**Distinction Level Project: Class Enrolment Data Storage**

**COS20015 – Fundamentals of Data Management**

Student Name: Cong Duc Danh (Dan) Ngo

Student ID: 104186810

Lab session: Wednesday 10:30 AM BA407

1. **Introduction**

* In this project, I have created a Relational Database named “Classes”, inspired by the timetable in Swinburne University of Technology; the database contains 7 entities which store different multiple information about a class session that a student can possibly have:

+ Convener: Stores personal information about a unit convener.

+ Tutor: Stores personal information about a tutor.

+ Student: Stores personal information about a student.

+ Room: Stores information about a teaching room at Swinburne.

+ Unit: Is a child table of Convener table, stores information about a unit at Swinburne.

+ Class: Is a child table of Unit, Room and Tutor entities, containing information about a particular class at Swinburne.

+ Sessions: Is the weakest entity of the database, a child table of Class and Student tables, storing the enrolment of students into classes at Swinburne.

* Entities relationships:

+ Convener – Unit: 1 to Many

+ Unit – Convener: 1 to 1

+ (Unit, Room, Tutor) – Class: 1 to Many

+ Class – (Unit, Room, Tutor): 1 to 1

+ (Class, Student) – Sessions: 1 to Many

+ Sessions – (Class, Student): 1 to 1

1. **UML Entity Relationship Diagram**

**A screenshot of a computer

Description automatically generated**

Figure 1: UML Relationship Diagram

1. **Normalisation Design**

* All tables having no repeating groups and the attributes are atomic -> the design is in 1NF.
* No table has partial dependencies among non-primary attributes because 6 tables have 1 surrogate primary key in each and the Sessions table has no non-primary attributes -> the design is in 2NF.
* There are no transitive dependencies in any table as well due to most of them only have 1 surrogate key and 1 table contains only composite keys -> the design is in 3NF.

1. **Database implementation**
2. **Creating tables**

CREATE DATABASE Distinction;

USE Distinction;

CREATE TABLE IF NOT EXISTS Convener

(

Convener\_ID CHAR(7) NOT NULL,

First\_name VARCHAR(15) NOT NULL,

Last\_name VARCHAR(15) NOT NULL,

Nationality VARCHAR(15) NOT NULL,

Gender VARCHAR(6) NOT NULL,

DOB DATE NOT NULL,

Email CHAR(19) NOT NULL,

Phone\_number CHAR(10) NOT NULL,

Highest\_qualification VARCHAR(50),

PRIMARY KEY(Convener\_ID)

);

CREATE TABLE IF NOT EXISTS Unit

(

Unit\_ID CHAR(8) NOT NULL,

Convener\_ID CHAR(7) NOT NULL,

Unit\_name VARCHAR(50) NOT NULL,

PRIMARY KEY(Unit\_ID),

FOREIGN KEY(Convener\_ID) REFERENCES Convener(Convener\_ID)

);

CREATE TABLE IF NOT EXISTS Tutor

(

Tutor\_ID CHAR(7) NOT NULL,

First\_name VARCHAR(15) NOT NULL,

Last\_name VARCHAR(15) NOT NULL,

Nationality VARCHAR(15) NOT NULL,

Gender VARCHAR(6) NOT NULL,

DOB DATE NOT NULL,

Email CHAR(19) NOT NULL,

Phone\_number CHAR(10) NOT NULL,

Highest\_qualification VARCHAR(50),

PRIMARY KEY(Tutor\_ID)

);

CREATE TABLE IF NOT EXISTS Room

(

Room\_ID VARCHAR(8) NOT NULL,

Capacity SMALLINT UNSIGNED NOT NULL,

PRIMARY KEY(Room\_ID)

);

CREATE TABLE IF NOT EXISTS Class

(

Class\_ID CHAR(13) NOT NULL,

Unit\_ID CHAR(8) NOT NULL,

Tutor\_ID CHAR(7) NOT NULL,

Room\_ID VARCHAR(8) NOT NULL,

Start\_time CHAR(12) NOT NULL,

End\_time CHAR(12) NOT NULL,

PRIMARY KEY(Class\_ID),

FOREIGN KEY(Unit\_ID) REFERENCES Unit(Unit\_ID),

FOREIGN KEY(Tutor\_ID) REFERENCES Tutor(Tutor\_ID),

FOREIGN KEY(Room\_ID) REFERENCES Room(Room\_ID)

);

CREATE TABLE IF NOT EXISTS Student

(

Student\_ID CHAR(9) NOT NULL,

First\_name VARCHAR(15) NOT NULL,

Last\_name VARCHAR(15) NOT NULL,

Nationality VARCHAR(15) NOT NULL,

Gender VARCHAR(6) NOT NULL,

DOB DATE NOT NULL,

Email CHAR(29) NOT NULL,

Phone\_number CHAR(10) NOT NULL,

Highest\_qualification VARCHAR(50) NOT NULL,

PRIMARY KEY(Student\_ID)

);

CREATE TABLE IF NOT EXISTS Sessions

(

Class\_ID CHAR(13) NOT NULL,

Student\_ID CHAR(9) NOT NULL,

PRIMARY KEY(Class\_ID, Student\_ID),

FOREIGN KEY(Class\_ID) REFERENCES Class(Class\_ID),

FOREIGN KEY(Student\_ID) REFERENCES Student(Student\_ID)

);

1. **Inserting Data Records**

INSERT INTO Convener(Convener\_ID, First\_name, Last\_name, Nationality, Gender, DOB, Email, Phone\_number, Highest\_qualification)

VALUES

                    ('1876543', "Henry", "McCartney", "Australian", "Male", "1967-06-05", "1876543@swin.edu.au", "0414563698", "PhD of Computer Engineering"),

                    ('1245678', "Emma", "Vega", "Australian", "Female", "1972-02-04", "1245678@swin.edu.au", "0444569987", "PhD of Software Engineering")

;

INSERT INTO Unit(Unit\_ID, Convener\_ID, Unit\_name)

VALUES

                ("COS20007", '1876543', "Object Oriented Programming"),

                ("SWE30009", '1876543', "Software Testing and Reliability"),

                ("TNE30023", '1245678', "Advanced Switching"),

                ("COS20015", '1245678', "Fundamentals of Data Management")

;

INSERT INTO Room(Room\_ID, Capacity)

VALUES

                ("ATC405", 30),

                ("BA502", 24),

                ("AMDC301", 35),

                ("EN310", 30),

                ("ATC101", 300)

;

INSERT INTO Tutor(Tutor\_ID, First\_name, Last\_name, Nationality, Gender, DOB, Email, Phone\_number, Highest\_qualification)

VALUES

                ('1056321', "John", "Clark", "Australian", "Male", "1985-05-05", "1256321@swin.edu.au", "0456365213", "Master of Data Science"),

                ('1034567', "Linda", "Li", "Chinese", "Female", "1995-06-23", "1034567@swin.edu.au", "0435217896", "PhD of Information System"),

                ('1012345', "Michael", "Martin", "American", "Other", "1992-02-24", "1012345@swin.edu.au", "0416596324", "PhD of ICT"),

                ('1098765', "Ishaan", "Patel", 'Indian', 'Male', '2000-09-29', '1098765@swin.edu.au', '0425369852', 'PhD of Computer Science'),

                ('1456721', 'Grace', 'Thompson', 'Australian', 'Female', '1976-12-02', '1456721@swin.edu.au', '0415632025', 'Master of Cyber Security')

;

INSERT INTO Student(Student\_ID, First\_name, Last\_name, Nationality, Gender, DOB, Email, Phone\_number, Highest\_qualification)

VALUES

                ('104563214', "Kevin", "Nguyen", "Vietnamese", "Male", "2002-03-14", "104563214@student.swin.edu.au", "0465369852", "High School Diploma"),

                ('125463978', "Adam", "West", "Australian", "Male", "2004-06-26", "125463978@student.swin.edu.au", "0423654123", "Foundation"),

                ('102365975', "Jane", "Kelly", "Australian", "Other", "2003-11-06", "102365975@student.swin.edu.au", "0425369963", "Bachelor of Computer Science"),

                ('123068466', "Sofia", "Garcia", 'Spanish', 'Female', '2004-03-26', '123068466@student.swin.edu.au', '0432653213', 'High School Diploma'),

                ('168306833', 'Ibrahim', 'Abara', 'Nigerian', 'Male', '2003-12-06', '168306833@student.swin.edu.au', '0425556321', 'Vocational Education')

;

INSERT INTO Class (Class\_ID, Unit\_ID, Tutor\_ID, Room\_ID, Start\_time, End\_time)

VALUES

    ('COS20007LEC01', 'COS20007', '1098765', 'ATC101', 'Mon 09:00 AM', 'Mon 10:30 AM'),

    ('COS20007WOK01', 'COS20007', '1034567', 'BA502', 'Tue 10:00 AM', 'Tue 11:30 AM'),

    ('COS20007WOK02', 'COS20007', '1012345', 'AMDC301', 'Wed 01:00 PM', 'Wed 02:30 PM'),

    ('COS20007LAB01', 'COS20007', '1098765', 'BA502', 'Thu 09:30 AM', 'Thu 11:00 AM'),

    ('COS20007LAB02', 'COS20007', '1034567', 'EN310', 'Fri 11:00 AM', 'Fri 12:30 PM'),

    ('COS20007WOK03', 'COS20007', '1456721', 'ATC405', 'Mon 03:00 PM', 'Mon 04:30 PM'),

    ('COS20007LAB03', 'COS20007', '1034567', 'BA502', 'Tue 02:00 PM', 'Tue 03:30 PM'),

    ('COS20007LEC02', 'COS20007', '1056321', 'ATC101', 'Wed 09:00 AM', 'Wed 10:30 AM'),

    ('SWE30009LEC01', 'SWE30009', '1098765', 'ATC101', 'Tue 09:00 AM', 'Tue 10:30 AM'),

    ('SWE30009WOK01', 'SWE30009', '1034567', 'BA502', 'Wed 10:00 AM', 'Wed 11:30 AM'),

    ('SWE30009WOK02', 'SWE30009', '1012345', 'AMDC301', 'Thu 01:00 PM', 'Thu 02:30 PM'),

    ('SWE30009LAB01', 'SWE30009', '1098765', 'BA502', 'Fri 09:30 AM', 'Fri 11:00 AM'),

    ('SWE30009LAB02', 'SWE30009', '1034567', 'EN310', 'Mon 11:00 AM', 'Mon 12:30 PM'),

    ('SWE30009WOK03', 'SWE30009', '1456721', 'ATC405', 'Tue 03:00 PM', 'Tue 04:30 PM'),

    ('SWE30009LAB03', 'SWE30009', '1098765', 'AMDC301', 'Wed 02:00 PM', 'Wed 03:30 PM'),

    ('TNE30023LEC01', 'TNE30023', '1098765', 'ATC101', 'Thu 09:00 AM', 'Thu 10:30 AM'),

    ('TNE30023WOK01', 'TNE30023', '1034567', 'BA502', 'Fri 10:00 AM', 'Fri 11:30 AM'),

    ('TNE30023WOK02', 'TNE30023', '1012345', 'AMDC301', 'Mon 01:00 PM', 'Mon 02:30 PM'),

    ('TNE30023LAB01', 'TNE30023', '1098765', 'BA502', 'Tue 09:30 AM', 'Tue 11:00 AM'),

    ('TNE30023LAB02', 'TNE30023', '1034567', 'EN310', 'Wed 11:00 AM', 'Wed 12:30 PM'),

    ('TNE30023WOK03', 'TNE30023', '1456721', 'ATC405', 'Thu 03:00 PM', 'Thu 04:30 PM'),

    ('TNE30023LAB03', 'TNE30023', '1098765', 'AMDC301', 'Fri 02:00 PM', 'Fri 03:30 PM'),

    ('COS20015LEC01', 'COS20015', '1098765', 'ATC101', 'Mon 09:00 AM', 'Mon 10:30 AM'),

    ('COS20015WOK01', 'COS20015', '1034567', 'BA502', 'Tue 10:00 AM', 'Tue 11:30 AM'),

    ('COS20015WOK02', 'COS20015', '1012345', 'AMDC301', 'Wed 01:00 PM', 'Wed 02:30 PM'),

    ('COS20015LAB01', 'COS20015', '1098765', 'BA502', 'Thu 09:30 AM', 'Thu 11:00 AM'),

    ('COS20015LAB02', 'COS20015', '1034567', 'EN310', 'Fri 11:00 AM', 'Fri 12:30 PM'),

    ('COS20015WOK03', 'COS20015', '1456721', 'ATC405', 'Mon 03:00 PM', 'Mon 04:30 PM'),

    ('COS20015LAB03', 'COS20015', '1098765', 'AMDC301', 'Tue 02:00 PM', 'Tue 03:30 PM')

;

INSERT INTO Sessions (Class\_ID, Student\_ID)

VALUES

    ('COS20007LEC01', '104563214'),

    ('COS20007LEC01', '125463978'),

    ('COS20007WOK01', '102365975'),

    ('COS20007WOK01', '123068466'),

    ('COS20007WOK02', '168306833'),

    ('COS20007LAB01', '104563214'),

    ('COS20007LAB01', '102365975'),

    ('COS20007LAB02', '125463978'),

    ('COS20007WOK03', '123068466'),

    ('COS20007LAB03', '104563214'),

    ('SWE30009LEC01', '102365975'),

    ('SWE30009LEC01', '104563214'),

    ('SWE30009WOK01', '125463978'),

    ('SWE30009WOK01', '168306833'),

    ('SWE30009WOK02', '102365975'),

    ('SWE30009LAB01', '104563214'),

    ('SWE30009LAB02', '123068466'),

    ('SWE30009WOK03', '125463978'),

    ('TNE30023LEC01', '104563214'),

    ('TNE30023LEC01', '125463978'),

    ('TNE30023WOK01', '102365975'),

    ('TNE30023WOK01', '123068466'),

    ('TNE30023WOK02', '168306833'),

    ('TNE30023LAB01', '104563214'),

    ('TNE30023LAB02', '125463978'),

    ('TNE30023WOK03', '123068466'),

    ('COS20015LEC01', '102365975'),

    ('COS20015LEC01', '104563214'),

    ('COS20015WOK01', '125463978'),

    ('COS20015WOK01', '168306833')

;

1. **Results**

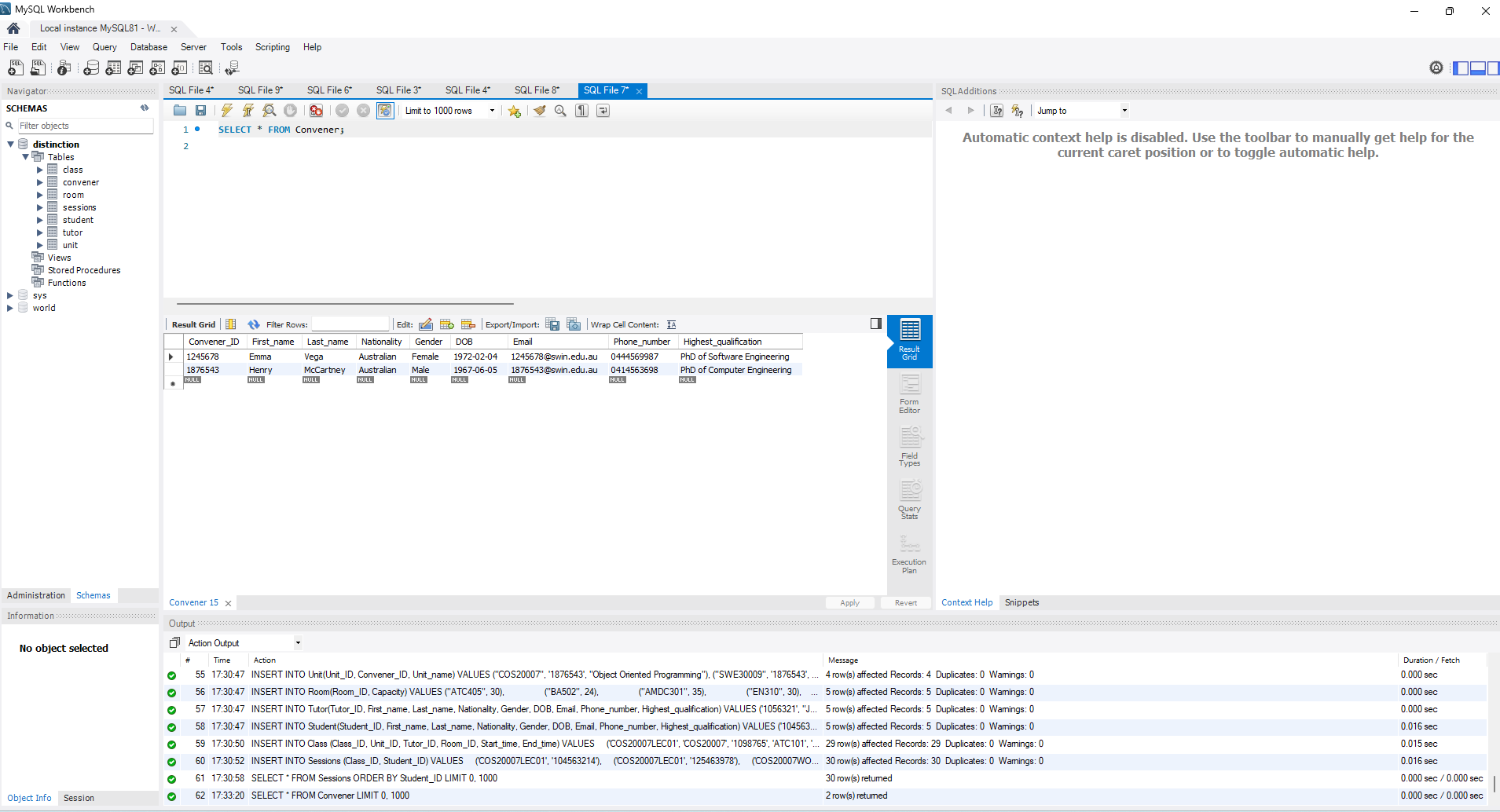


Figure 2: Convener table

A screenshot of a computer

Description automatically generated

Figure 3: Unit table

A screenshot of a computer

Description automatically generated

Figure 4: Room table

A screenshot of a computer

Description automatically generated

Figure 5: Tutor table

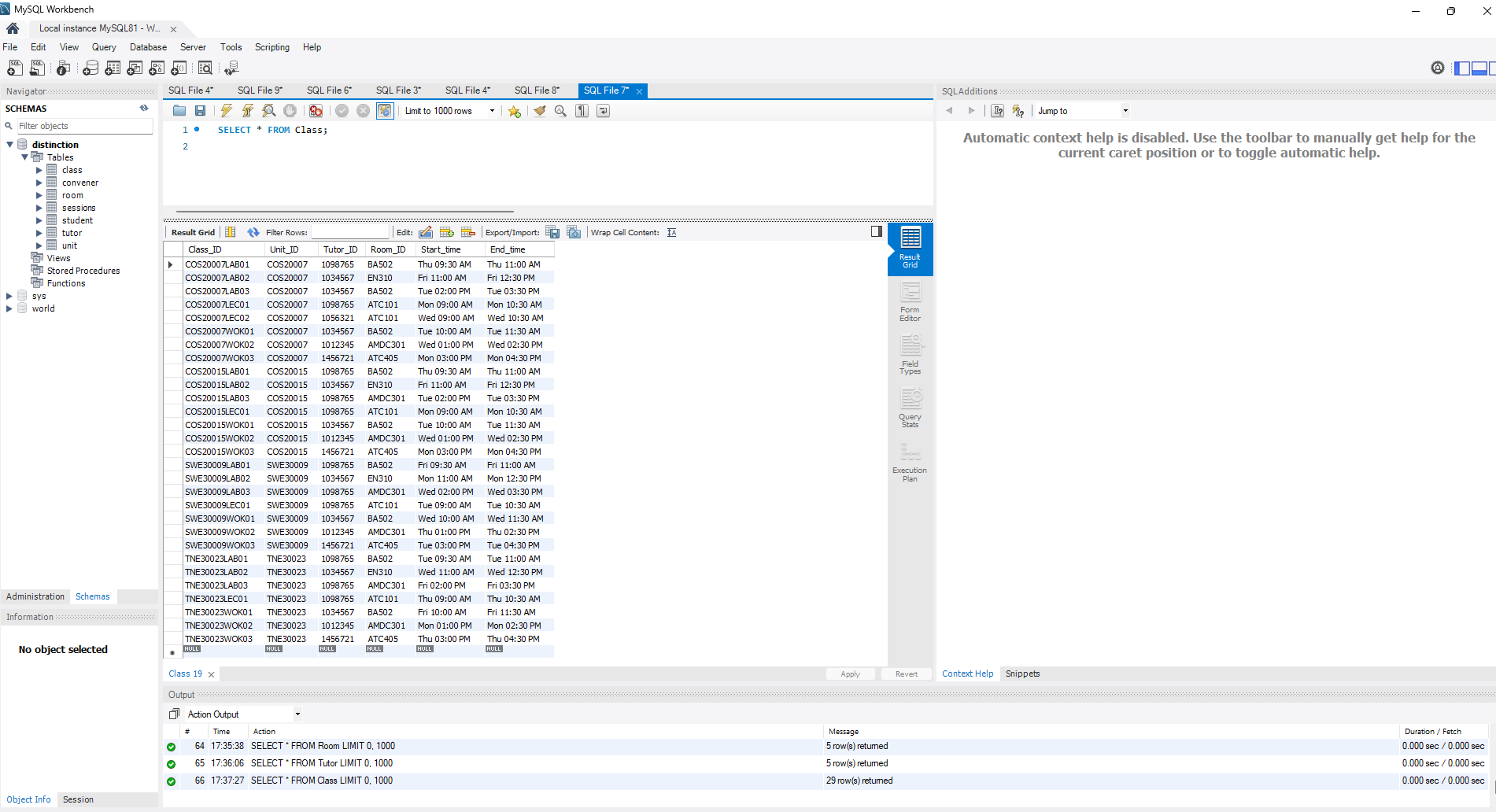


Figure 6: Class table

A screenshot of a computer

Description automatically generated

Figure 7: Student table

A screenshot of a computer

Description automatically generated

Figure 8: Sessions table

1. **5 queries using JOIN and GROUP operations.**
2. **Query 1:**

SELECT

    c.Class\_ID,

    c.Start\_time,

    c.End\_time,

    CONCAT(t.First\_name, ' ', t.Last\_name) AS Tutor\_Name,

    r.Room\_ID,

    r.Capacity,

    COUNT(s.Student\_ID) AS Students

FROM

    Class AS c

INNER JOIN Unit AS u ON c.Unit\_ID = u.Unit\_ID

INNER JOIN Tutor AS t ON c.Tutor\_ID = t.Tutor\_ID

INNER JOIN Room AS r ON c.Room\_ID = r.Room\_ID

LEFT JOIN Sessions AS s ON c.Class\_ID = s.Class\_ID

GROUP BY c.Class\_ID

HAVING Students != 0

ORDER BY c.Class\_ID;

**Result:**

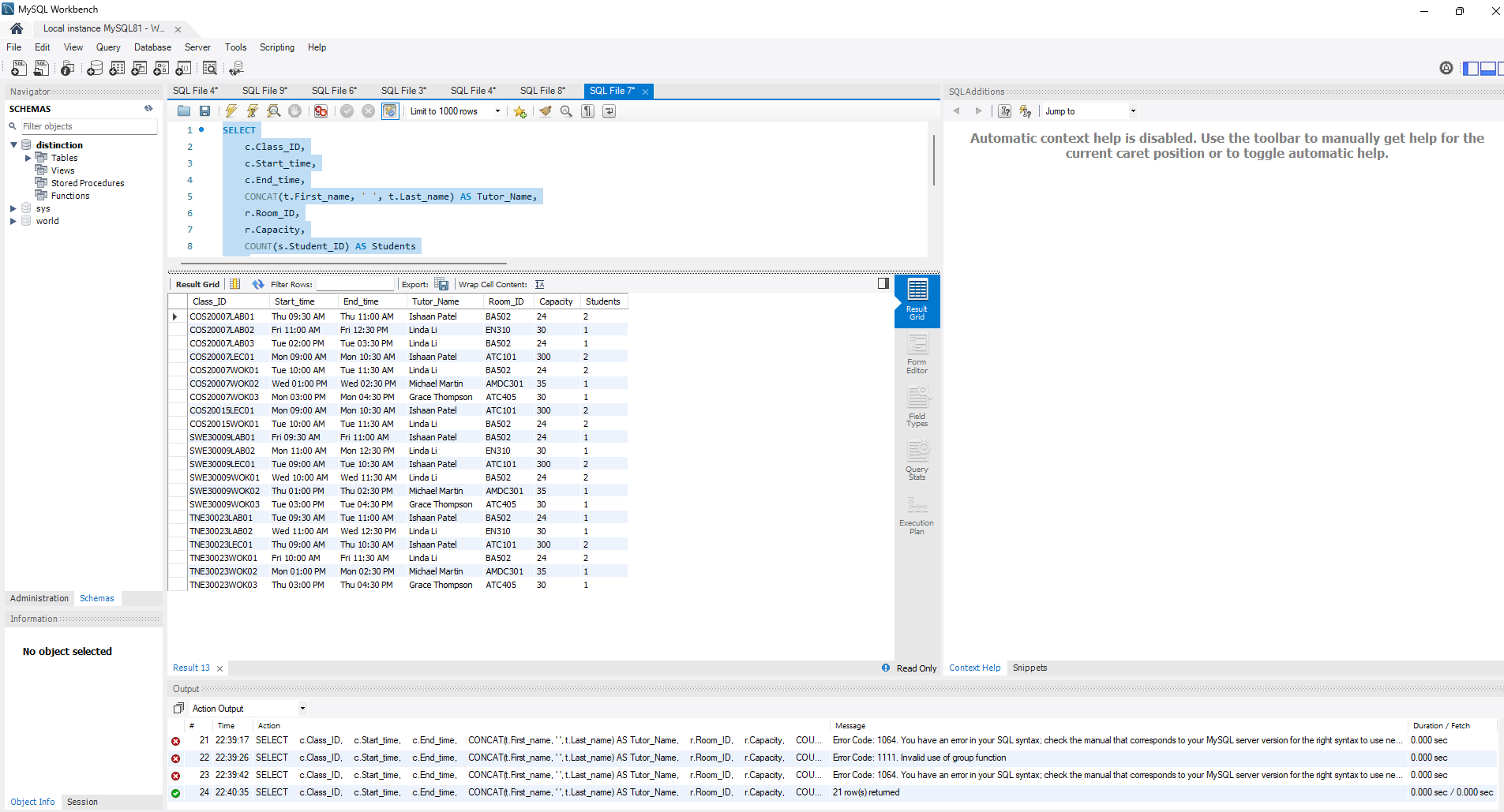
****

Figure 9: Query 1’s result

* This query returns detailed information about class sessions that have at least 1 student attending using LEFT JOIN.

1. **Query 2:**

SELECT

    c.Tutor\_ID,

    CONCAT(t.First\_name, ' ', t.Last\_name) AS Tutor\_Name,

    c.Class\_ID,

    c.Start\_time,

    c.End\_time,

    r.Room\_ID

FROM

    Class AS c

INNER JOIN Tutor AS t ON c.Tutor\_ID = t.Tutor\_ID

INNER JOIN Unit AS u ON c.Unit\_ID = u.Unit\_ID

INNER JOIN Room AS r ON c.Room\_ID = r.Room\_ID

WHERE c.Start\_time LIKE '%Mon%'

GROUP BY c.Tutor\_ID, Tutor\_Name, c.Class\_ID, c.Start\_time, c.End\_time, r.Room\_ID

ORDER BY c.Class\_ID;

**Result:**

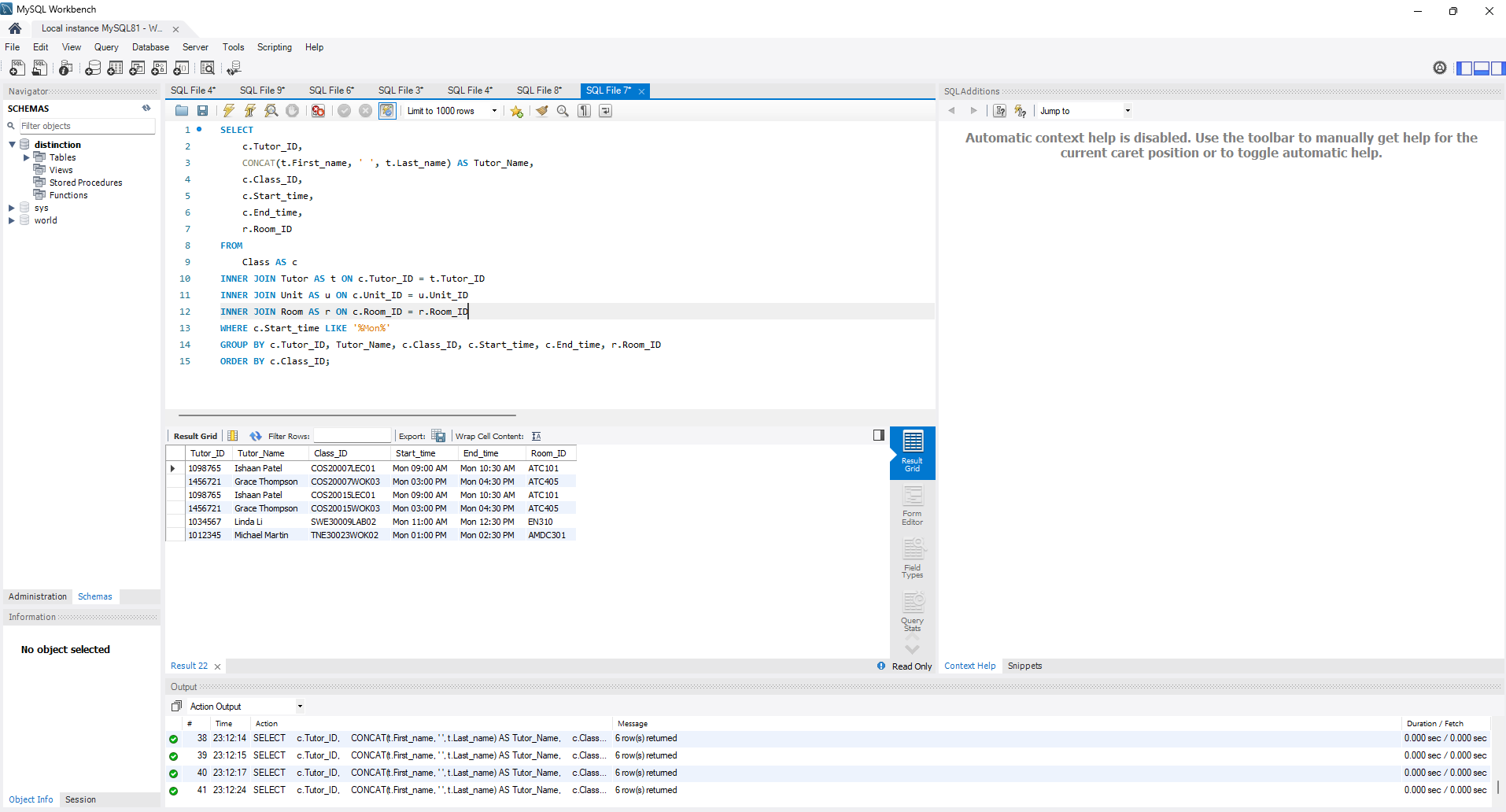
****

Figure 10: Query 2’s result

* This query returns the tutors that have classes on Monday.

1. **Query 3:**

SELECT

    c.Class\_ID,

    c.Start\_time,

    c.End\_time,

    c.Room\_ID,

    COUNT(s.Student\_ID) AS Students

FROM

    Sessions AS s

RIGHT JOIN Class AS c ON s.Class\_ID = c.Class\_ID

GROUP BY c.Class\_ID

ORDER BY Students DESC;

**Result:**

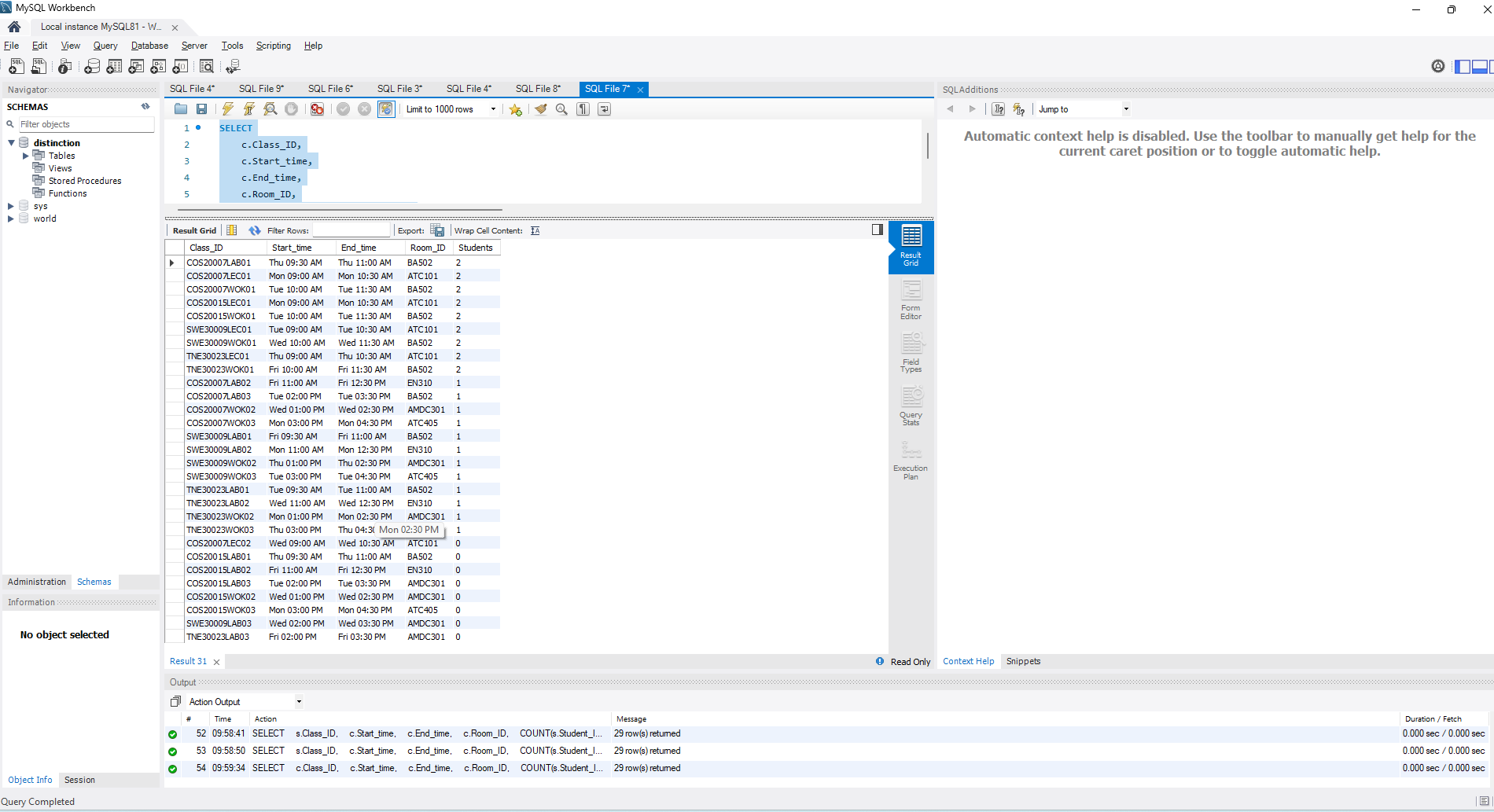
****

Figure 11: Query 3’s result

* This query returns all classes in the order from most attending students to 0 attendance using RIGHT JOIN.

1. **Query 4:**

SELECT

    u.Unit\_name,

    u.Unit\_ID,

    COUNT(s.Student\_ID) AS Enrolled\_Students

FROM

    Unit AS u

LEFT JOIN Class AS c ON u.Unit\_ID = c.Unit\_ID

LEFT JOIN Sessions AS s ON c.Class\_ID = s.Class\_ID

GROUP BY u.Unit\_name, u.Unit\_ID

ORDER BY Enrolled\_Students DESC;

**Result:**

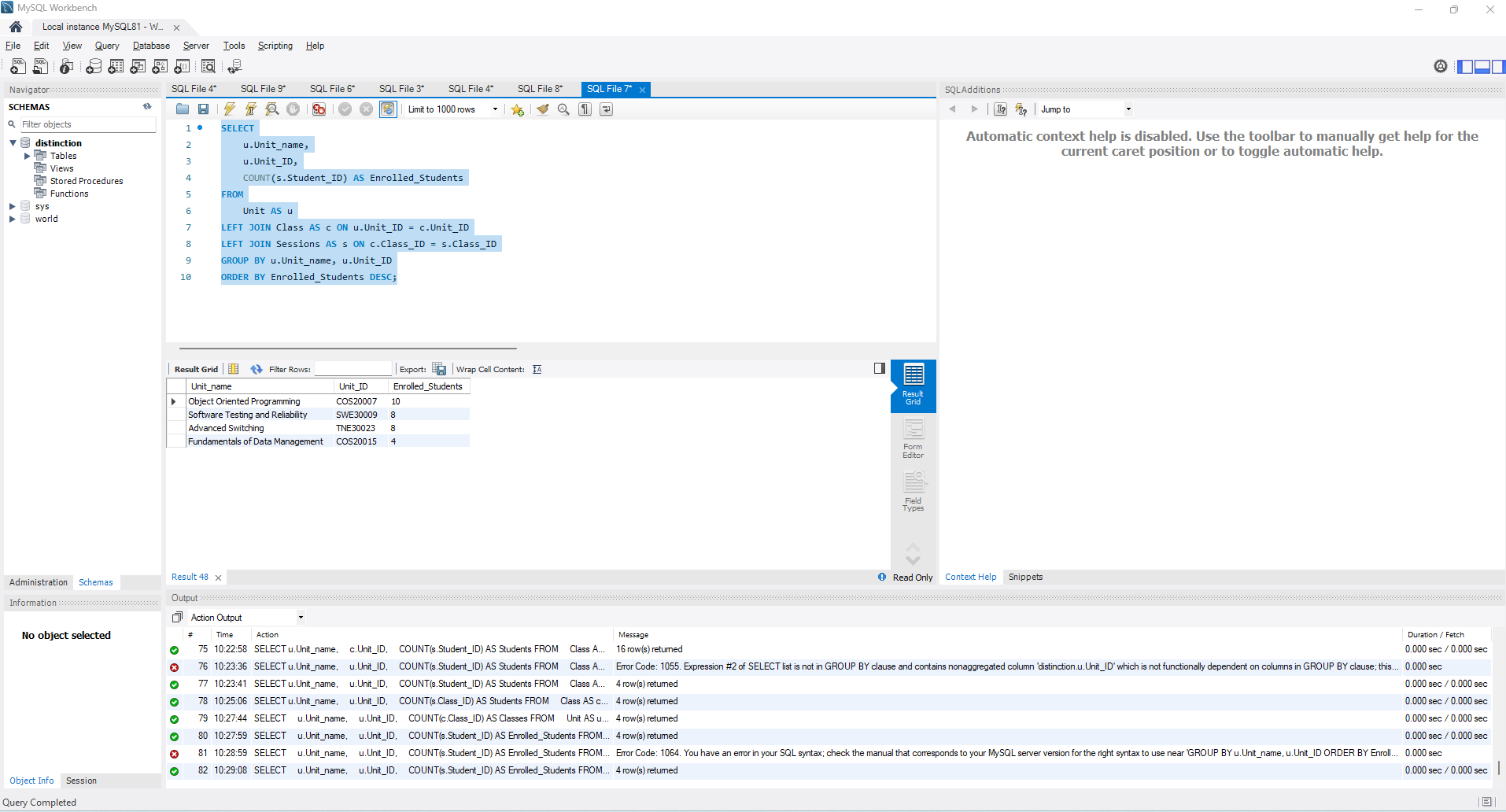
****

Figure 12: Query 4’s result

* This query returns the number of students enrolled in classes of each unit.

1. **Query 5:**

SELECT

    t.Tutor\_ID,

    CONCAT(t.First\_name, ' ', t.Last\_name) AS Tutor\_Name,

    COUNT(s.Student\_ID) AS Students

FROM Tutor as t

LEFT OUTER JOIN Class AS c ON t.Tutor\_ID = c.Tutor\_ID

LEFT OUTER JOIN Sessions AS s ON c.Class\_ID = s.Class\_ID

GROUP BY t.Tutor\_ID

ORDER BY Tutor\_Name;

**Result:**

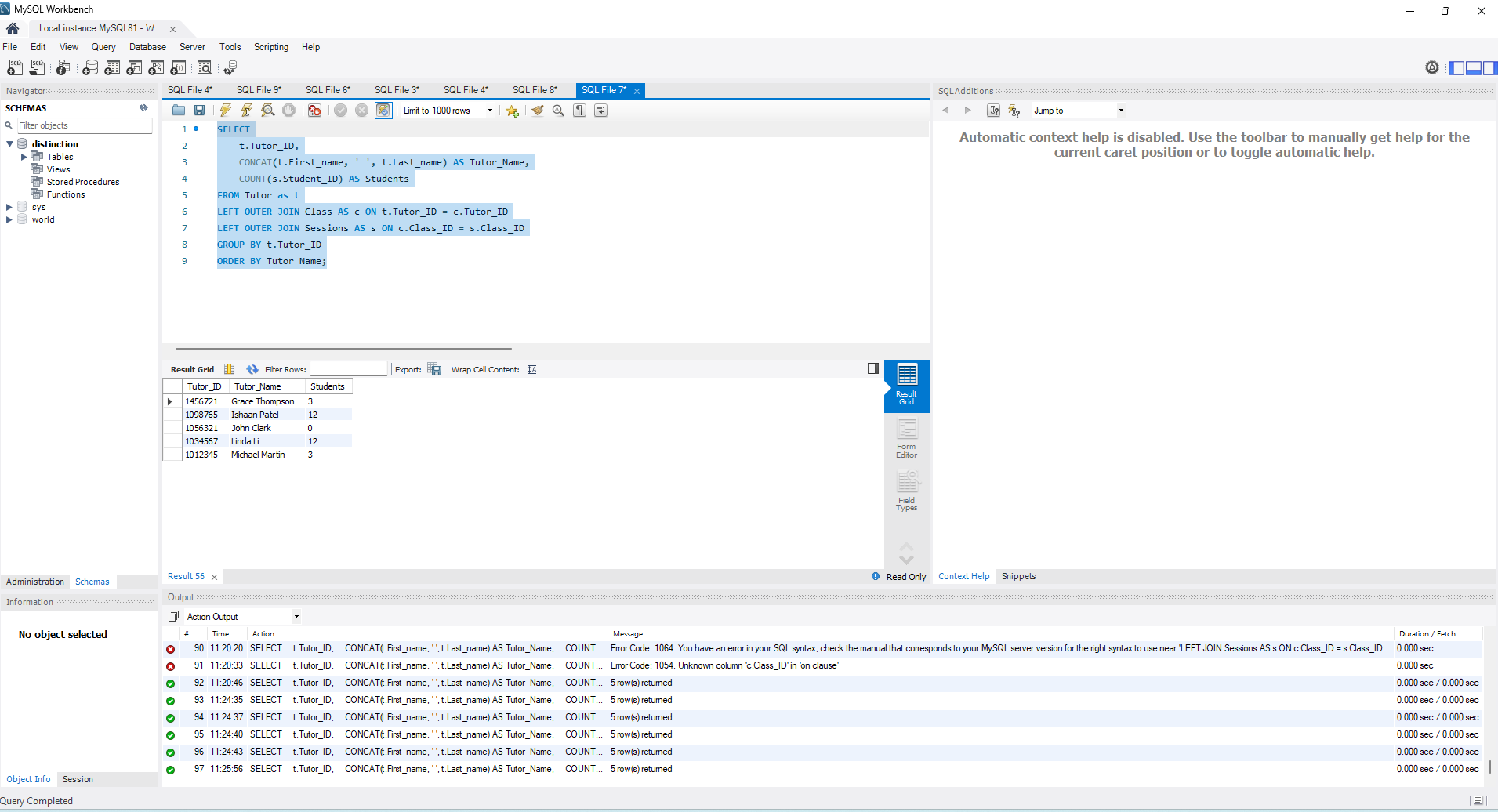
****

Figure 13: Query 5’s result

* This query returns the number of students that a tutor is teaching every week.

1. **Link to Presentation Video (In case the submitted video has issues)**

[**https://youtu.be/bpLih1vGkDw**](https://youtu.be/bpLih1vGkDw)