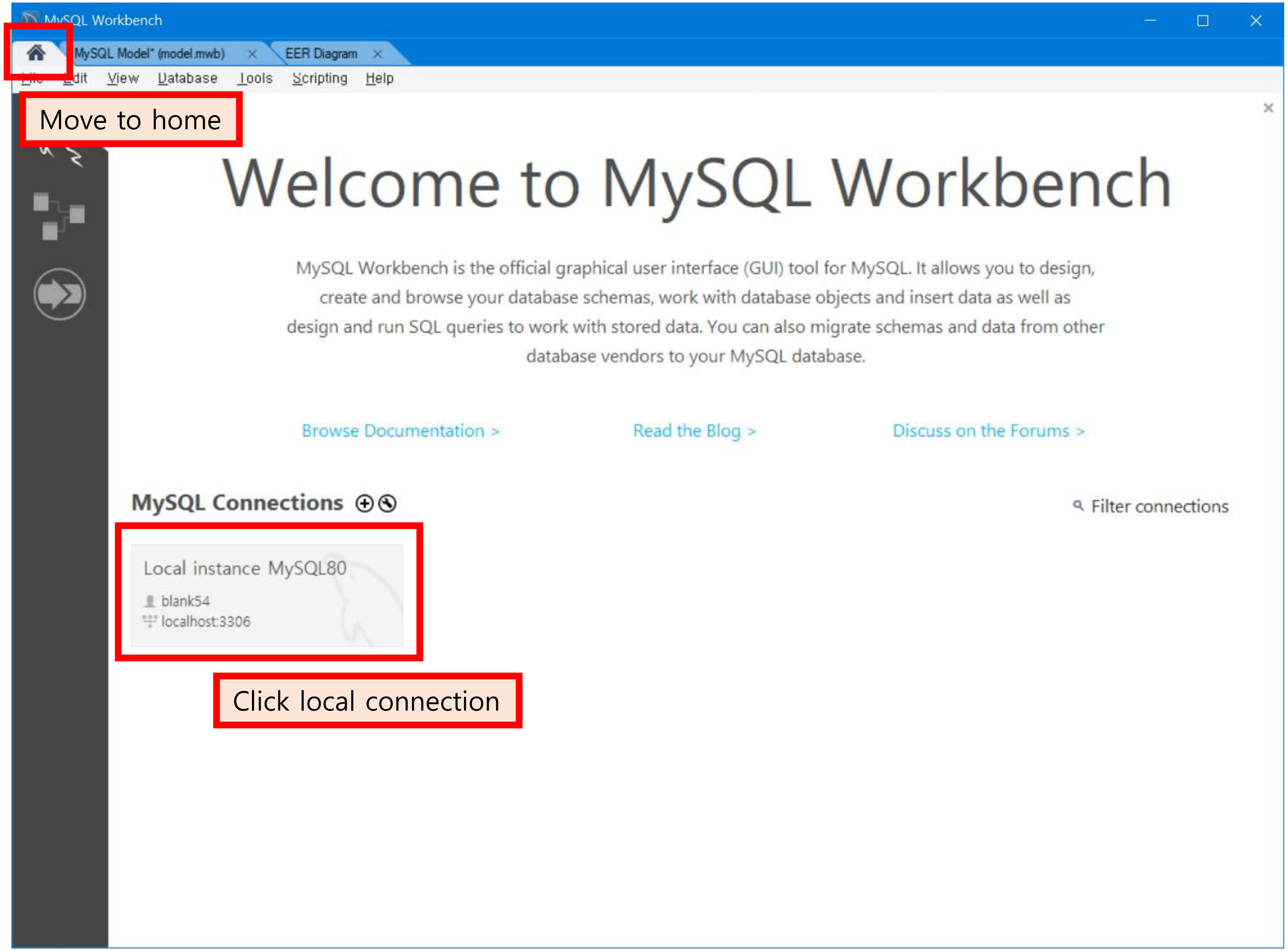


Forward Engineering Progress

The following tasks will now be executed. Please monitor the execution.
Press Show Logs to see the execution logs.

- ✓ Connect to DBMS
- ✓ Execute Forward Engineered Script
- ✓ Read Back Changes Made by Server
- ✓ Save Synchronization State

Forward Engineer Finished Successfully

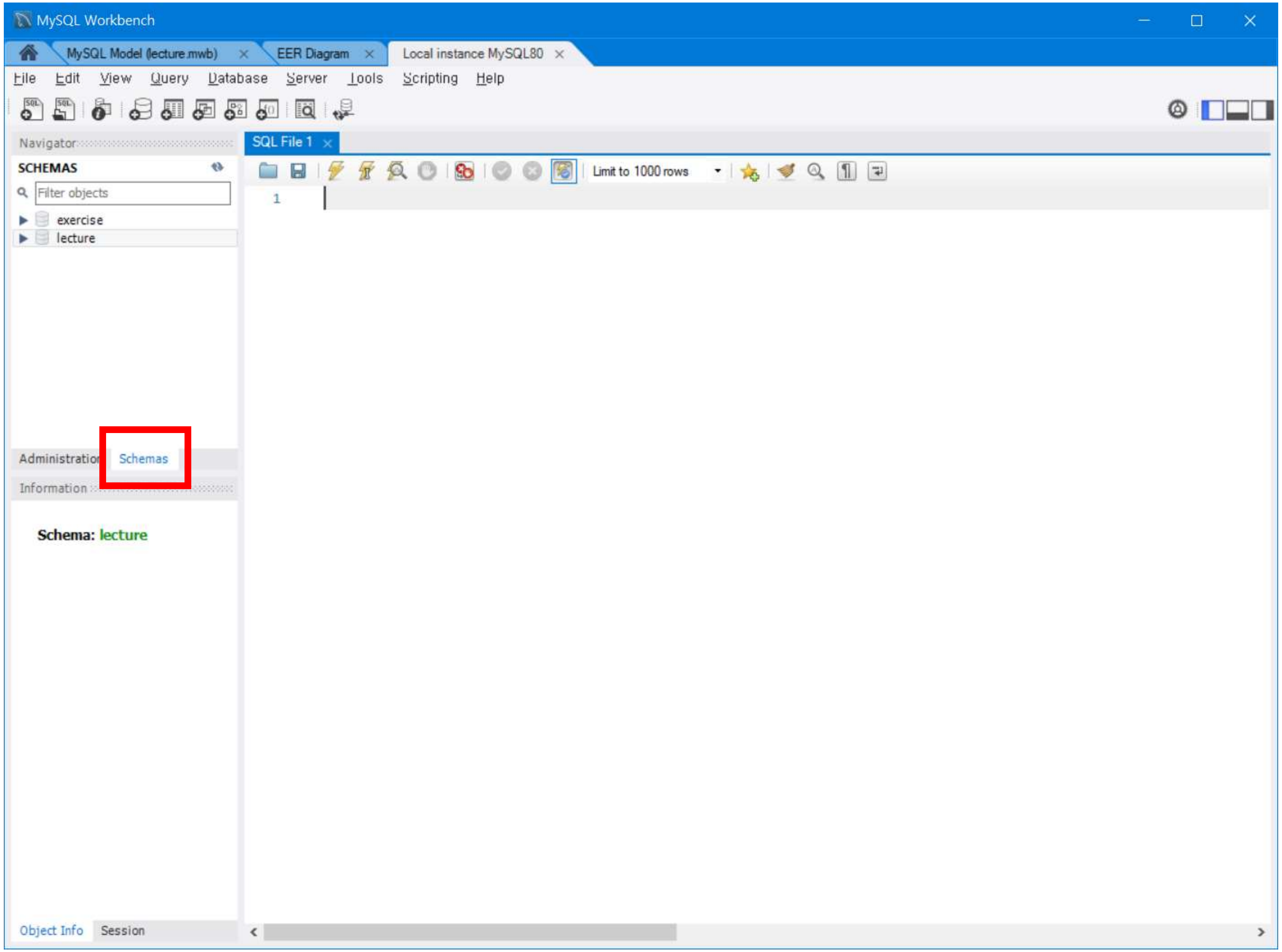


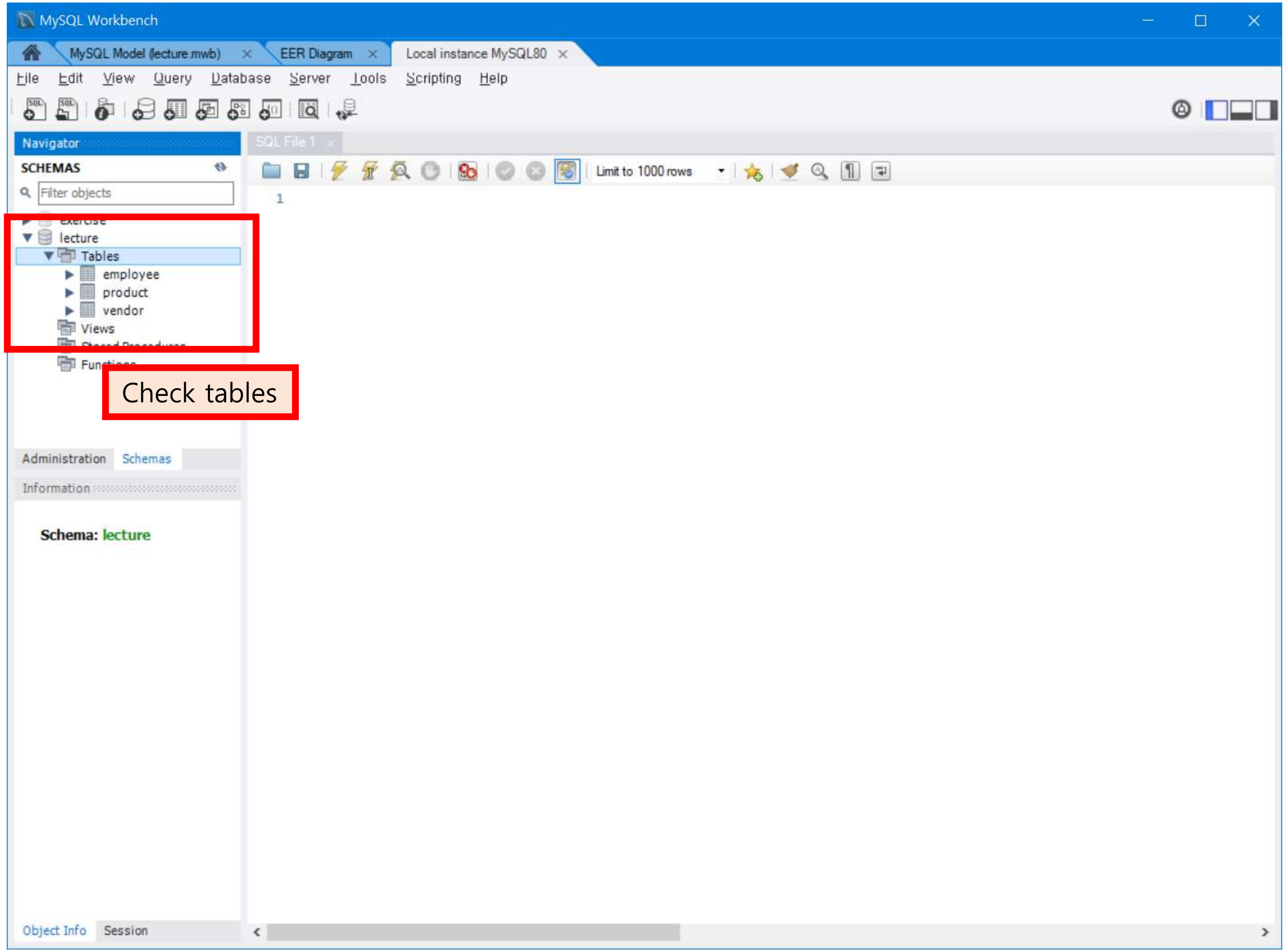
Move to home

Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

Click local connection


MySQL Workbench
MySQL Model (lecture.mwb)
EER Diagram
Local instance MySQL80
File
Edit
View
Query
Database
Server
Tools
Scripting
Help
Navigator
SCHEMAS
Filter objects
exercise
lecture
SQL File 1
Limit to 1000 rows
1
Administration
Schemas
Information
Schema: lecture
Object Info
Session



Schema: lecture

Check tables

4 Insert Data

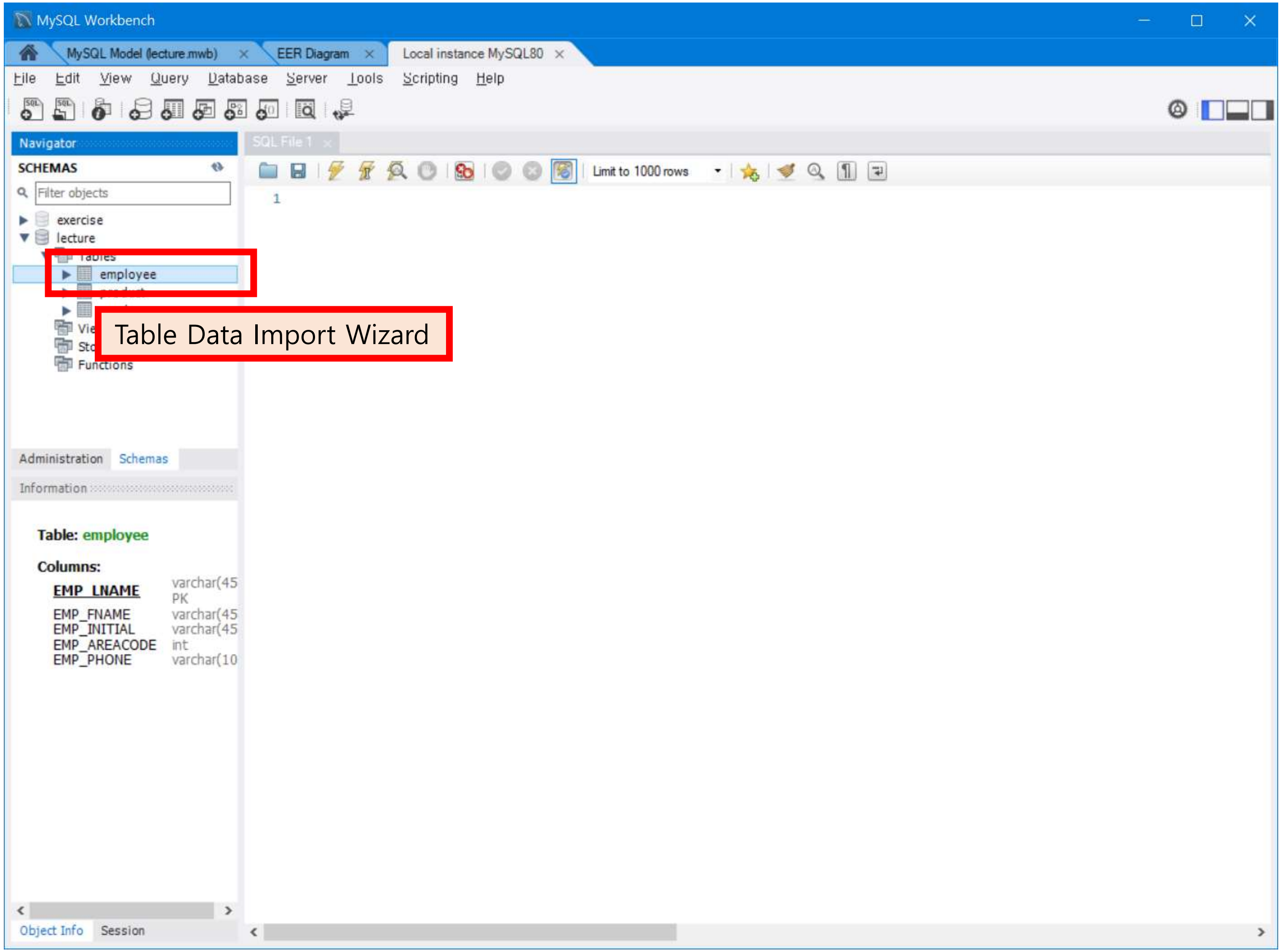


Table Data Import Wizard

Table: employee

Columns:

EMP_LNAME	varchar(45) PK
EMP_FNAME	varchar(45
EMP_INITIAL	varchar(45
EMP_AREACODE	int
EMP_PHONE	varchar(10

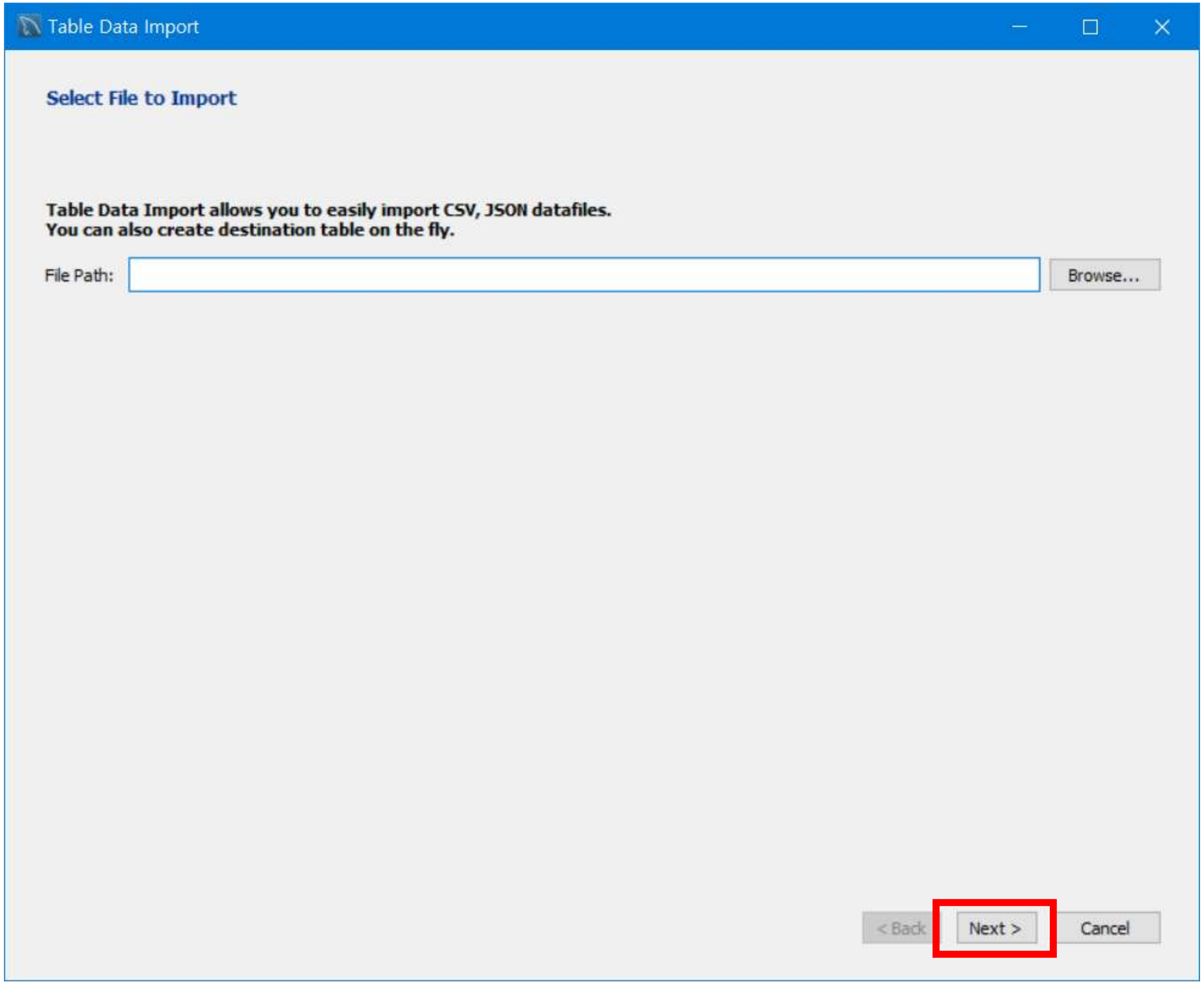
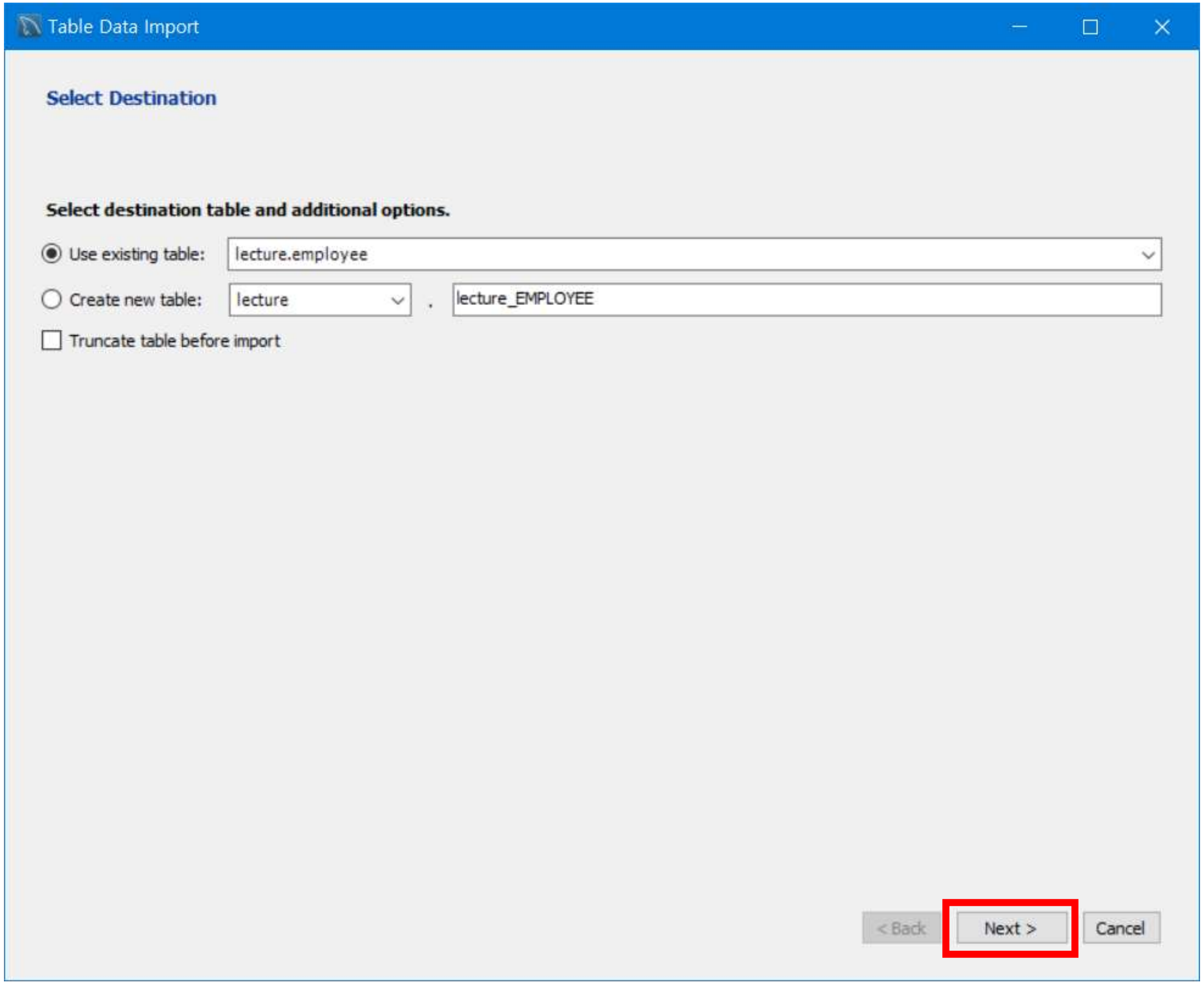


Table Data Import

Select File to Import

Table Data Import allows you to easily import CSV, JSON datafiles.
You can also create destination table on the fly.

File Path:



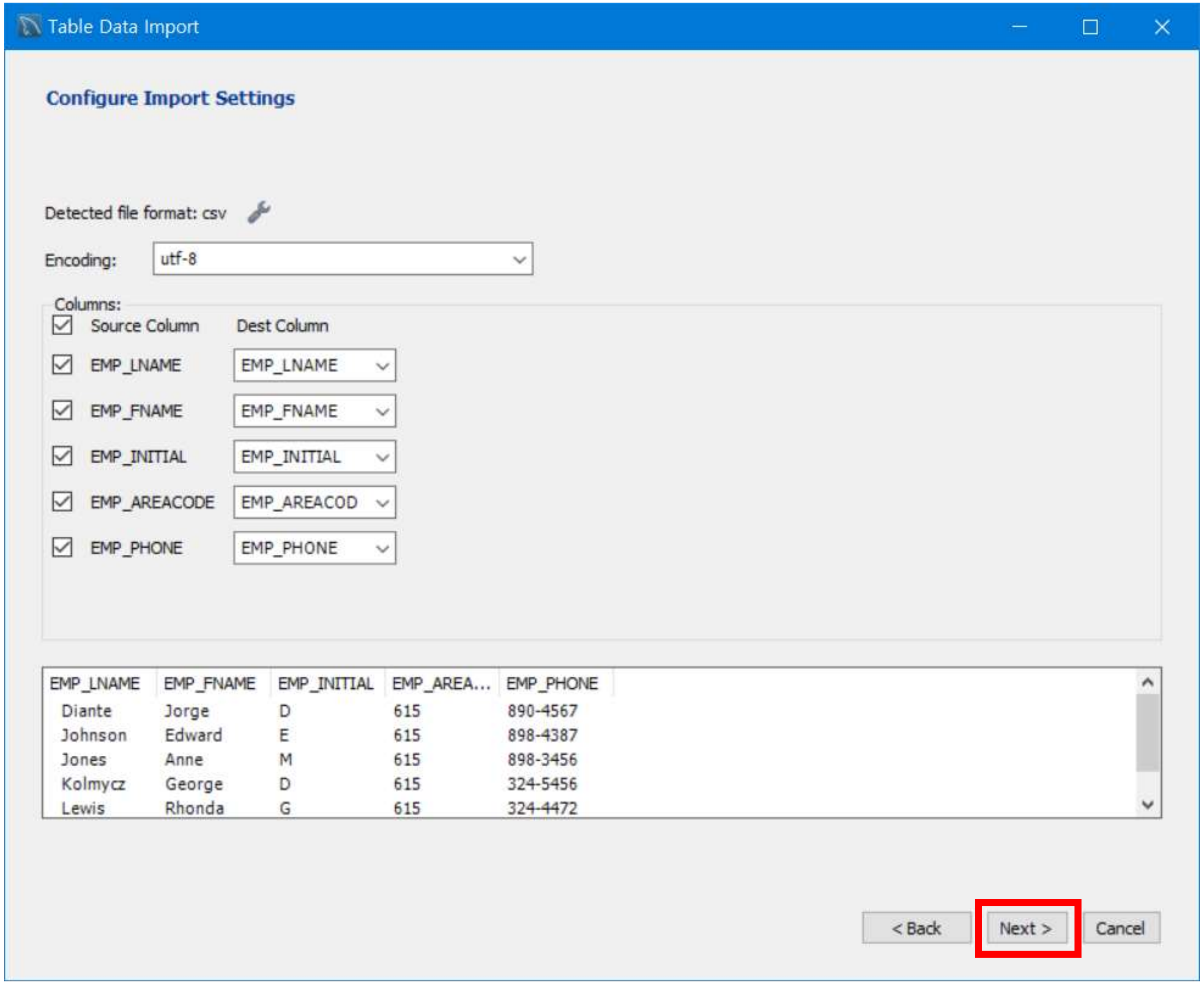
Select Destination

Select destination table and additional options.

● Use existing table: lecture.employee

○ Create new table: lecture . lecture_EMPLOYEE

□ Truncate table before import



Configure Import Settings

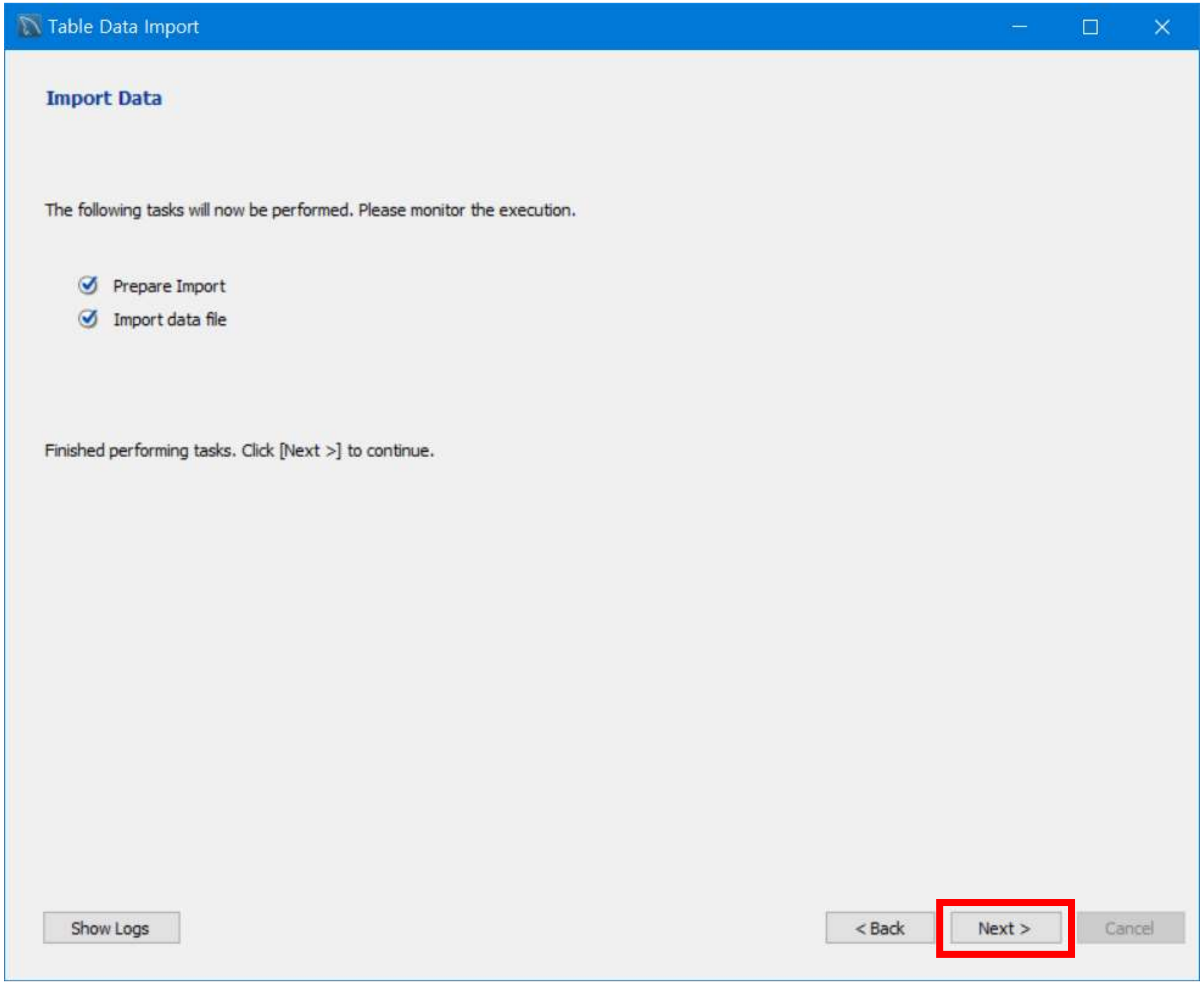
Detected file format: csv

Encoding: utf-8

Columns:

☑ Source Column	Dest Column
☑ EMP_LNAME	EMP_LNAME
☑ EMP_FNAME	EMP_FNAME
☑ EMP_INITIAL	EMP_INITIAL
☑ EMP_AREACODE	EMP_AREACOD
☑ EMP_PHONE	EMP_PHONE

EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_AREA...	EMP_PHONE
Diante	Jorge	D	615	890-4567
Johnson	Edward	E	615	898-4387
Jones	Anne	M	615	898-3456
Kolmycz	George	D	615	324-5456
Lewis	Rhonda	G	615	324-4472

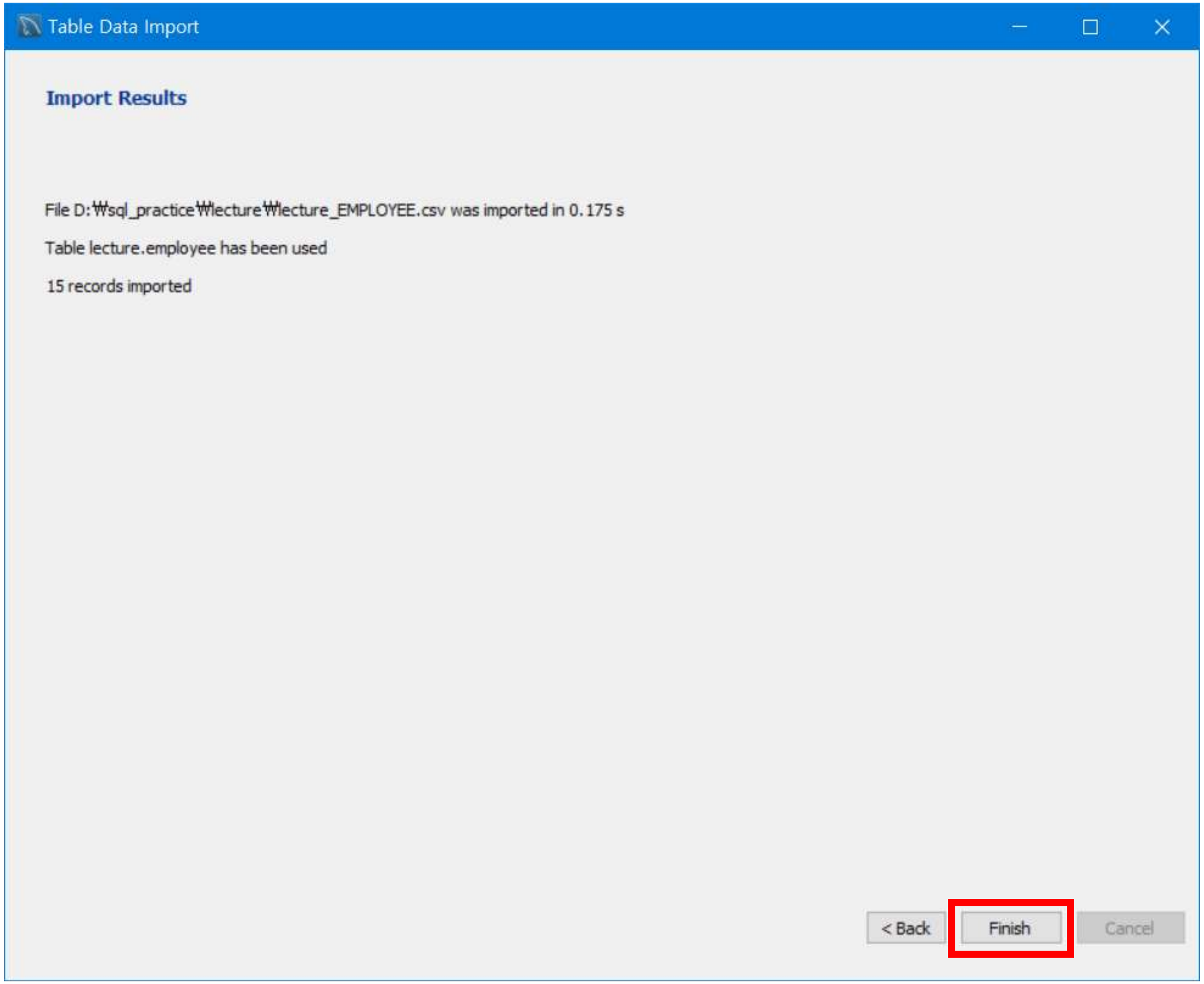


Import Data

The following tasks will now be performed. Please monitor the execution.

☑ Prepare Import
☑ Import data file

Finished performing tasks. Click [Next >] to continue.

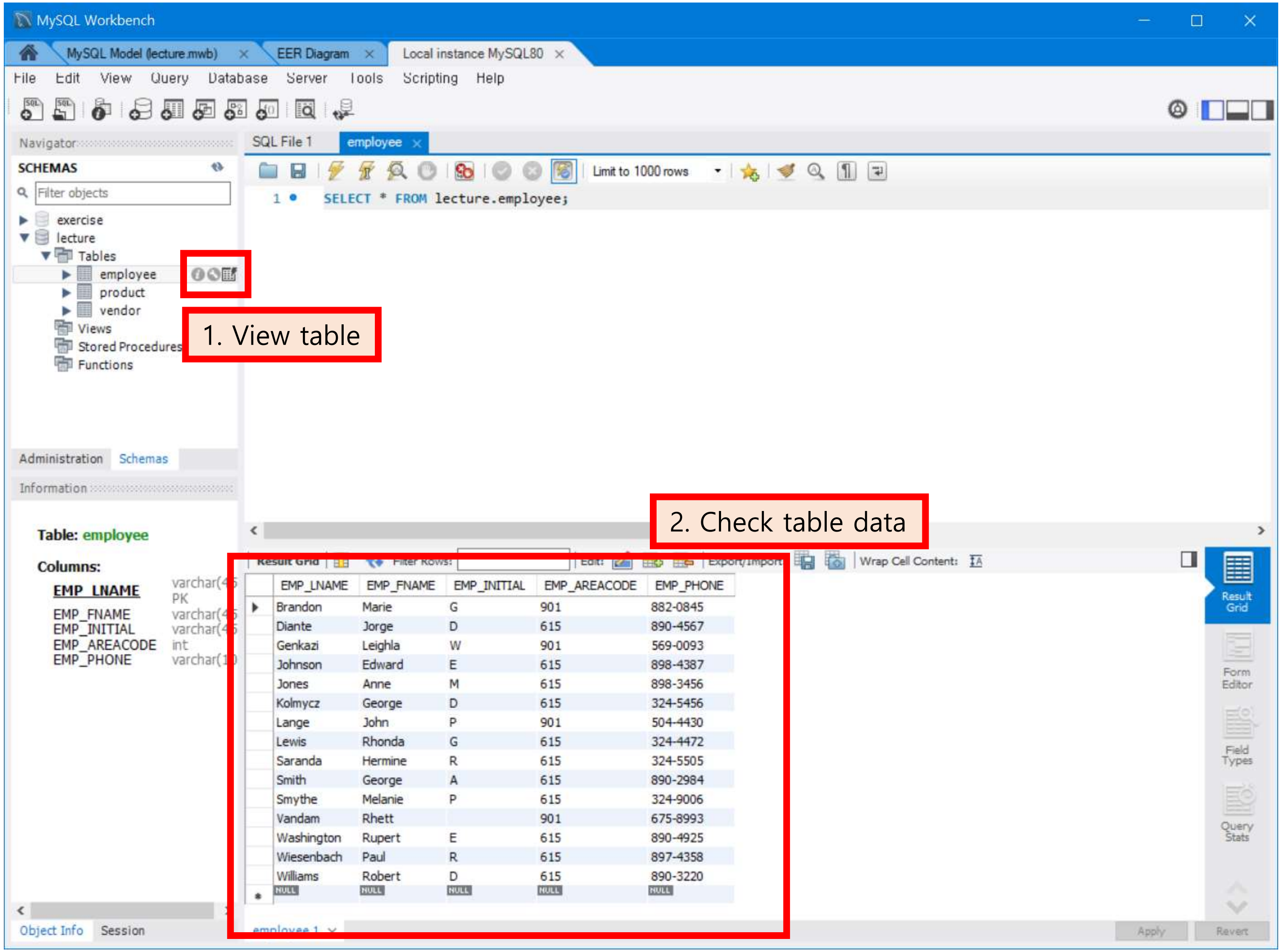


Import Results

File D:₩sql_practice₩lecture₩lecture_EMPLOYEE.csv was imported in 0.175 s

Table lecture.employee has been used

15 records imported



```sql
SELECT * FROM lecture.employee;
```

Table: employee

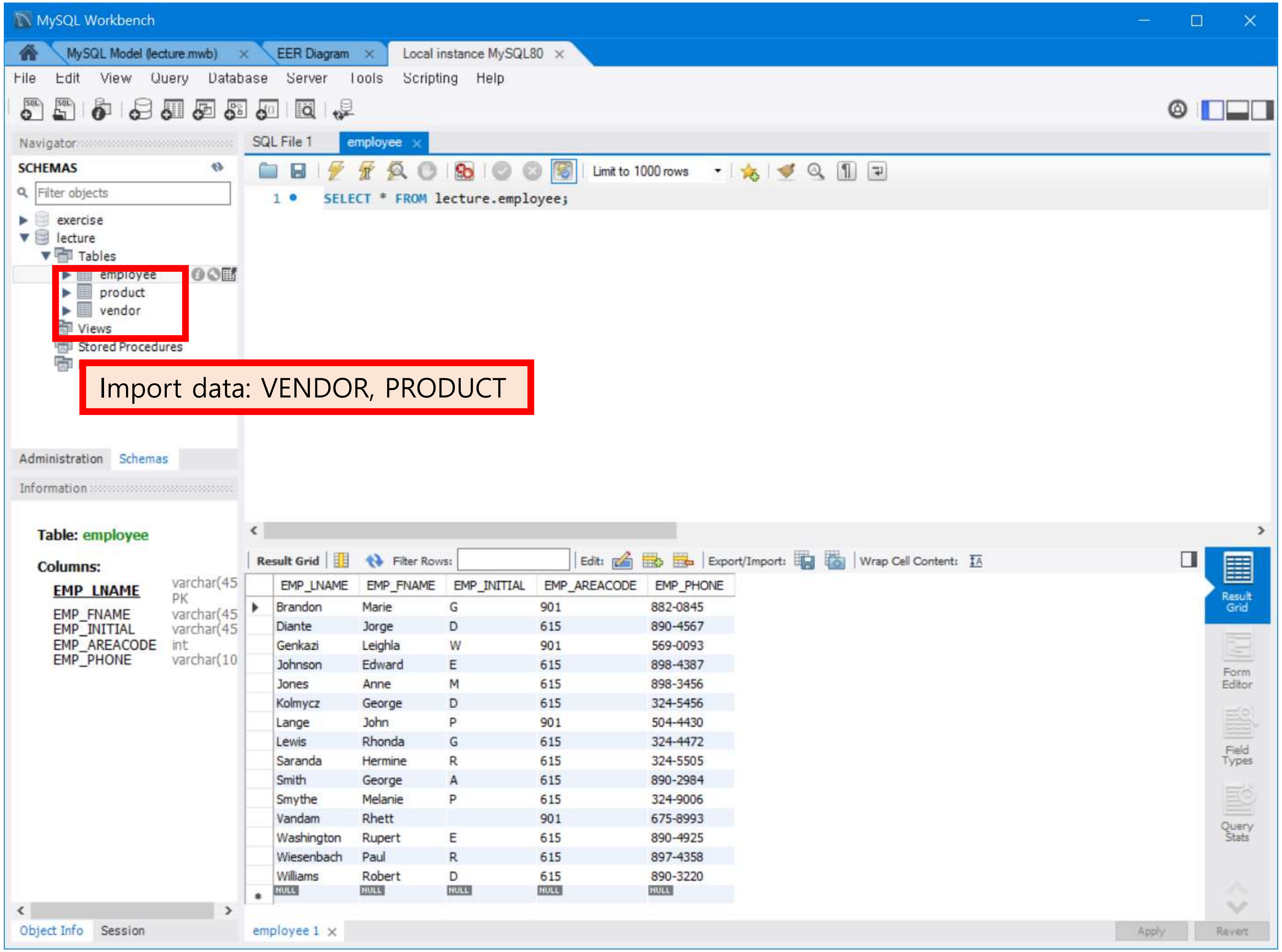
Columns:

EMP_LNAME varchar(4 PK
EMP_FNAME varchar(4
EMP_INITIAL varchar(4
EMP_AREACODE int
EMP_PHONE varchar(1

EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_AREACODE	EMP_PHONE
Brandon	Marie	G	901	882-0845
Diante	Jorge	D	615	890-4567
Genkazi	Leighla	W	901	569-0093
Johnson	Edward	E	615	898-4387
Jones	Anne	M	615	898-3456
Kolmycz	George	D	615	324-5456
Lange	John	P	901	504-4430
Lewis	Rhonda	G	615	324-4472
Saranda	Hermine	R	615	324-5505
Smith	George	A	615	890-2984
Smythe	Melanie	P	615	324-9006
Vandam	Rhett		901	675-8993
Washington	Rupert	E	615	890-4925
Wiesenbach	Paul	R	615	897-4358
Williams	Robert	D	615	890-3220
NULL	NULL	NULL	NULL	NULL

1. View table

2. Check table data



Import data: VENDOR, PRODUCT

```
SELECT * FROM lecture.employee;
```

Table: employee

Columns:

EMP_LNAME	varchar(45 PK
EMP_FNAME	varchar(45
EMP_INITIAL	varchar(45
EMP_AREACODE	int
EMP_PHONE	varchar(10

EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_AREACODE	EMP_PHONE
Brandon	Marie	G	901	882-0845
Diante	Jorge	D	615	890-4567
Genkazi	Leighla	W	901	569-0093
Johnson	Edward	E	615	898-4387
Jones	Anne	M	615	898-3456
Kolmycz	George	D	615	324-5456
Lange	John	P	901	504-4430
Lewis	Rhonda	G	615	324-4472
Saranda	Hermine	R	615	324-5505
Smith	George	A	615	890-2984
Smythe	Melanie	P	615	324-9006
Vandam	Rhett		901	675-8993
Washington	Rupert	E	615	890-4925
Wiesenbach	Paul	R	615	897-4358
Williams	Robert	D	615	890-3220
NULL	NULL	NULL	NULL	NULL