MySQL Practice

1. Running the program using the MySQL 8.0 Command Line Client

1.1 Enter the command to display all databases.

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 8.0.35 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| cool_company2      |
| coolcompany        |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
6 rows in set (0.00 sec)

mysql>
```

1.2 Select the database named "sys".

```
mysql> use sys;
Database changed
mysql>
```

1.3 Show all tables in the database "sys".

```
mysql> show tables;
+-----------------------------------------------+
| Tables_in_sys                                 |
+-----------------------------------------------+
| host_summary                                  |
| host_summary_by_file_io                       |
| host_summary_by_file_io_type                  |
| host_summary_by_stages                        |
| host_summary_by_statement_latency             |
| host_summary_by_statement_type                |
| innodb_buffer_stats_by_schema                 |
| innodb_buffer_stats_by_table                  |
| innodb_lock_waits                             |
```

1.4 Show all data in the "host-summary" table

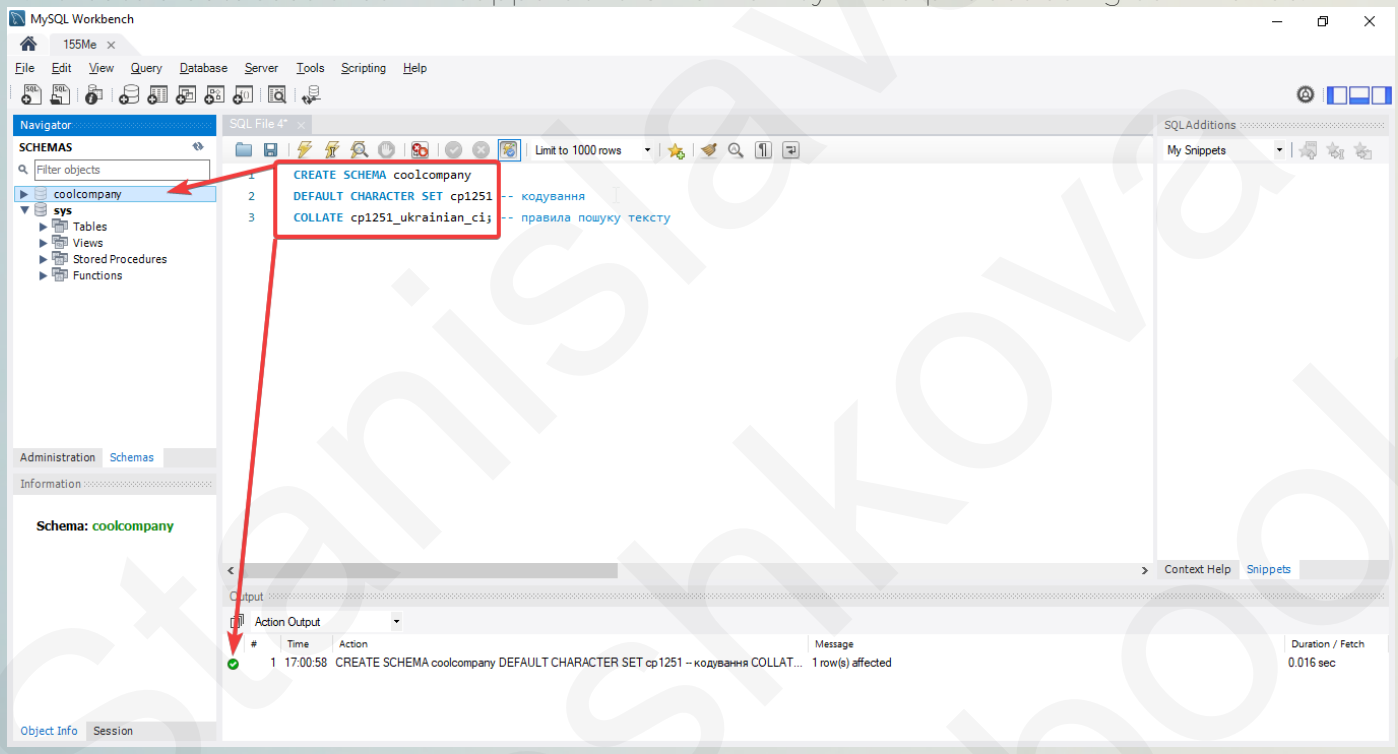
```
mysql> select * from host_summary;
+-----------+------------+-----------------+----------------------+-------------+---------+---------------+---------------------+
| host      | statements | statement_latency | statement_avg_latency | table_scans | file_ios | file_io_latency | current_connections |
 | current_memory | total_memory_allocated |
+-----------+------------+-----------------+----------------------+-------------+---------+---------------+---------------------+
| localhost |       3477 | 7.63 s          | 2.19 ms              |         330 |      279 | 139.24 ms       |                   2 |
 | 6.22 MiB       | 1.25 GiB               |
+-----------+------------+-----------------+----------------------+-------------+---------+---------------+---------------------+
1 row in set (0.07 sec)
```

1.5 Exit the command line.

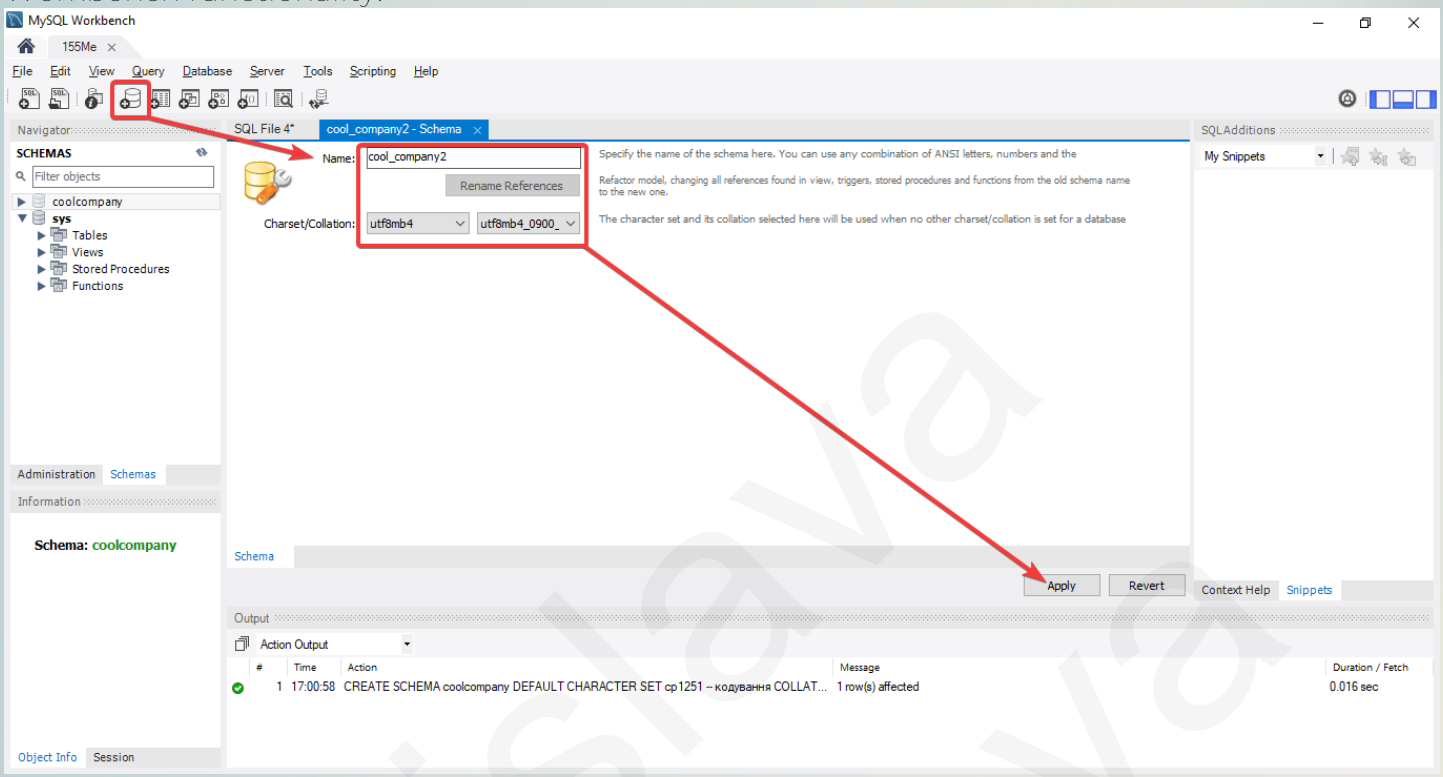
```
mysql> exit
```

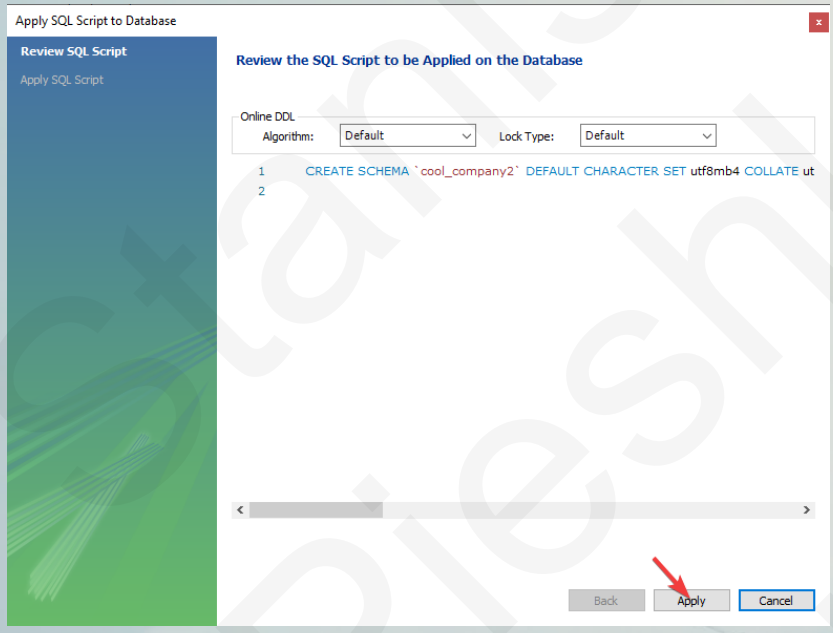
2. Create a database in two ways

2.1 Create a database that will support the Ukrainian Cyrillic alphabet using commands.



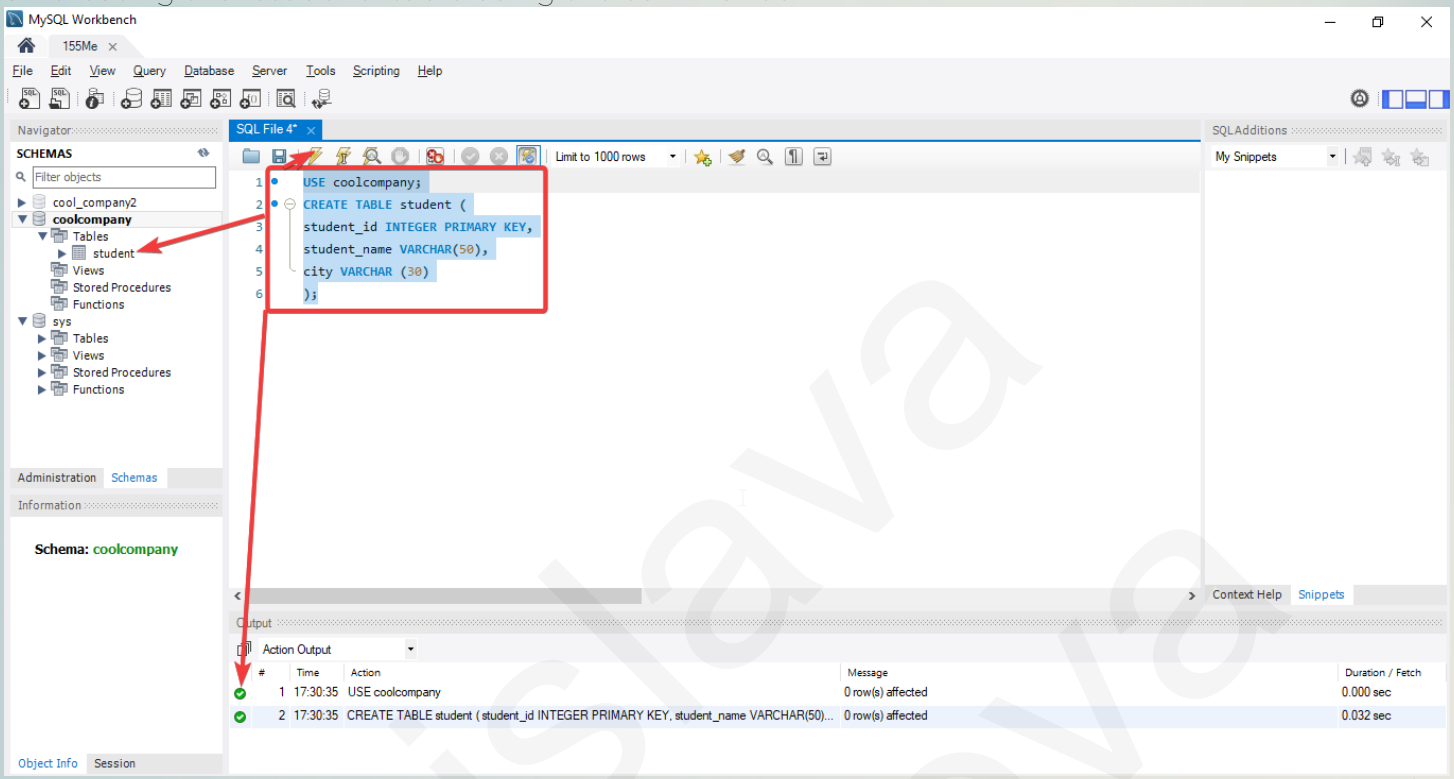
2.2 Create a database that will support the Ukrainian Cyrillic alphabet using the MySQL Workbench functionality.



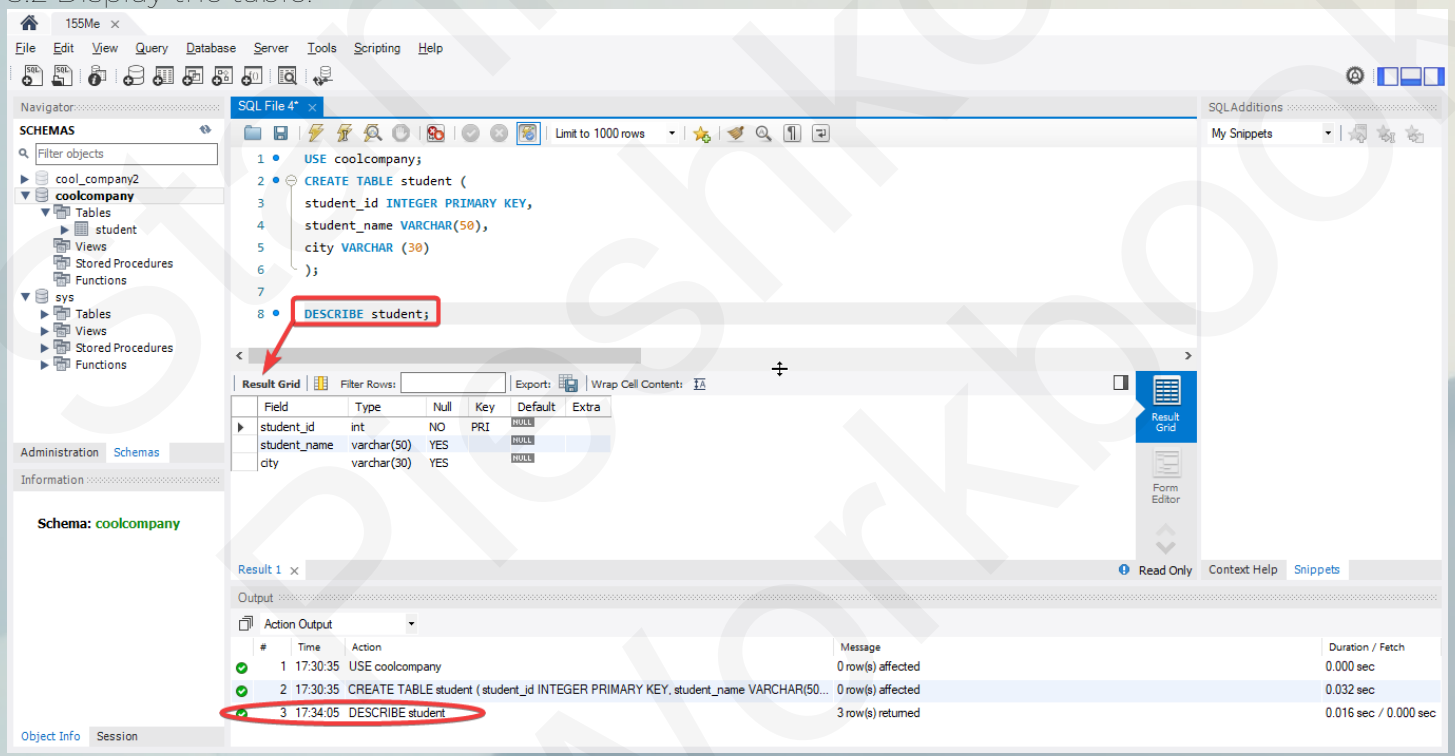


3. How to create a table in the Database (and what is the Primary Key)

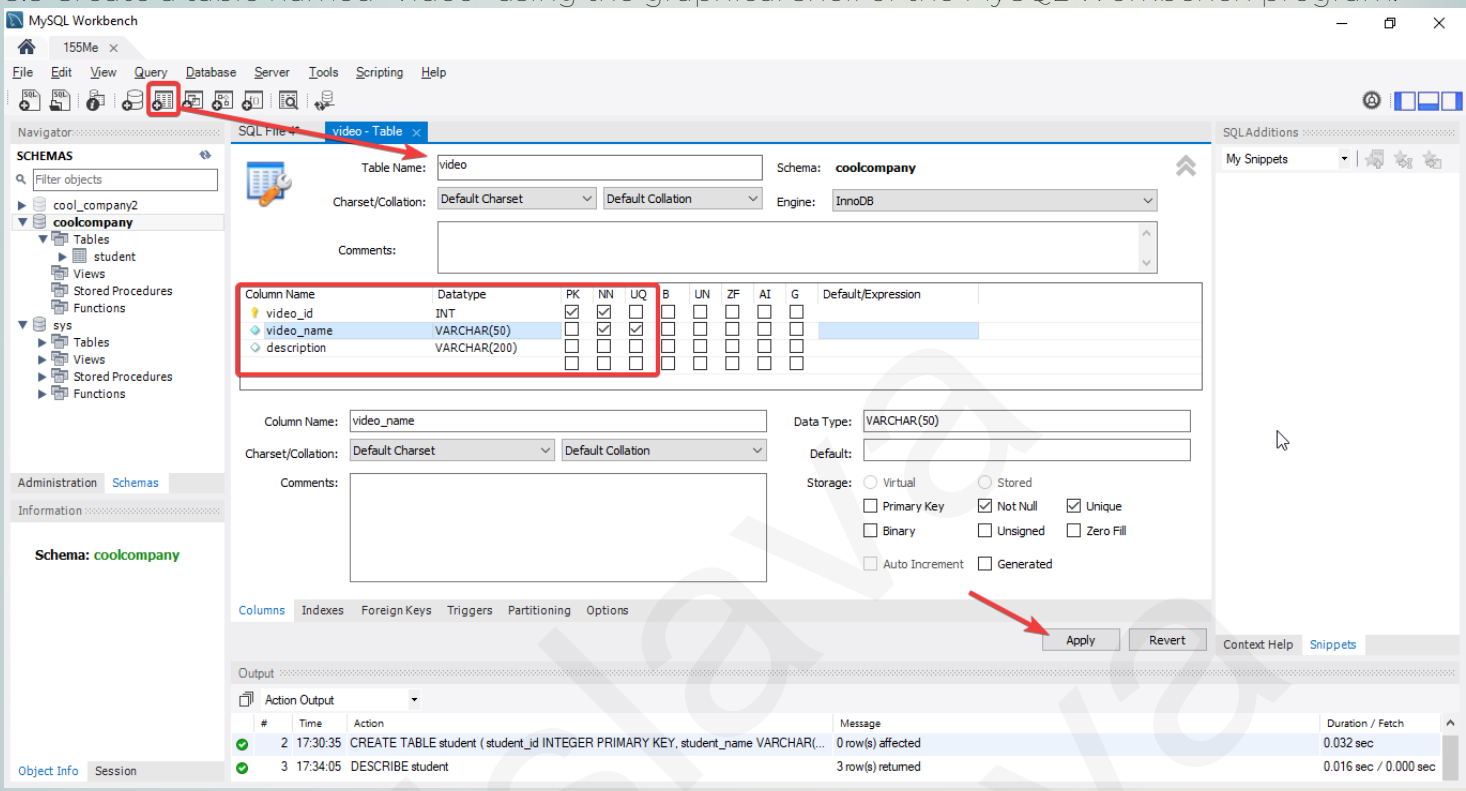
3.1 Creating the "student" table using the commands.

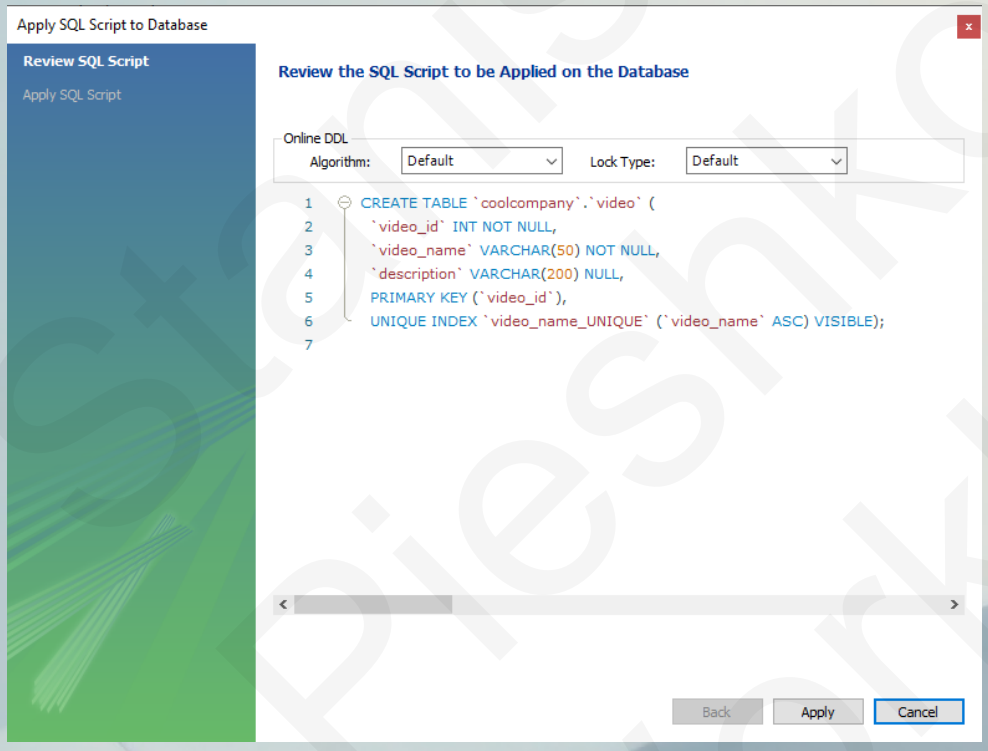


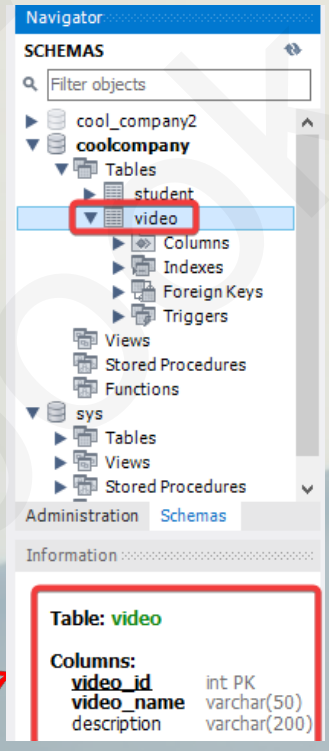
3.2 Display the table.



3.3 Create a table named "video" using the graphical shell of the MySQL Workbench program.

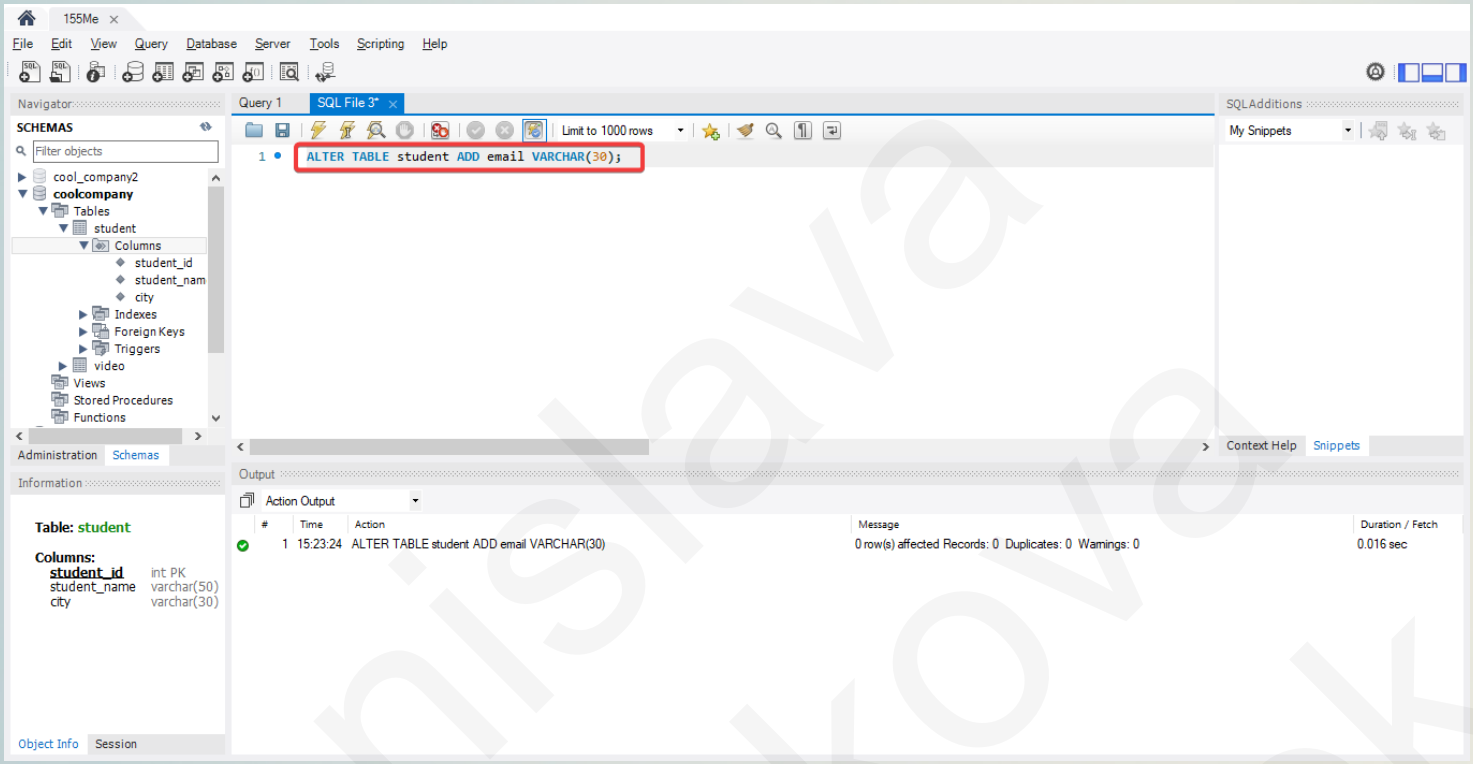


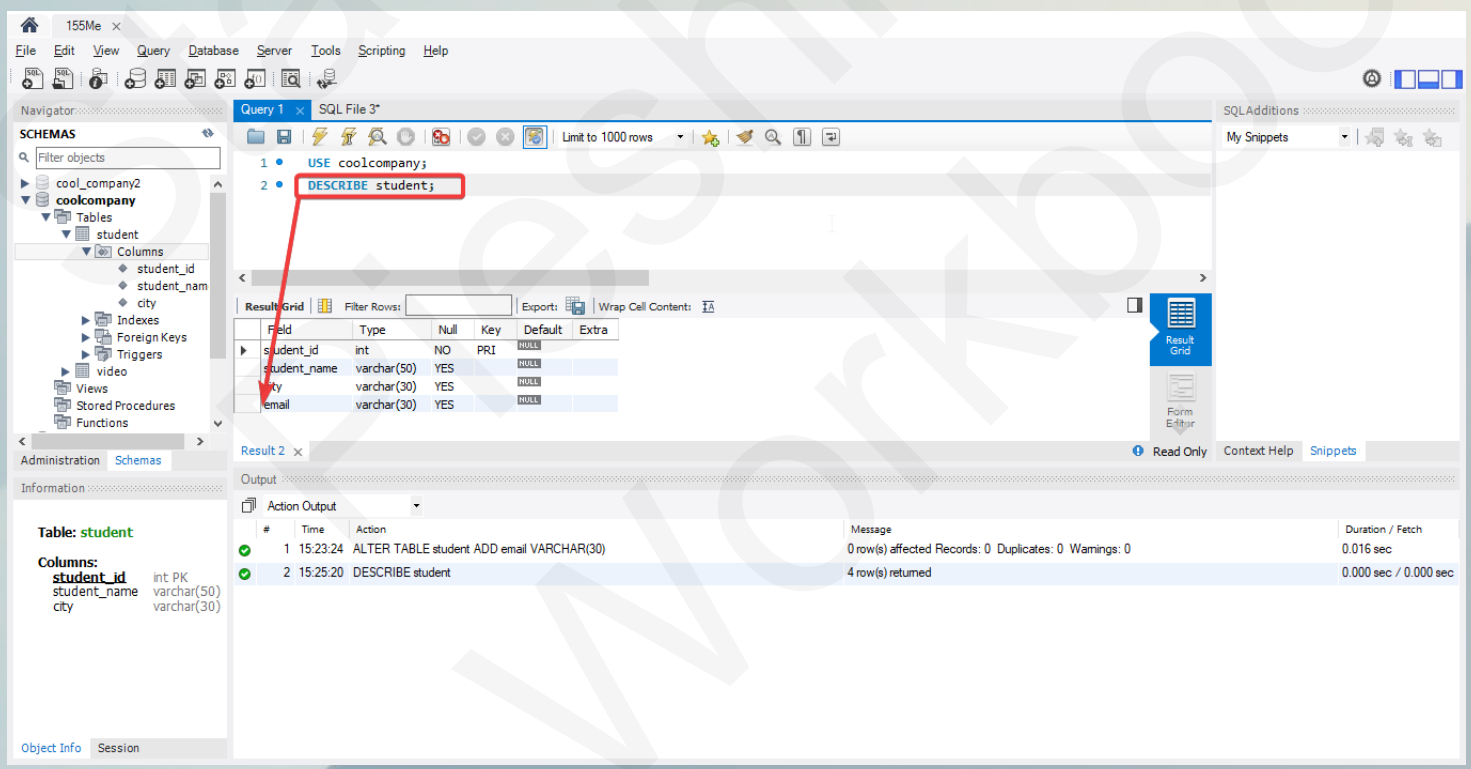




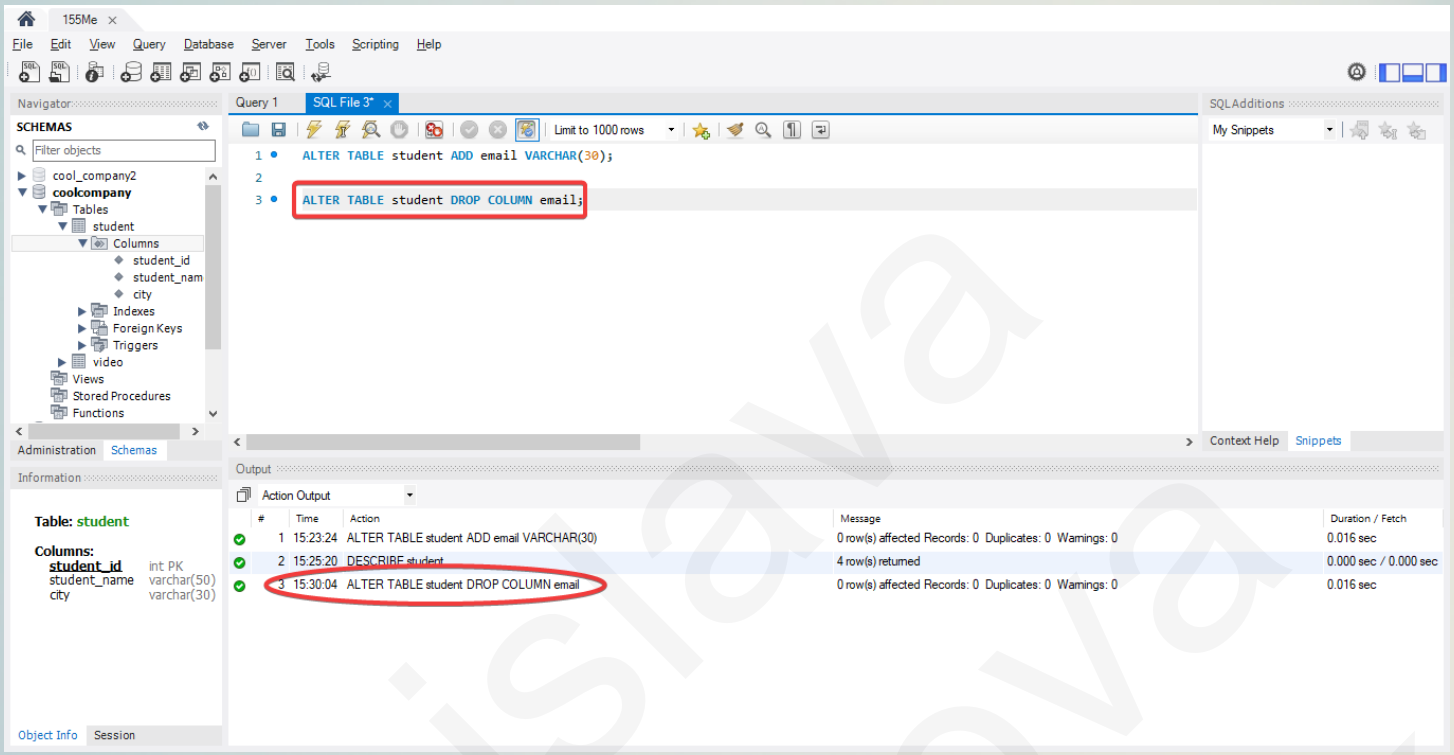
4. How to edit tables in a SQL database

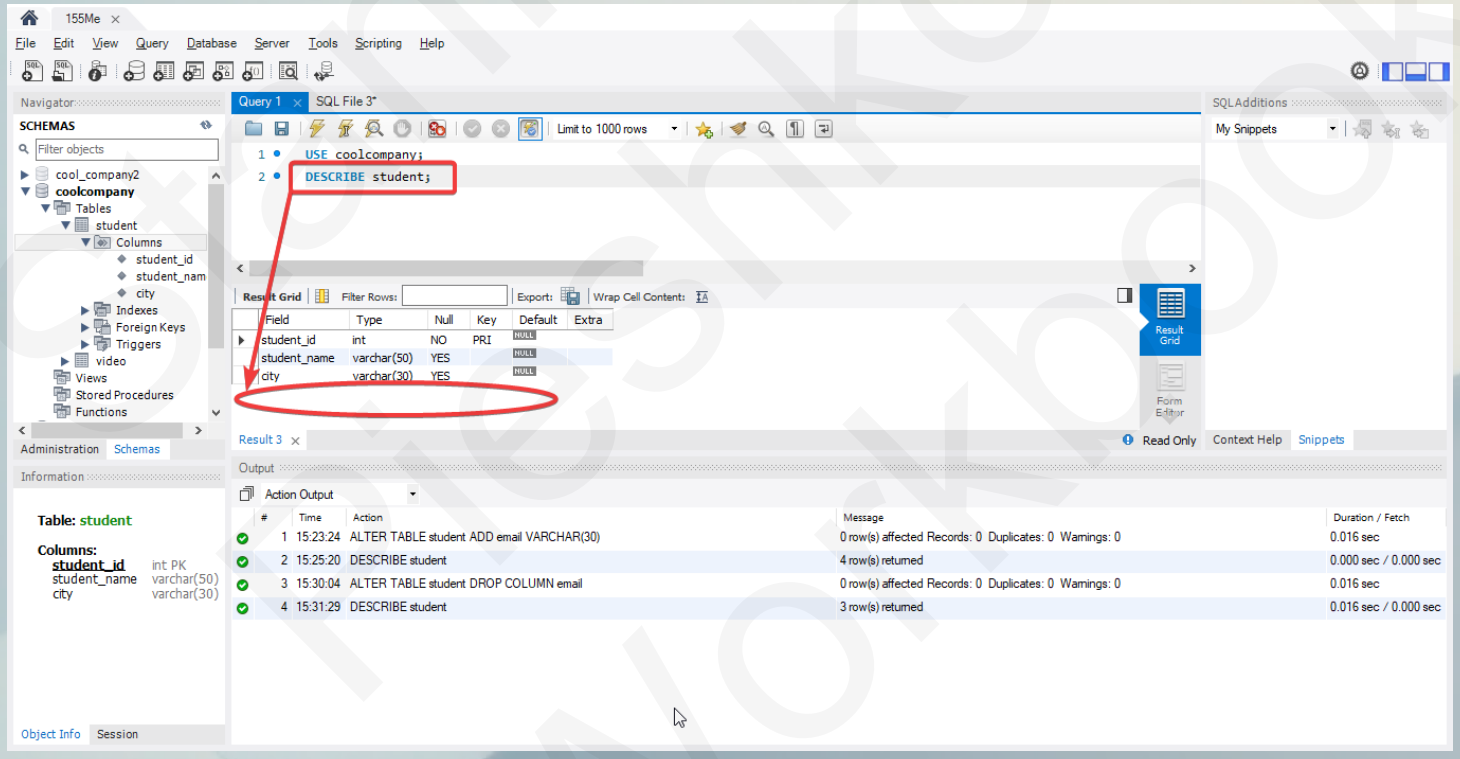
4.1 Add a column with emails to the student table.



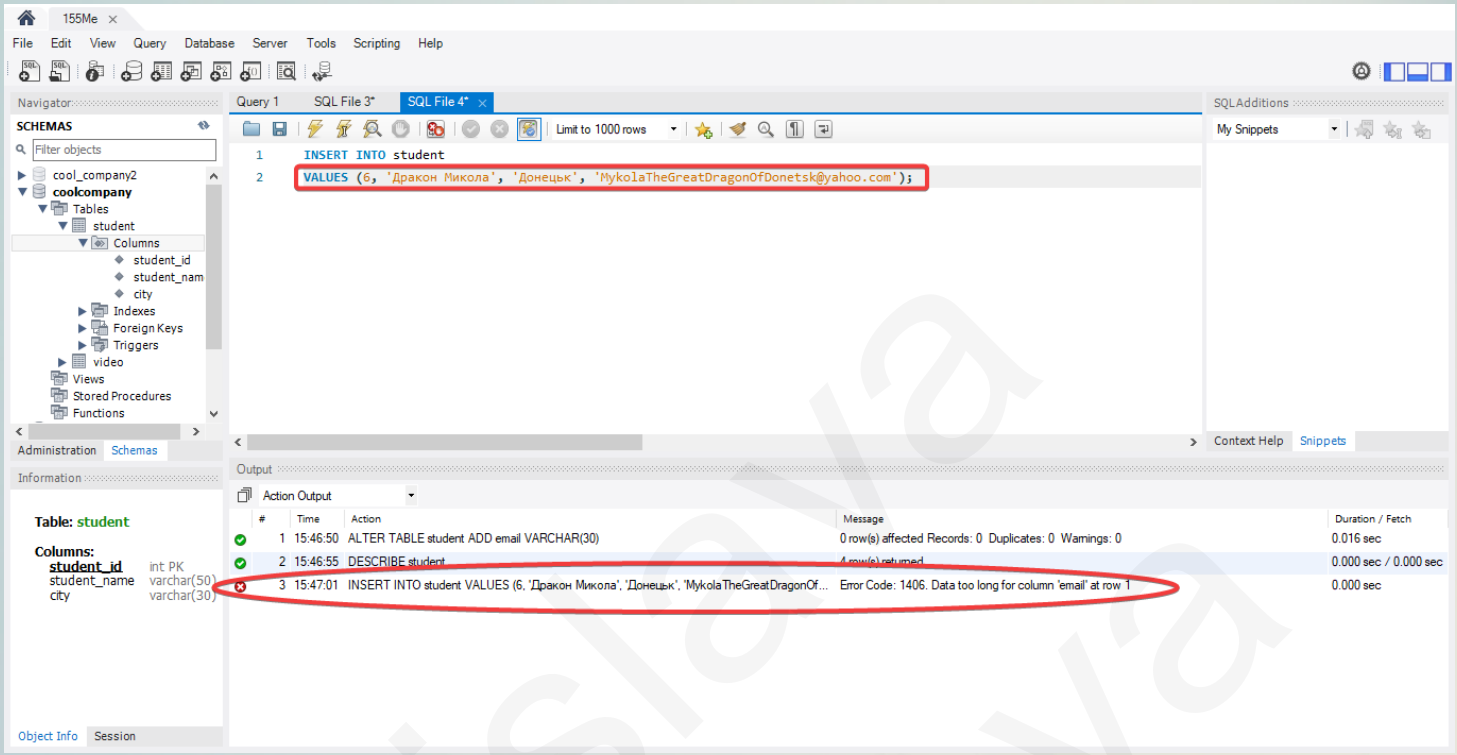


4.2 Delete the email column from the Student table.

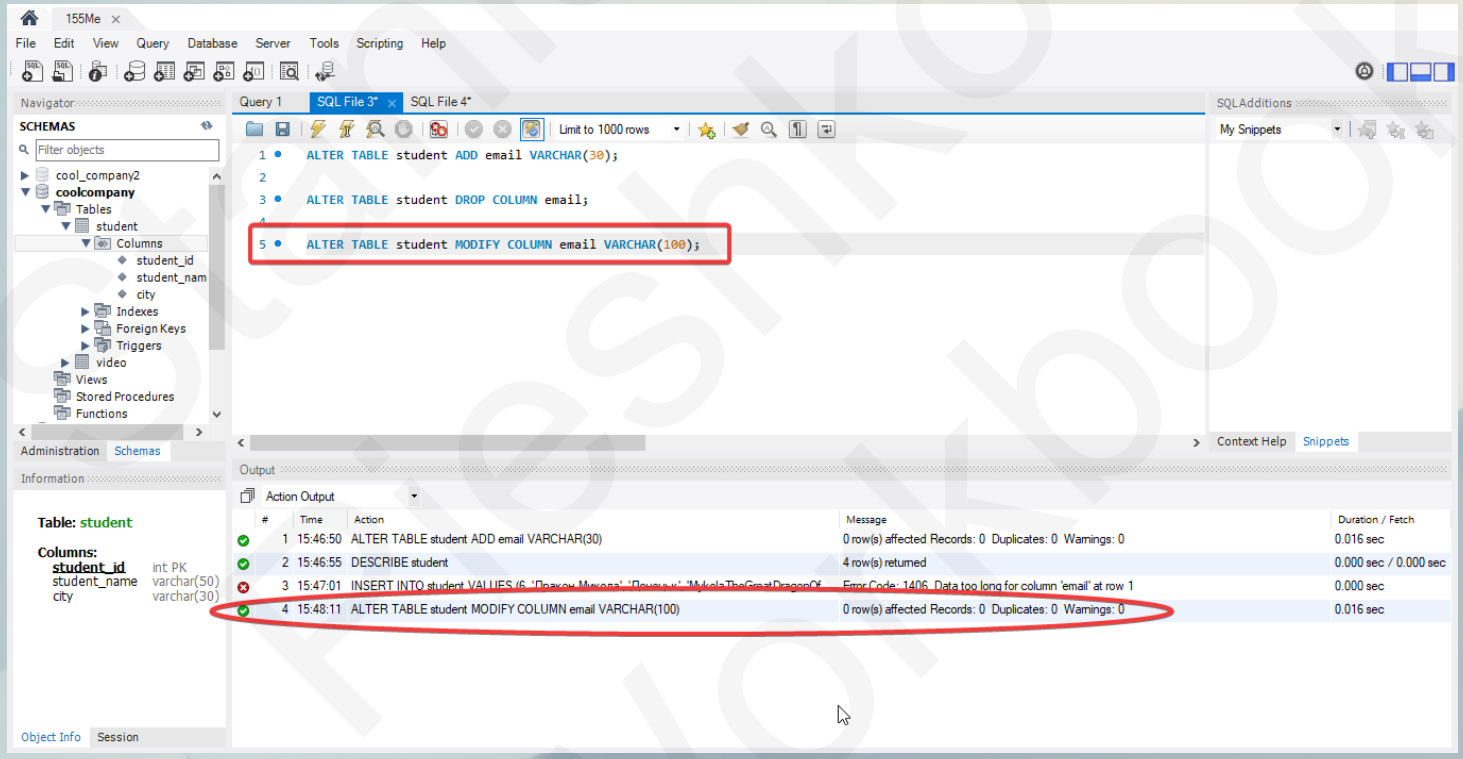


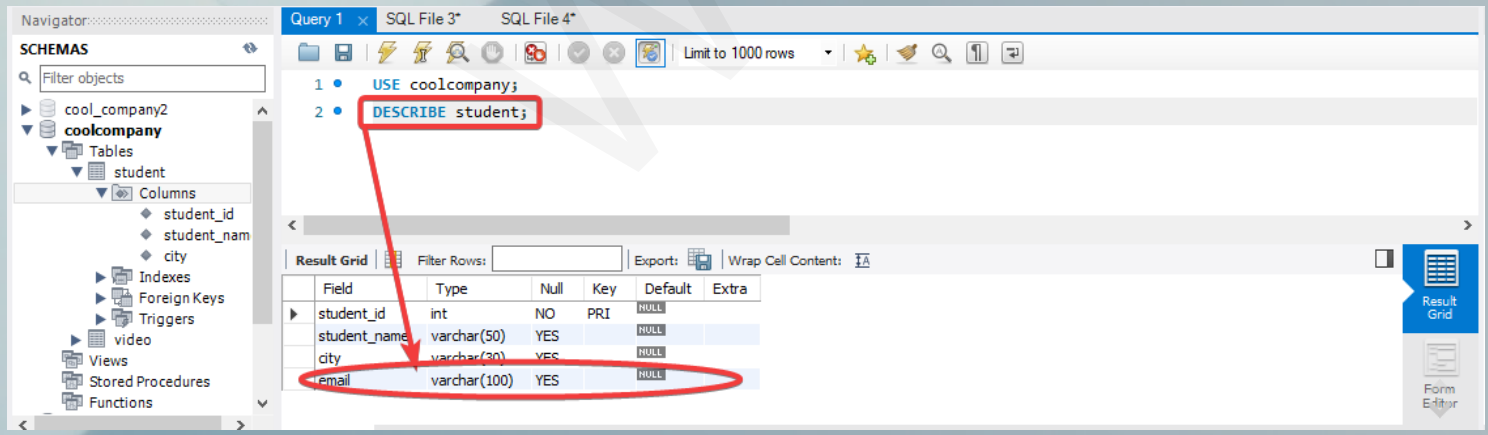


* Attempting to enter more than 30 characters in an email column.

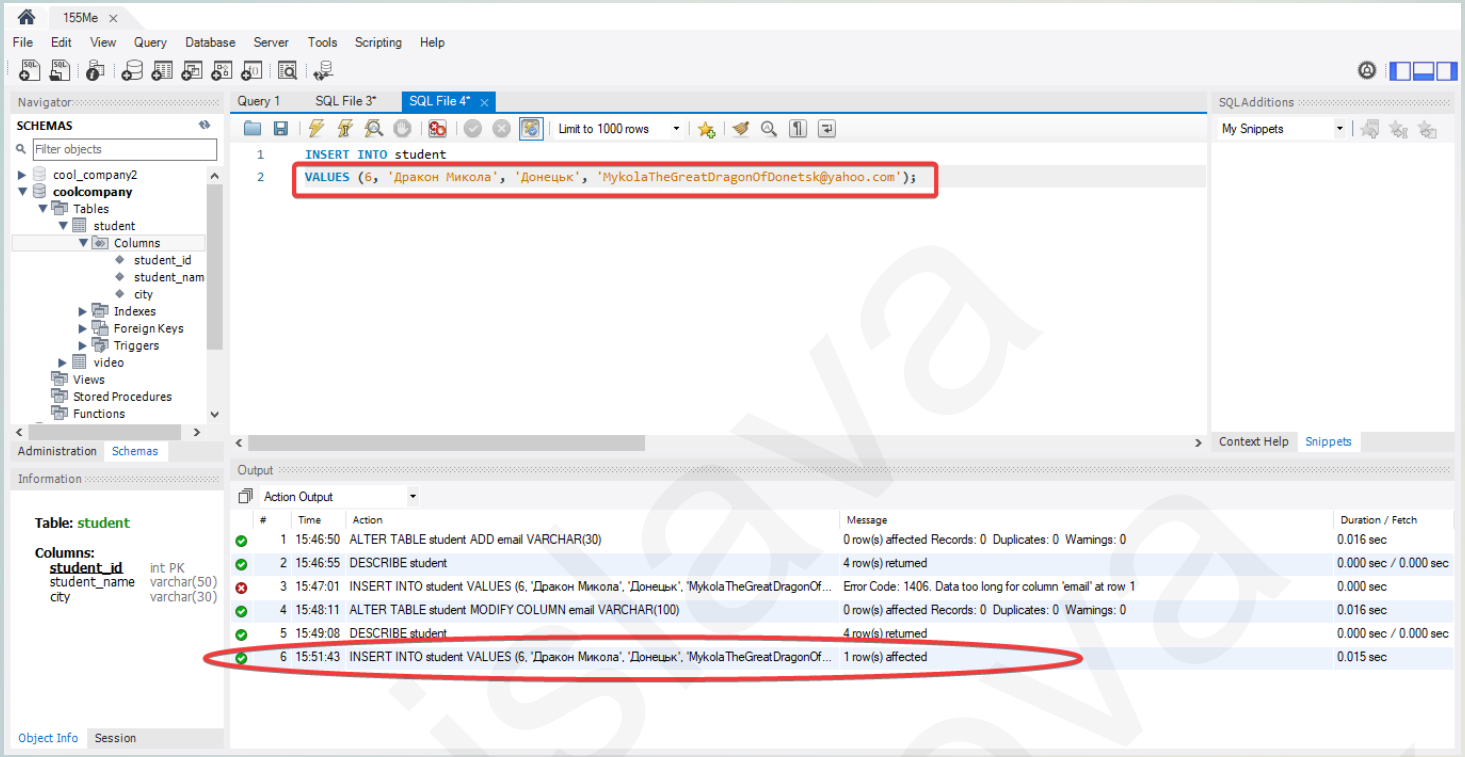


4.3 Change the size of characters that can be contained in the email column.

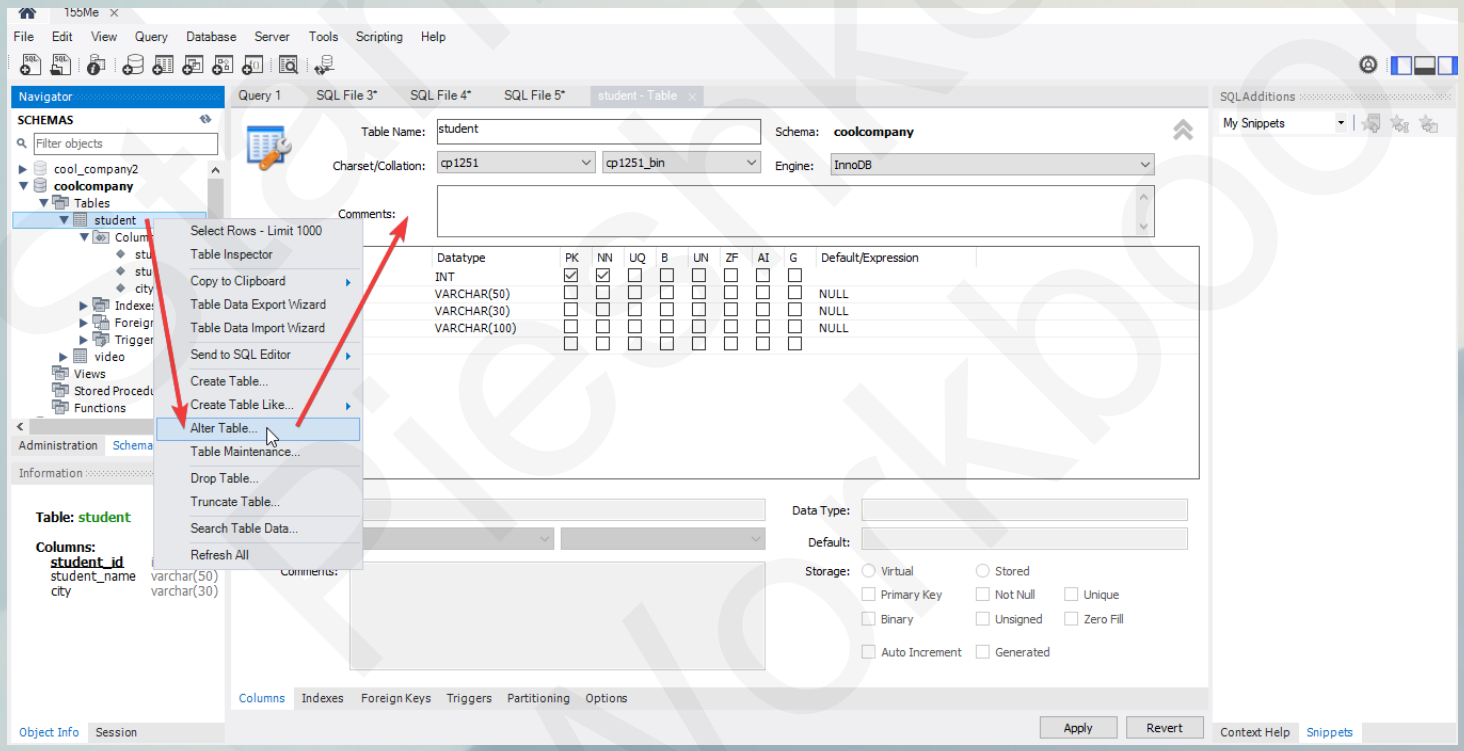




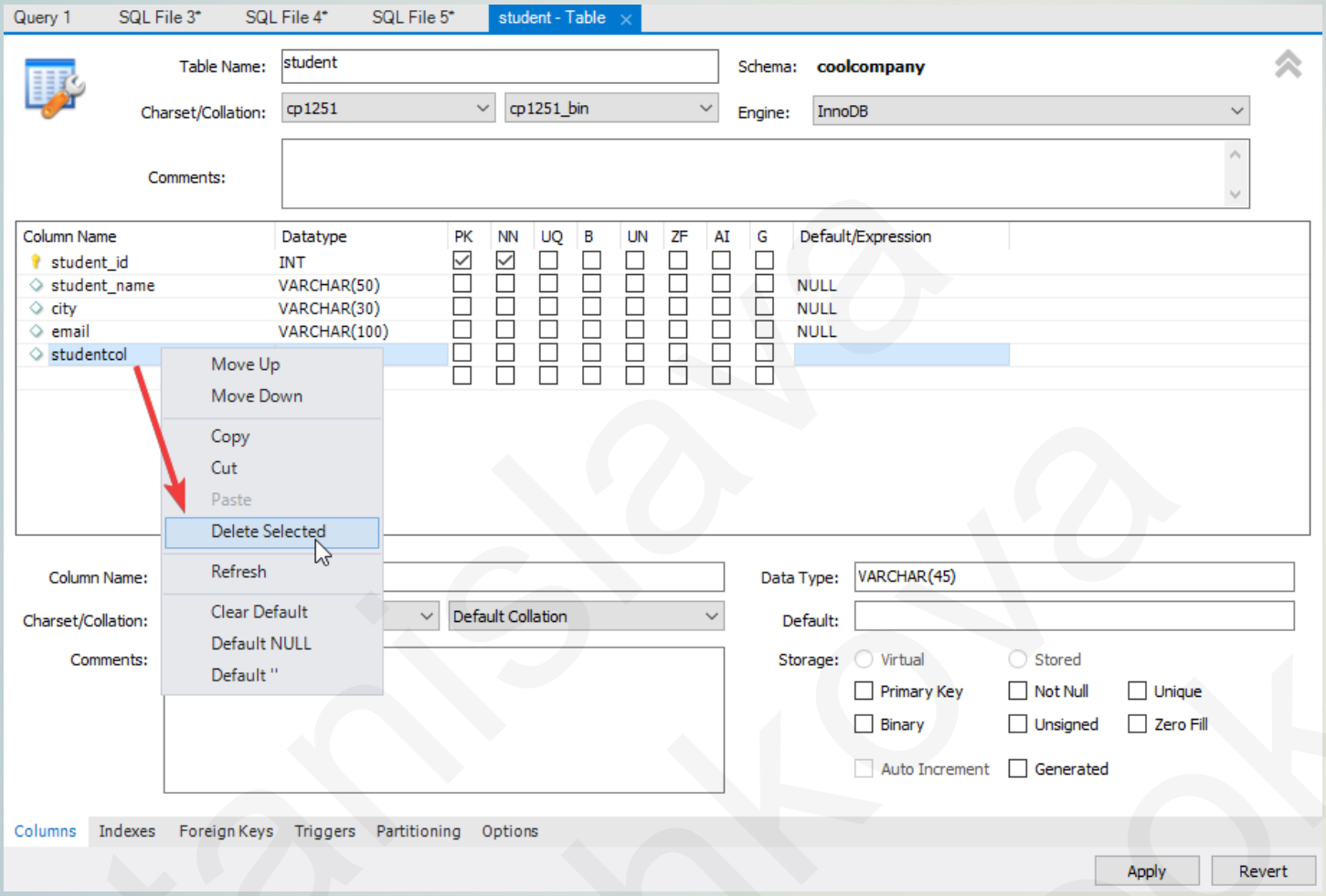
** Attempting to enter more than 30 characters in an email column after expanding the column to 100 characters.



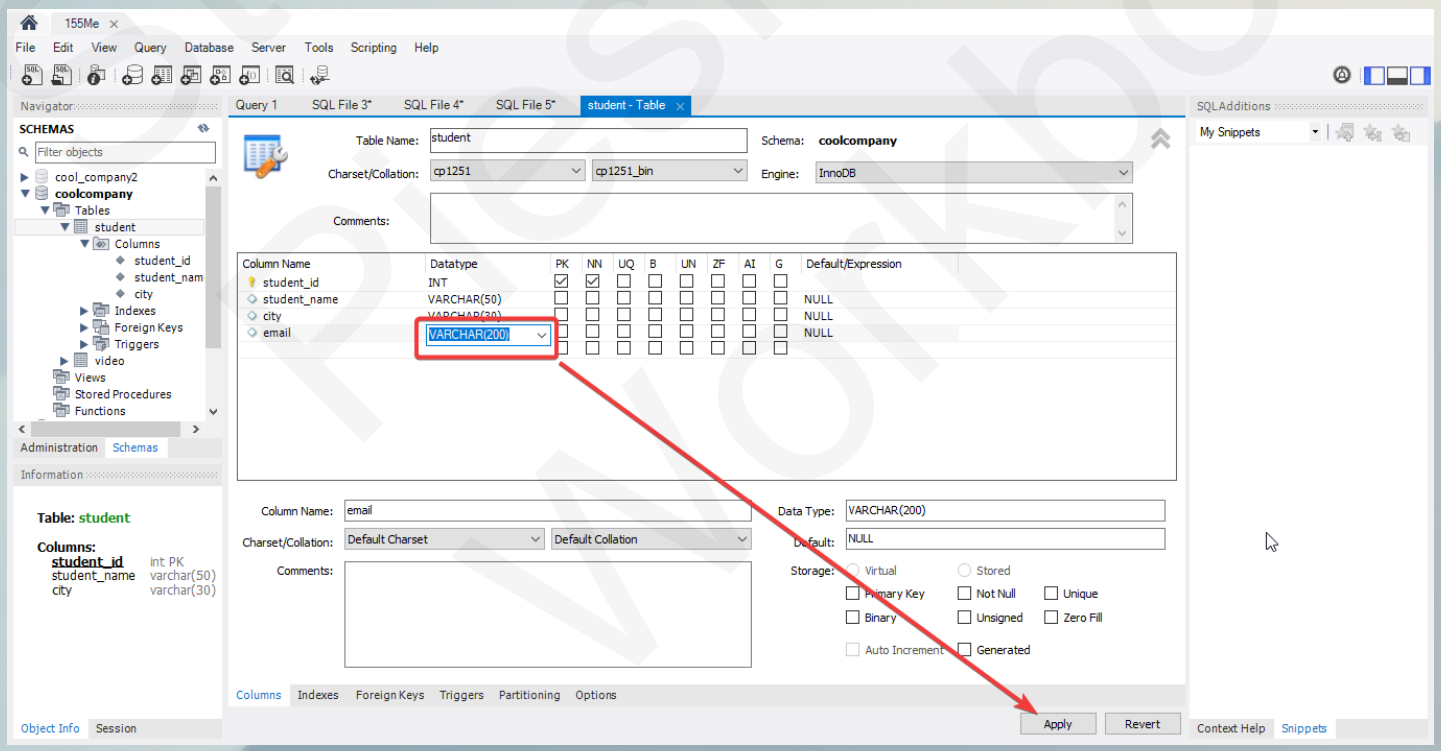
4.4 Edit a table using MySQL graphical tools

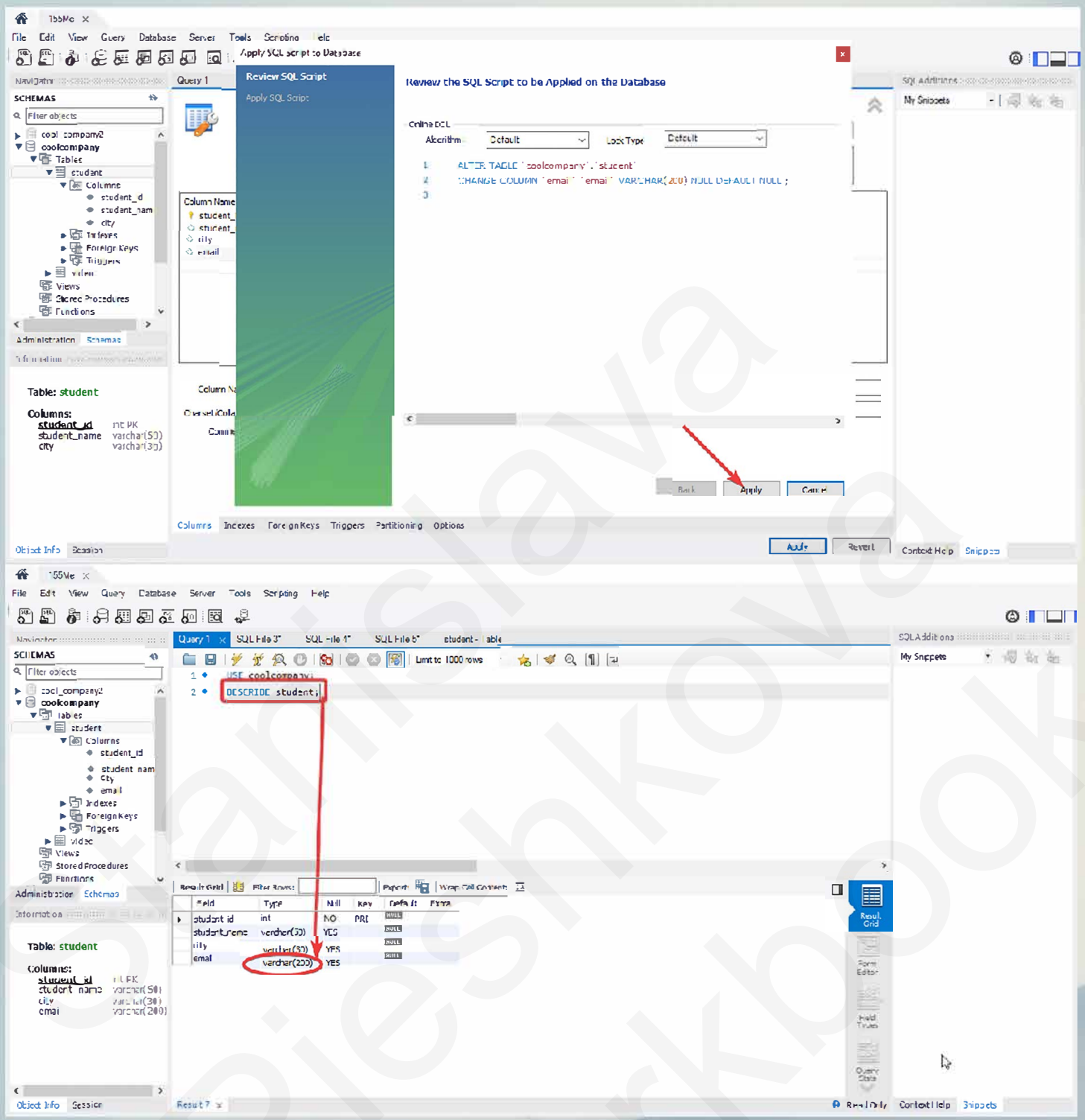


4.5 Delete the column you just added



4.6 Change the size of characters that can be contained in the email column from 100 to 200.

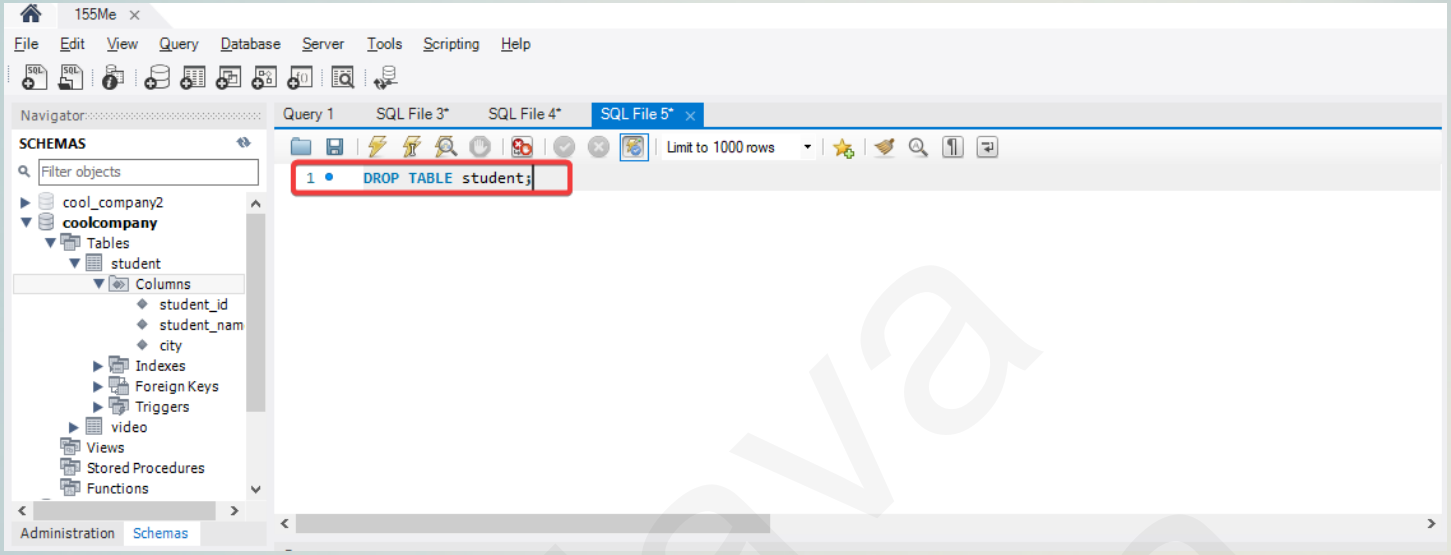



Apply SQL script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
Online DDL
Algorithm
Default
Lock Type
Default
ALTER TABLE `coolcompany`.`student`
CHANGE COLUMN `email` `email` VARCHAR(200) NULL DEFAULT NULL ;
Apply
Cancel
Table: student
Columns:
student_id int PK
student_name varchar(50)
city varchar(30)
Columns
Indexes
Foreign Keys
Triggers
Partitioning
Options
Apply
Revert

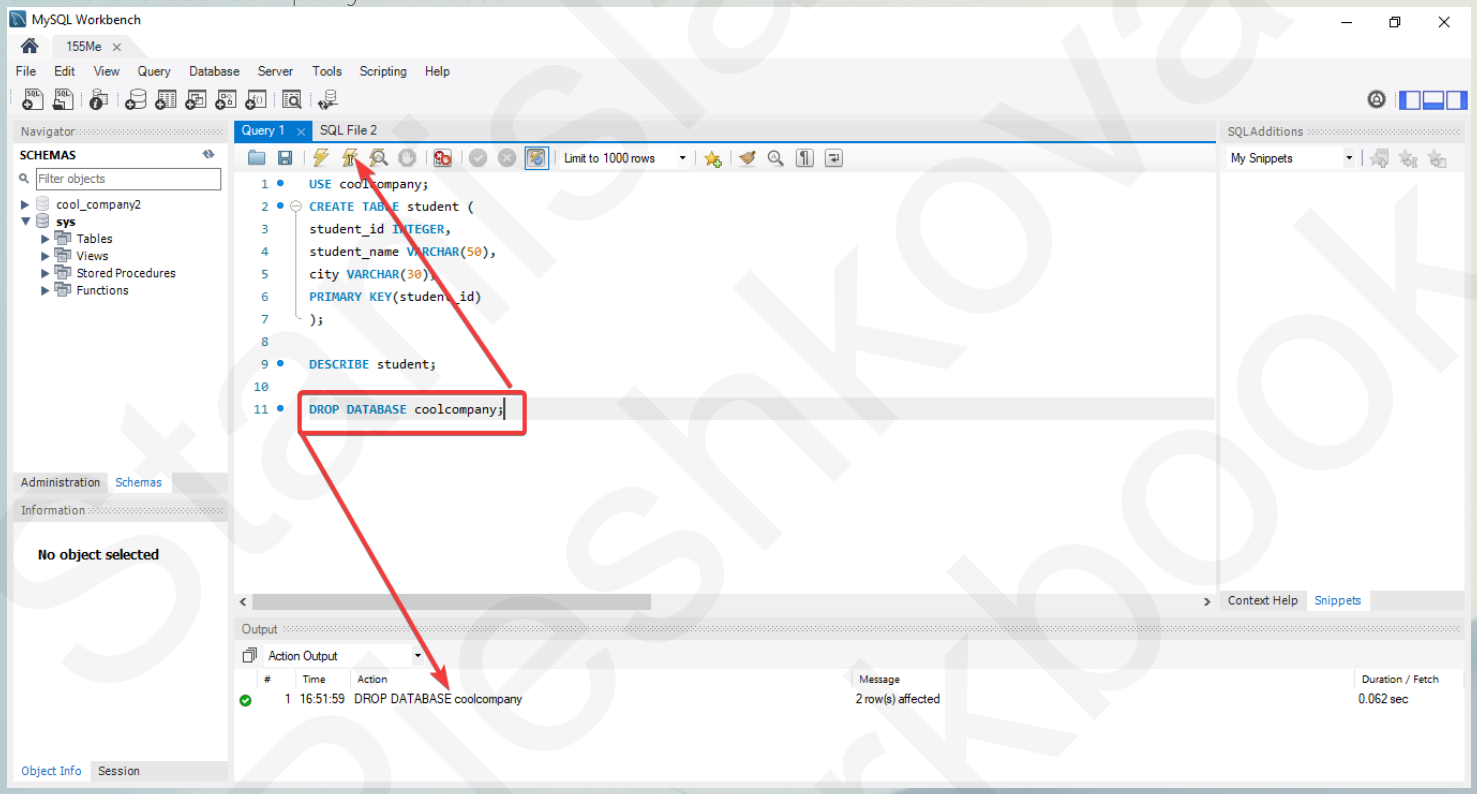
USE coolcompany;
DESCRIBE student;
Table: student
Columns:
student_id int PK
student_name varchar(50)
city varchar(30)
email varchar(200)
Field
Type
Null
Key
Default
Extra
student_id
int
NO
PRI
student_name
varchar(50)
YES
city
varchar(30)
YES
email
varchar(200)
YES

5. Delete a table and database

5.1 Delete the student table from the database.

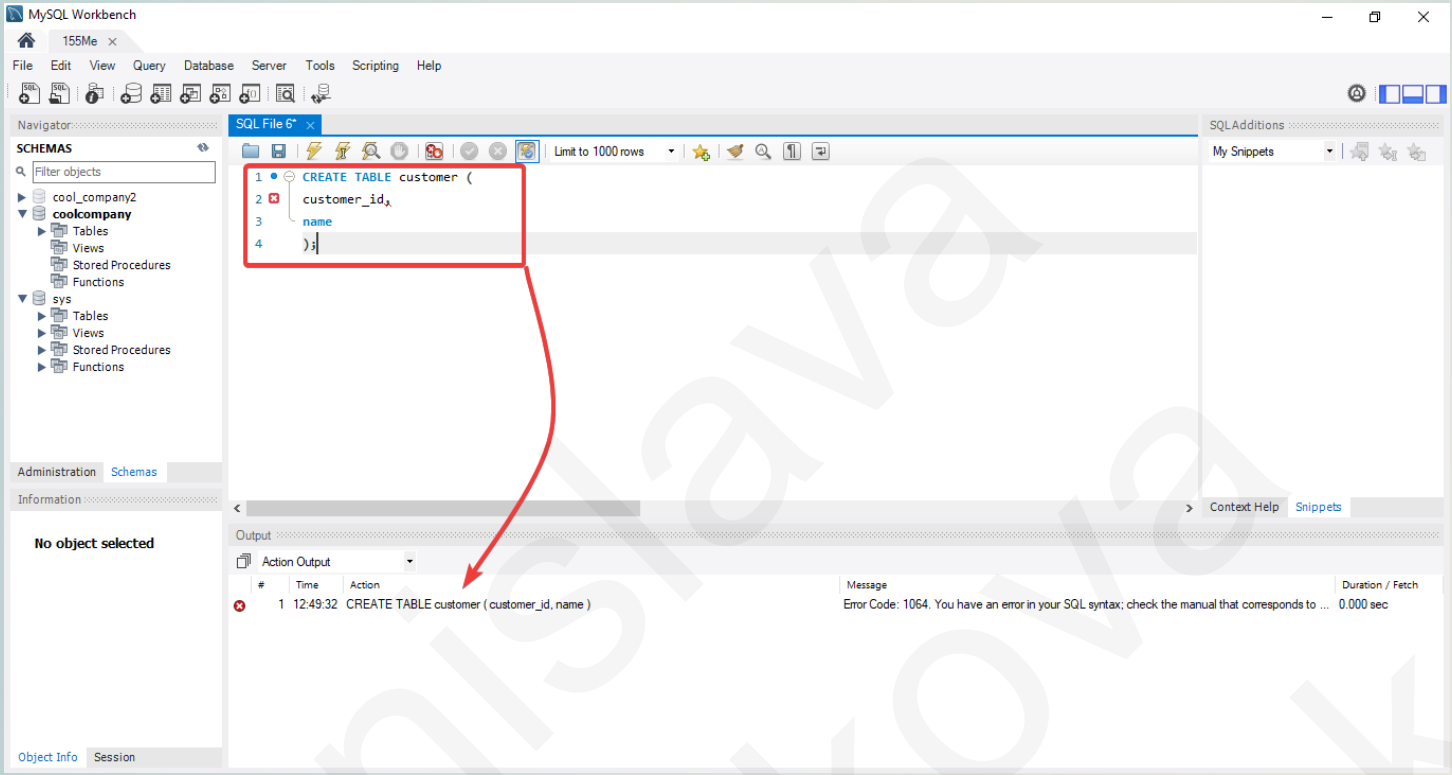


5.2 Delete coolcompany database.



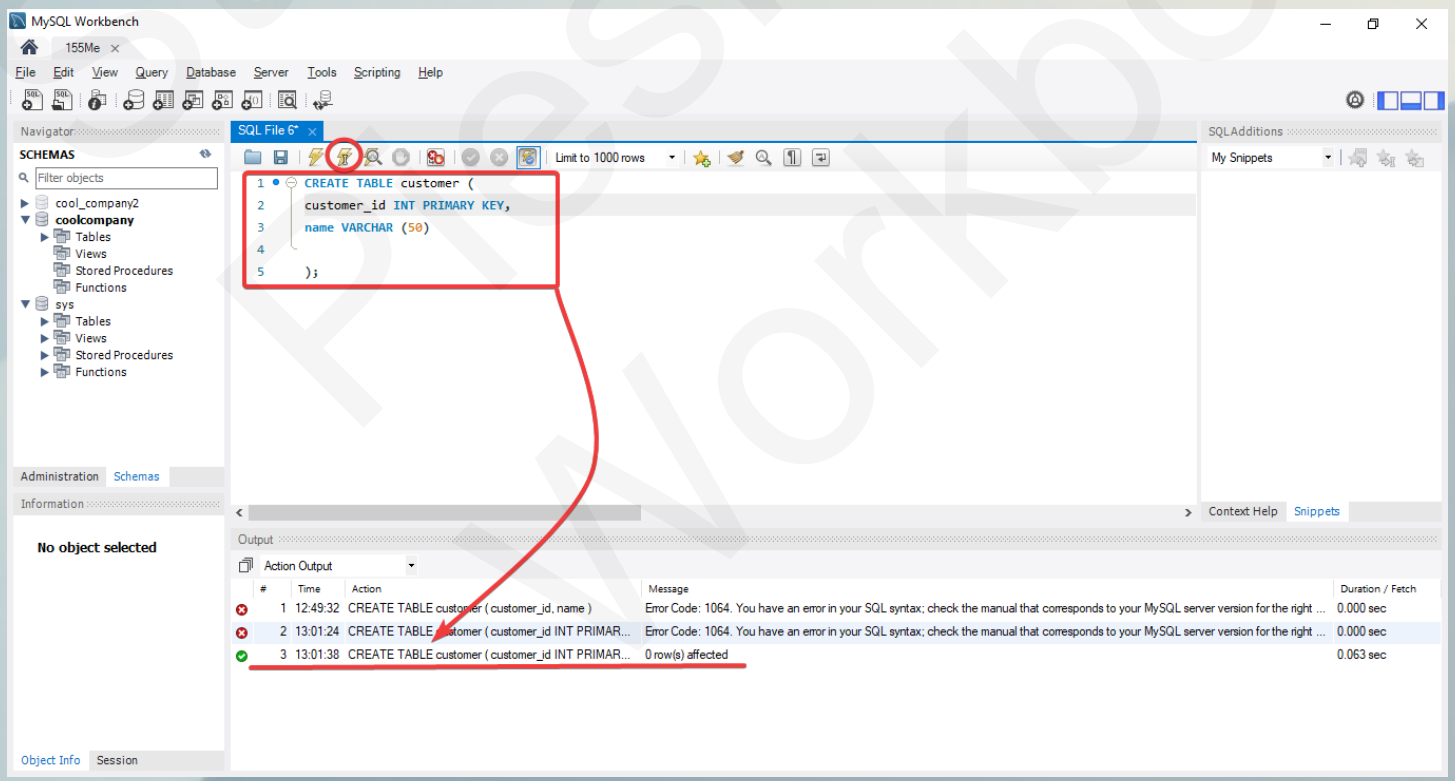
6. Working with data types

6.1Create a table without specifying a data type.



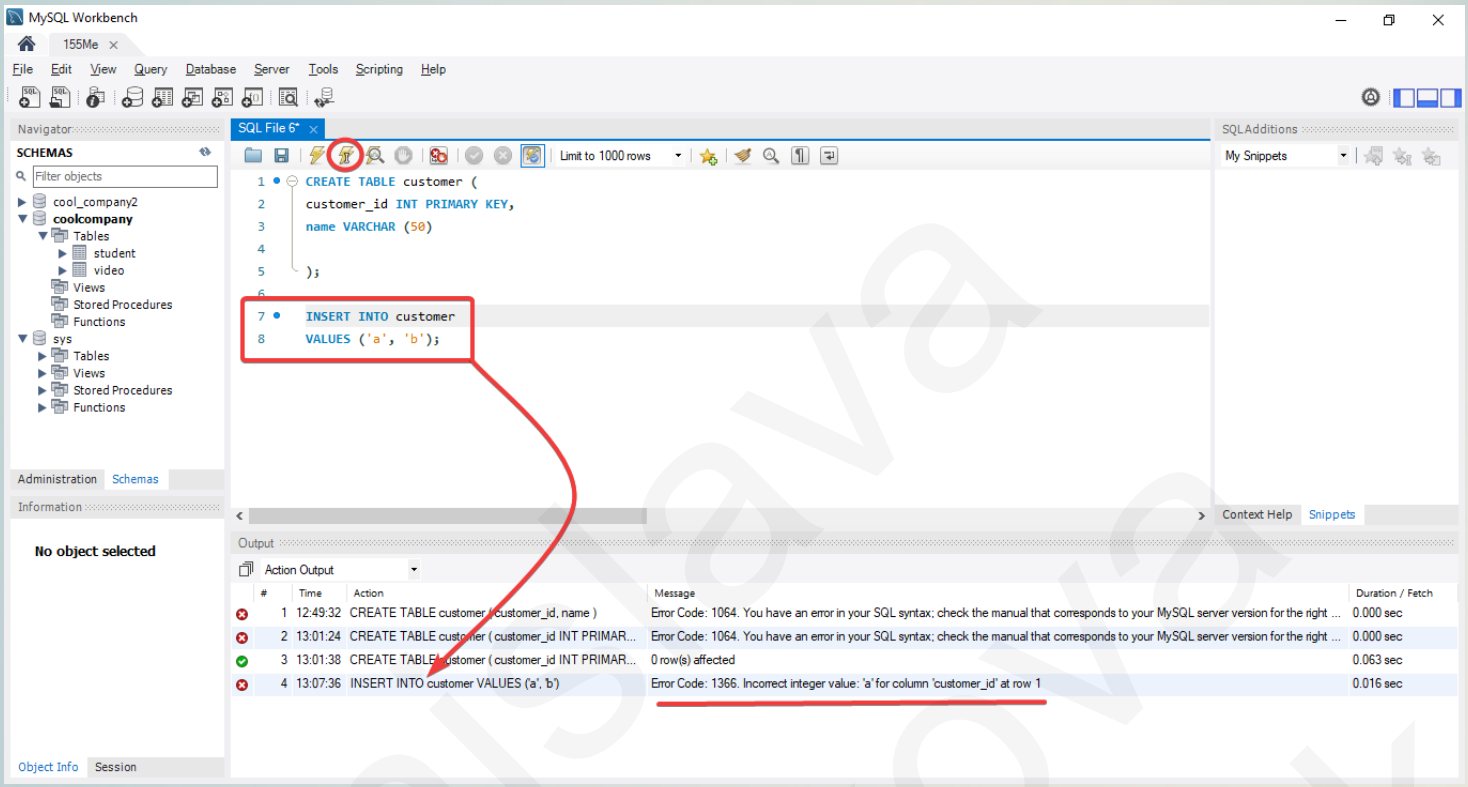
As expected, the system generates an error, because you cannot create a column without specifying the data type of each of them.

6.2 Specify the data types of each column and create a table.



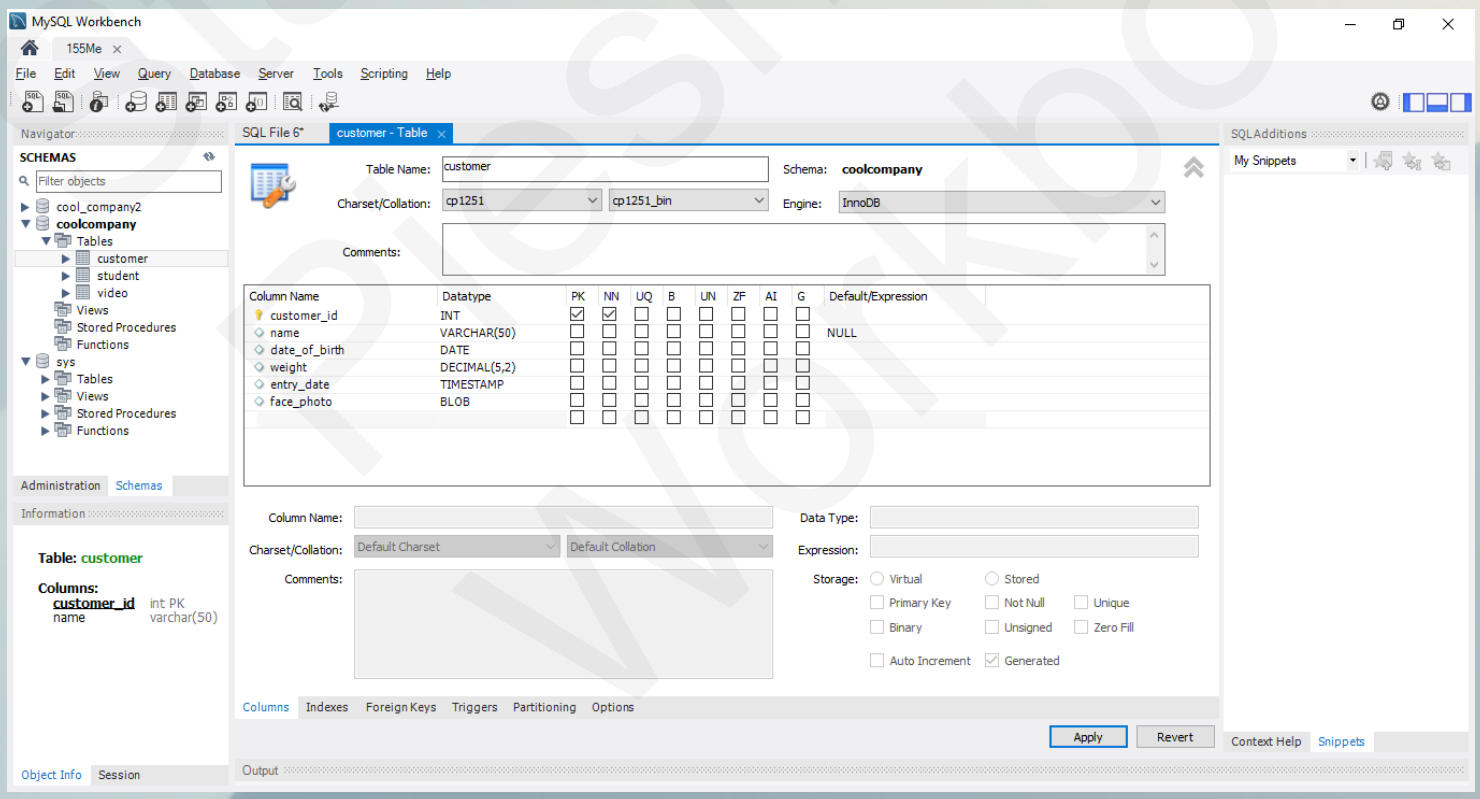
The table was created successfully without errors.

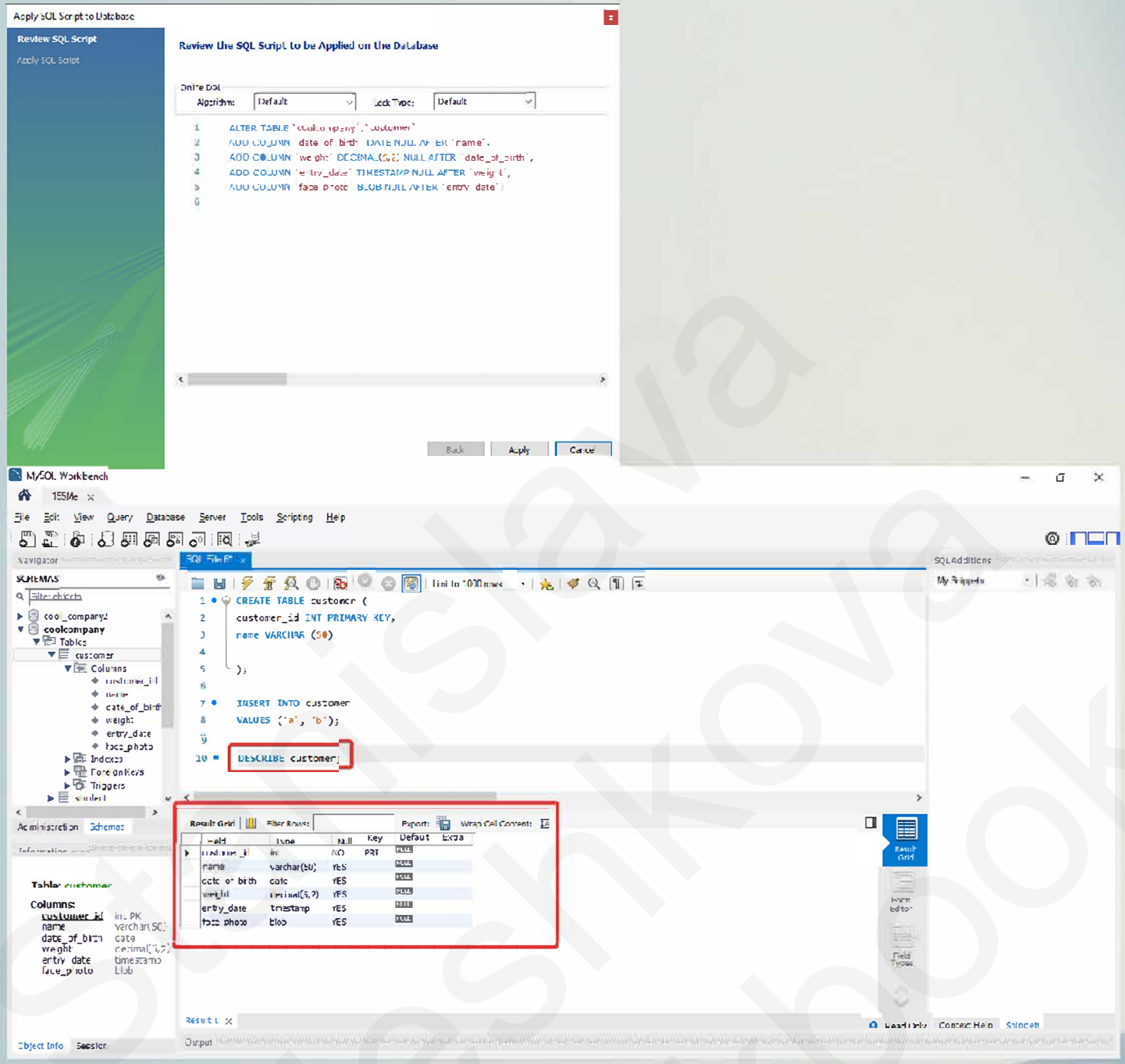
6.3 Add data to the table that does not fit its type.



We get the expected error. After all, you can't insert a text value into a column with an ID (INT type).

6.4 Fill in the table with the appropriate data types. To speed up this process, we use the visual components of MySQL Workbench.




Apply SQL Script to Database
Review SQL Script
Apply SQL Script
Review the SQL Script to be Applied on the Database
Online DDL
Algorithm:
Default
Lock Type:
Default
1 ALTER TABLE `coolcompany`.`customer`
2 ADD COLUMN `date_of_birth` DATE NULL AFTER `name`,
3 ADD COLUMN `weight` DECIMAL(5,2) NULL AFTER `date_of_birth`,
4 ADD COLUMN `entry_date` TIMESTAMP NULL AFTER `weight`,
5 ADD COLUMN `face_photo` BLOB NULL AFTER `entry_date`;
6
Back
Apply
Cancel
MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
cool_company2
coolcompany
Tables
customer
Columns
customer_id
name
date_of_birth
weight
entry_date
face_photo
Indexes
Foreign Keys
Triggers
Administration
Schemas
Table: customer
Columns:
customer_id int PK
name varchar(50)
date_of_birth date
weight decimal(5,2)
entry_date timestamp
face_photo blob
SQL File 8*
1 CREATE TABLE customer (
2 customer_id INT PRIMARY KEY,
3 name VARCHAR (50)
4
5);
6
7 INSERT INTO customer
8 VALUES ('a', 'b');
9
10 DESCRIBE customer;
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
Field
Type
Null
Key
Default
Extra
customer_id
int
NO
PRI
NULL
name
varchar(50)
YES
NULL
date_of_birth
date
YES
NULL
weight
decimal(5,2)
YES
NULL
entry_date
timestamp
YES
NULL
face_photo
blob
YES
NULL
SQL Additions
My Snippets
Result 1
Object Info
Session
Output
Read Only
Context Help
Snippets