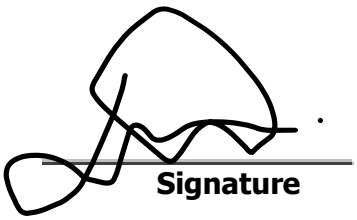
Faculty of Information Technology

I declare that I am familiar with, and will abide to the Examination rules of CTU



Signature

SUBJECT NAME: Business Programming
SUBJECT CODE: PRG522

Formative Assessment 2
Duration: 36 Days
Date: 22-09-2023
Total Marks: 100
Total pages: -

Examiner: Mr Junior Manganyi
Moderator:

Student number

2	0	2	3	1	4	0	5	

Surname: Moshoeshoe

Initials: Leduma Tlotliso

/ %

Question 1:

Code:

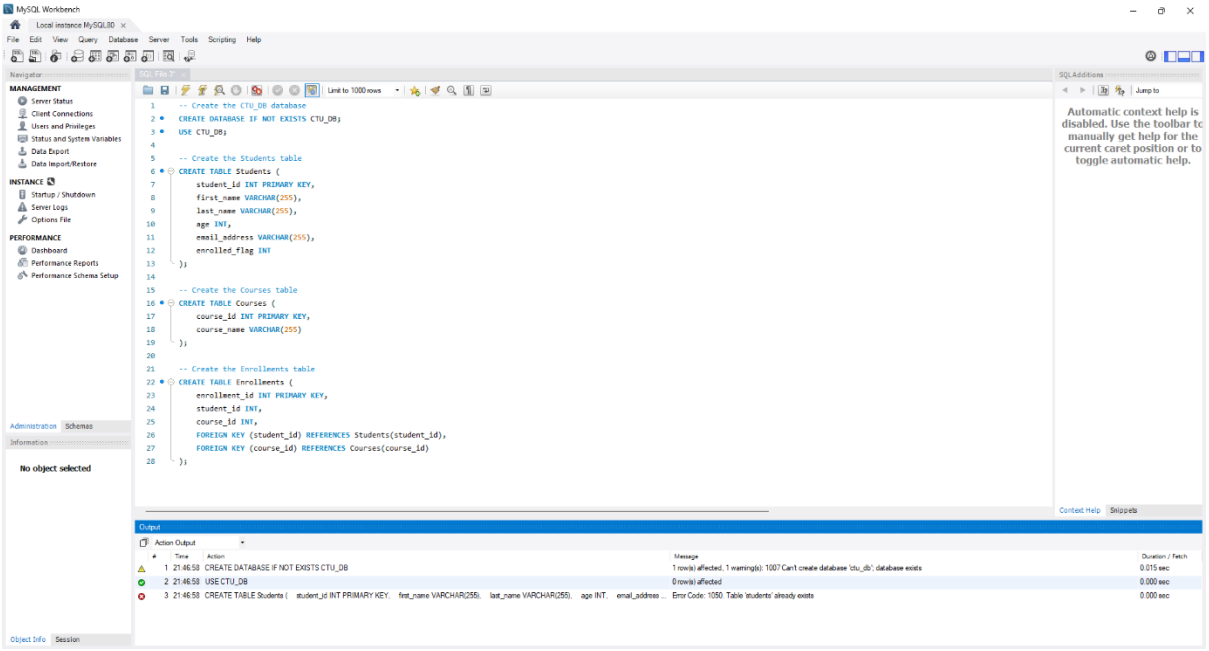
```
-- Create the CTU_DB database
CREATE DATABASE IF NOT EXISTS CTU_DB;
USE CTU_DB;

-- Create the Students table
CREATE TABLE Students (
   student_id INT PRIMARY KEY,
   first_name VARCHAR(255),
   last_name VARCHAR(255),
   age INT,
   email_address VARCHAR(255),
   enrolled_flag INT
);

-- Create the Courses table
CREATE TABLE Courses (
   course_id INT PRIMARY KEY,
   course_name VARCHAR(255)
);

-- Create the Enrollments table
CREATE TABLE Enrollments (
   enrollment_id INT PRIMARY KEY,
   student_id INT,
   course_id INT,
   FOREIGN KEY (student_id) REFERENCES Students(student_id),
   FOREIGN KEY (course_id) REFERENCES Courses(course_id)
);
```

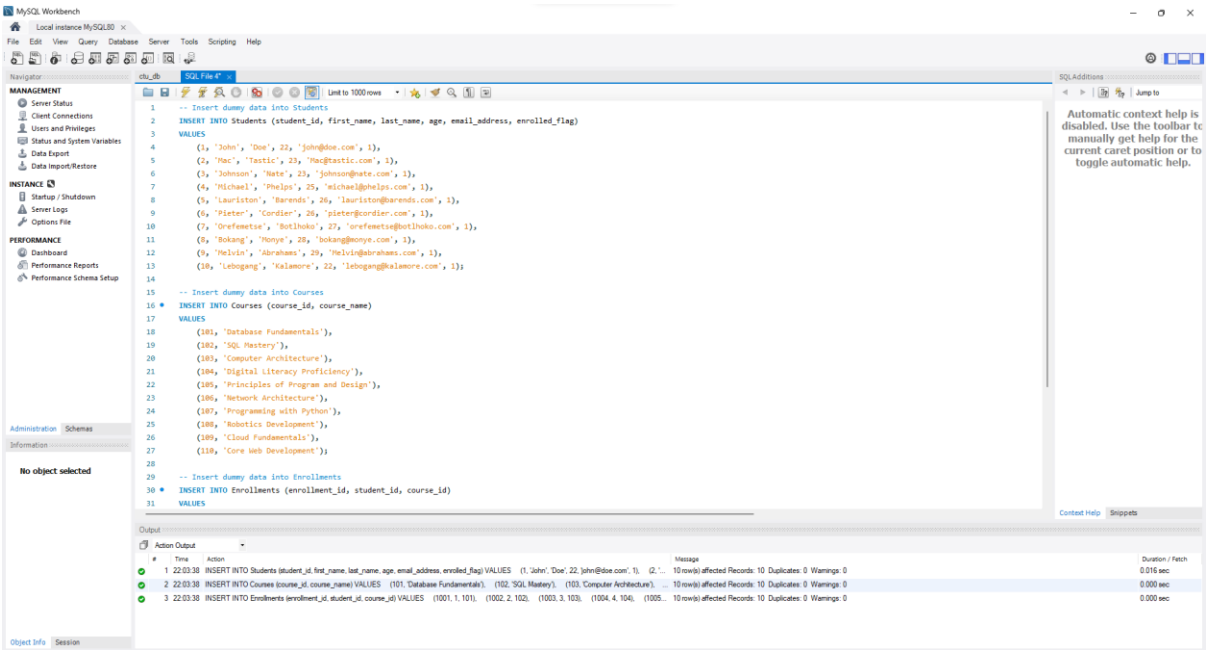
```
-- Insert dummy data into Students
INSERT INTO Students (student_id, first_name, last_name, age, email_address, enrolled_flag)
VALUES
    (1, 'John', 'Doe', 22, 'john@doe.com', 1),
    (2, 'Mac', 'Tastic', 23, 'Mac@tastic.com', 1),
    (3, 'Johnson', 'Nate', 23, 'johnson@nate.com', 1),
    (4, 'Michael', 'Phelps', 25, 'michael@phelps.com', 1),
    (5, 'Lauriston', 'Barends', 26, 'lauriston@barends.com', 1),
    (6, 'Pieter', 'Cordier', 26, 'pieter@cordier.com', 1),
    (7, 'Orefemetse', 'Botlhoko', 27, 'orefemetse@botlhoko.com', 1),
    (8, 'Bokang', 'Monye', 28, 'bokang@monye.com', 1),
    (9, 'Melvin', 'Abrahams', 29, 'Melvin@abrahams.com', 1),
    (10, 'Lebogang', 'Kalamore', 22, 'lebogang@kalamore.com', 1);

-- Insert dummy data into Courses
INSERT INTO Courses (course_id, course_name)
VALUES
    (101, 'Database Fundamentals'),
    (102, 'SQL Mastery'),
    (103, 'Computer Architecture'),
    (104, 'Digital Literacy Proficiency'),
    (105, 'Principles of Program and Design'),
    (106, 'Network Architecture'),
    (107, 'Programming with Python'),
    (108, 'Robotics Development'),
    (109, 'Cloud Fundamentals'),
    (110, 'Core Web Development');

-- Insert dummy data into Enrollments
INSERT INTO Enrollments (enrollment_id, student_id, course_id)
```

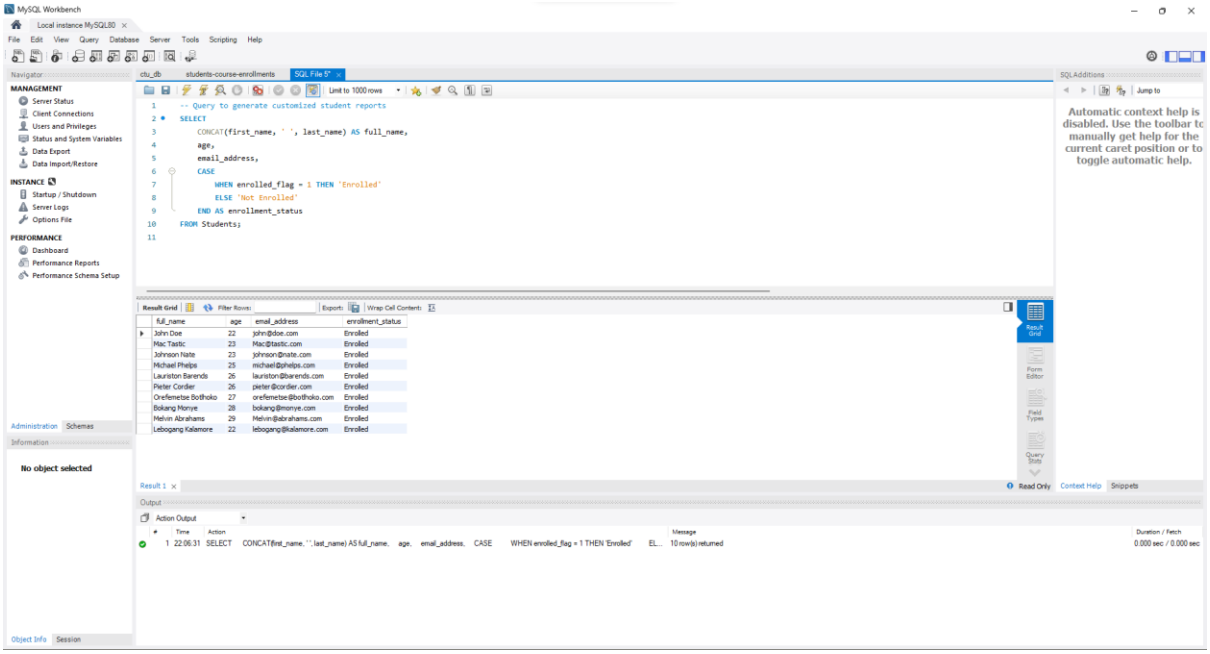
```
VALUES
    (1001, 1, 101),
    (1002, 2, 102),
    (1003, 3, 103),
    (1004, 4, 104),
    (1005, 5, 105),
    (1006, 6, 106),
    (1007, 7, 107),
    (1008, 8, 108),
    (1009, 9, 109),
    (1010, 10, 110);

-- Query to generate customized student reports
SELECT
    CONCAT(first_name, ' ', last_name) AS full_name,
    age,
    email_address,
    CASE
        WHEN enrolled_flag = 1 THEN 'Enrolled'
        ELSE 'Not Enrolled'
    END AS enrollment_status
FROM Students;

ALTER TABLE Students ADD COLUMN assessment_score INT;

-- Query to convert email addresses to lowercase and classify students
SELECT
    CONCAT(first_name, ' ', last_name) AS full_name,
    age,
    LOWER(email_address) AS lowercase_email,
    CASE
```

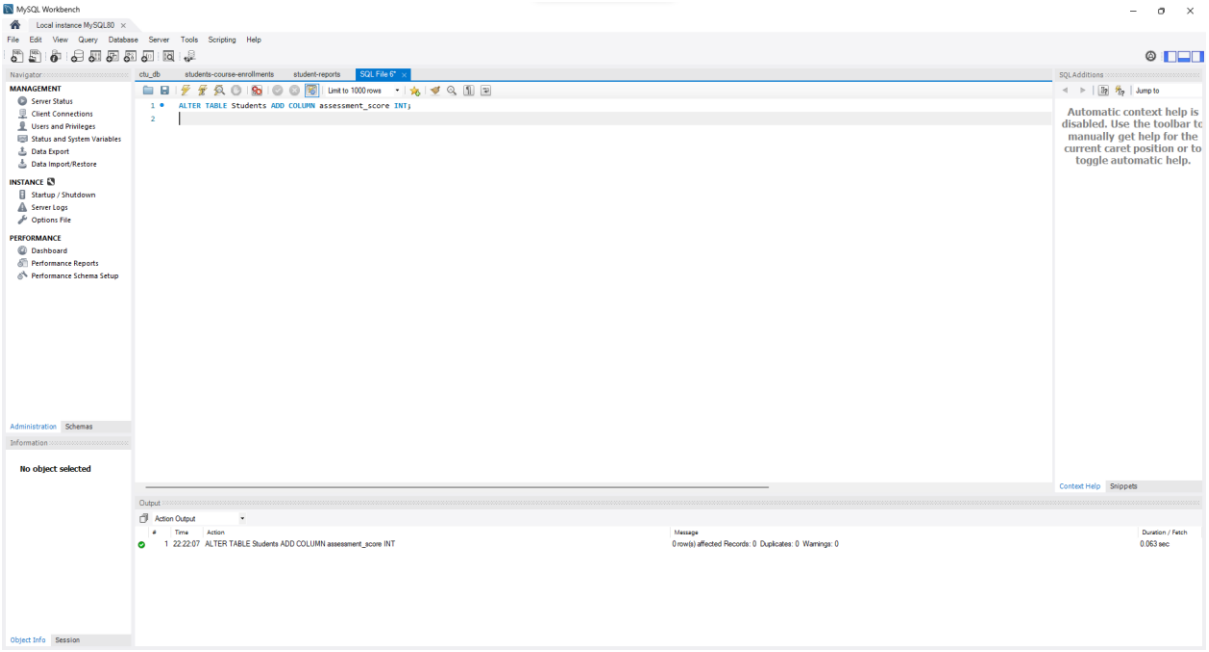
```
        WHEN assessment_score >= 90 THEN 'Advanced'
        WHEN assessment_score >= 70 THEN 'Intermediate'
        ELSE 'Beginner'
    END AS student_category
FROM Students;
-- Note: You need to add the assessment_score column to the Students table to run this query.

-- Query to calculate and display the average score for each course
SELECT
    C.course_name,
    COUNT(E.student_id) AS num_students_enrolled,
    AVG(S.assessment_score) AS average_score
FROM Courses C
LEFT JOIN Enrollments E ON C.course_id = E.course_id
LEFT JOIN Students S ON E.student_id = S.student_id
GROUP BY C.course_name;
-- Note: You need to add the assessment_score column to the Students table to run this query.

-- Query to retrieve student information and their enrolled courses
SELECT
    CONCAT(S.first_name, ' ', S.last_name) AS student_name,
    S.email_address,
    C.course_name
FROM Students S
INNER JOIN Enrollments E ON S.student_id = E.student_id
INNER JOIN Courses C ON E.course_id = C.course_id
WHERE S.enrolled_flag = 1
ORDER BY student_name;
```

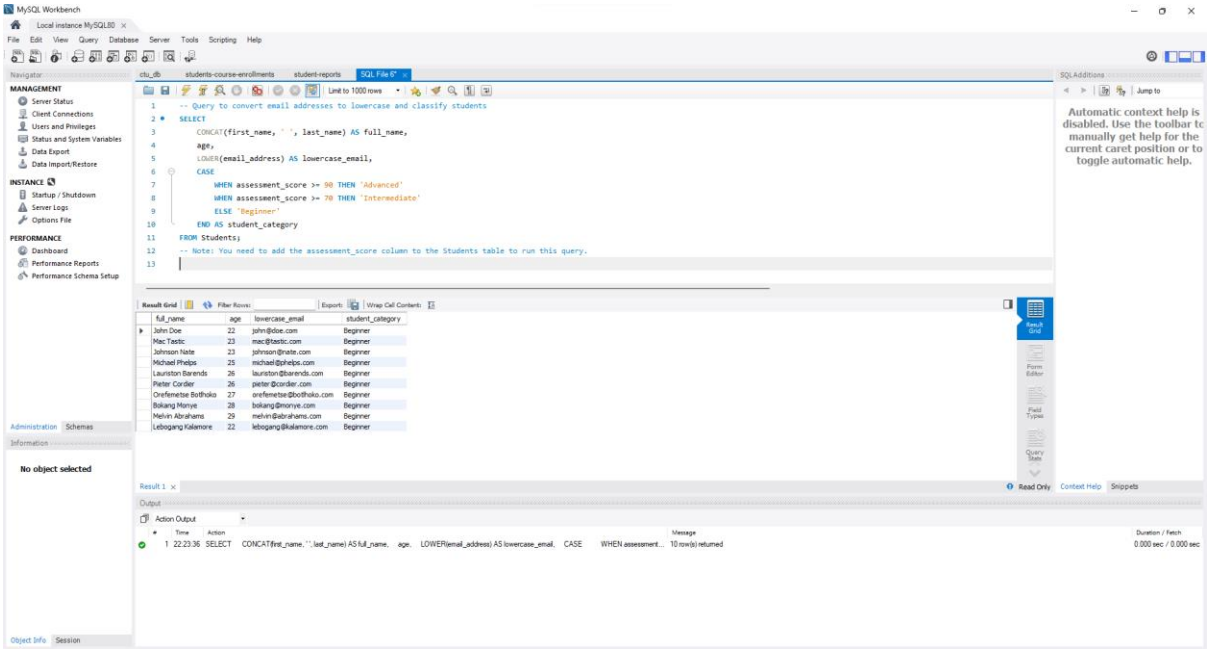
Output:

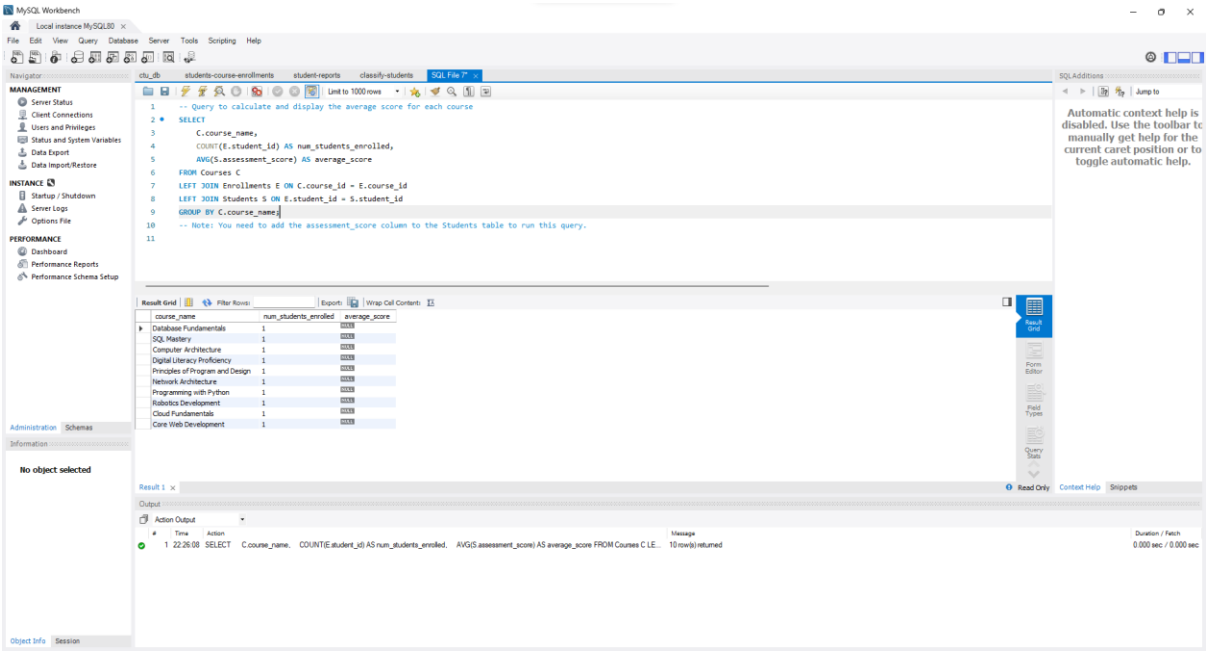

MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Information
No object selected
Object Info Session
SQL File 7*
Limit to 1000 rows
-- Create the CTU_DB database
CREATE DATABASE IF NOT EXISTS CTU_DB;
USE CTU_DB;
-- Create the Students table
CREATE TABLE Students (
student_id INT PRIMARY KEY,
first_name VARCHAR(255),
last_name VARCHAR(255),
age INT,
email_address VARCHAR(255),
enrolled_flag INT
);
-- Create the Courses table
CREATE TABLE Courses (
course_id INT PRIMARY KEY,
course_name VARCHAR(255)
);
-- Create the Enrollments table
CREATE TABLE Enrollments (
enrollment_id INT PRIMARY KEY,
student_id INT,
course_id INT,
FOREIGN KEY (student_id) REFERENCES Students(student_id),
FOREIGN KEY (course_id) REFERENCES Courses(course_id)
);
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 21:46:58 CREATE DATABASE IF NOT EXISTS CTU_DB 1 row(s) affected, 1 warning(s): 1007 Can't create database 'ctu_db'; database exists 0.015 sec
2 21:46:58 USE CTU_DB 0 row(s) affected 0.000 sec
3 21:46:58 CREATE TABLE Students (student_id INT PRIMARY KEY, first_name VARCHAR(255), last_name VARCHAR(255), age INT, email_address ... Error Code: 1050. Table 'students' already exists 0.000 sec

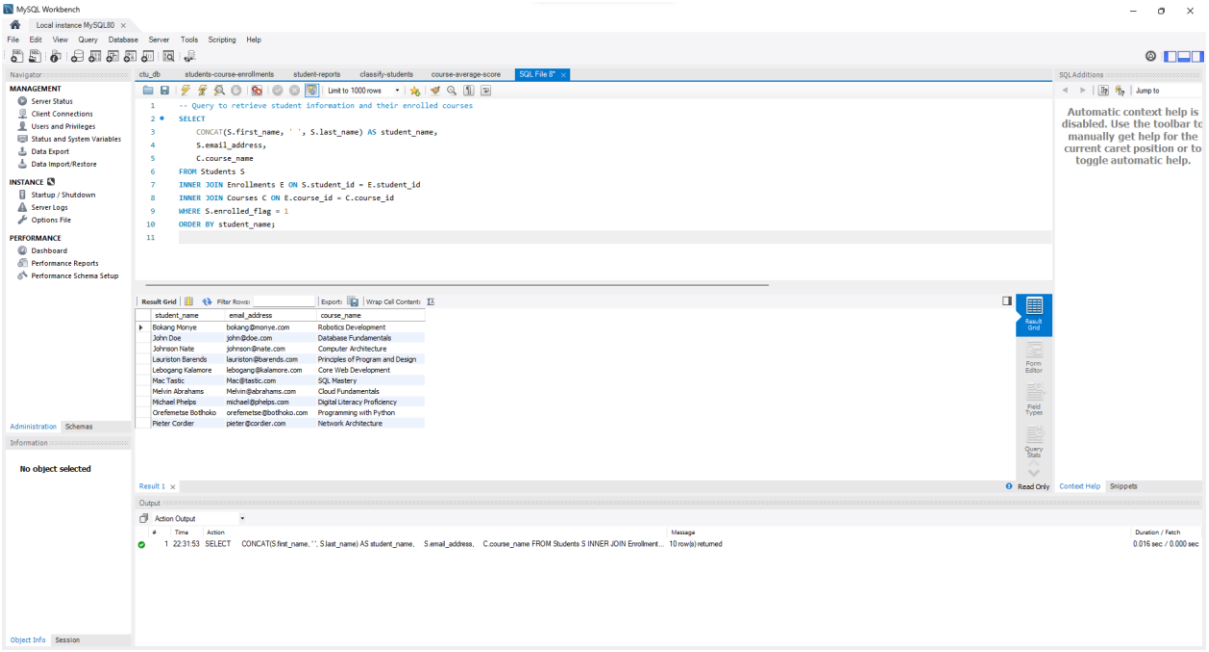

MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Information
No object selected
Object Info Session
ctu_db
SQL File 4*
Limit to 1000 rows
-- Insert dummy data into Students
INSERT INTO Students (student_id, first_name, last_name, age, email_address, enrolled_flag)
VALUES
(1, 'John', 'Doe', 22, 'john@doe.com', 1),
(2, 'Mac', 'Tastic', 23, 'Mac@tastic.com', 1),
(3, 'Johnson', 'Nate', 23, 'johnson@nate.com', 1),
(4, 'Michael', 'Phelps', 25, 'michael@phelps.com', 1),
(5, 'Lauriston', 'Barends', 26, 'lauriston@barends.com', 1),
(6, 'Pieter', 'Cordier', 26, 'pieter@cordier.com', 1),
(7, 'Orefemetse', 'Botlhoko', 27, 'orefemetse@botlhoko.com', 1),
(8, 'Bokang', 'Monye', 28, 'bokang@monye.com', 1),
(9, 'Melvin', 'Abrahams', 29, 'Melvin@abrahams.com', 1),
(10, 'Lebogang', 'Kalamore', 22, 'lebogang@kalamore.com', 1);
-- Insert dummy data into Courses
INSERT INTO Courses (course_id, course_name)
VALUES
(101, 'Database Fundamentals'),
(102, 'SQL Mastery'),
(103, 'Computer Architecture'),
(104, 'Digital Literacy Proficiency'),
(105, 'Principles of Program and Design'),
(106, 'Network Architecture'),
(107, 'Programming with Python'),
(108, 'Robotics Development'),
(109, 'Cloud Fundamentals'),
(110, 'Core Web Development');
-- Insert dummy data into Enrollments
INSERT INTO Enrollments (enrollment_id, student_id, course_id)
VALUES
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 22:03:38 INSERT INTO Students (student_id, first_name, last_name, age, email_address, enrolled_flag) VALUES (1, 'John', 'Doe', 22, 'john@doe.com', 1), (2, '... 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0 0.016 sec
2 22:03:38 INSERT INTO Courses (course_id, course_name) VALUES (101, 'Database Fundamentals'), (102, 'SQL Mastery'), (103, 'Computer Architecture'), ... 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0 0.000 sec
3 22:03:38 INSERT INTO Enrollments (enrollment_id, student_id, course_id) VALUES (1001, 1, 101), (1002, 2, 102), (1003, 3, 103), (1004, 4, 104), (1005... 10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0 0.000 sec


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Information
No object selected
Object Info Session
ctu_db
students-course-enrollments
SQL File 5*
Limit to 1000 rows
-- Query to generate customized student reports
SELECT
CONCAT(first_name, ' ', last_name) AS full_name,
age,
email_address,
CASE
WHEN enrolled_flag = 1 THEN 'Enrolled'
ELSE 'Not Enrolled'
END AS enrollment_status
FROM Students;
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
full_name age email_address enrollment_status
John Doe 22 john@doe.com Enrolled
Mac Tastic 23 Mac@tastic.com Enrolled
Johnson Nate 23 johnson@nate.com Enrolled
Michael Phelps 25 michael@phelps.com Enrolled
Lauriston Barends 26 lauriston@barends.com Enrolled
Pieter Cordier 26 pieter@cordier.com Enrolled
Orefemetse Bodhoko 27 orefemetse@bodhoko.com Enrolled
Bokang Monye 28 bokang@monye.com Enrolled
Melvin Abrahams 29 Melvin@abrahams.com Enrolled
Lebogang Kalamore 22 lebogang@kalamore.com Enrolled
Result 1
Read Only
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 22:06:31 SELECT CONCAT(first_name, ' ', last_name) AS full_name, age, email_address, CASE WHEN enrolled_flag = 1 THEN 'Enrolled' EL... 10 row(s) returned 0.000 sec / 0.000 sec


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Information
No object selected
Object Info Session
ctu_db
students-course-enrollments
student-reports
SQL File 6*
Limit to 1000 rows
ALTER TABLE Students ADD COLUMN assessment_score INT;
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 22:22:07 ALTER TABLE Students ADD COLUMN assessment_score INT 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0 0.063 sec


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
ctu_db
students-course-enrollments
student-reports
SQL File 6*
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Information
No object selected
Object Info Session
Limit to 1000 rows
1 -- Query to convert email addresses to lowercase and classify students
2 SELECT
3 CONCAT(first_name, ' ', last_name) AS full_name,
4 age,
5 LOWER(email_address) AS lowercase_email,
6 CASE
7 WHEN assessment_score >= 90 THEN 'Advanced'
8 WHEN assessment_score >= 70 THEN 'Intermediate'
9 ELSE 'Beginner'
10 END AS student_category
11 FROM Students;
12 -- Note: You need to add the assessment_score column to the Students table to run this query.
13
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
full_name age lowercase_email student_category
John Doe 22 john@doe.com Beginner
Mac Tastic 23 mac@tastic.com Beginner
Johnson Nate 23 johnson@nate.com Beginner
Michael Phelps 25 michael@phelps.com Beginner
Lauriston Barends 26 lauriston@barends.com Beginner
Pieter Cordier 26 pieter@cordier.com Beginner
Orefemetse Bothoko 27 orefemetse@bothoko.com Beginner
Bokang Monye 28 bokang@monye.com Beginner
Melvin Abrahams 29 melvin@abrahams.com Beginner
Lebogang Kalamore 22 lebogang@kalamore.com Beginner
Result 1
Read Only
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 22:23:36 SELECT CONCAT(first_name, ' ', last_name) AS full_name, age, LOWER(email_address) AS lowercase_email, CASE WHEN assessment... 10 row(s) returned 0.000 sec / 0.000 sec
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
ctu_db
students-course-enrollments
student-reports
classify-students
SQL File 7*
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Information
No object selected
Object Info Session
Limit to 1000 rows
1 -- Query to calculate and display the average score for each course
2 SELECT
3 C.course_name,
4 COUNT(E.student_id) AS num_students_enrolled,
5 AVG(S.assessment_score) AS average_score
6 FROM Courses C
7 LEFT JOIN Enrollments E ON C.course_id = E.course_id
8 LEFT JOIN Students S ON E.student_id = S.student_id
9 GROUP BY C.course_name;
10 -- Note: You need to add the assessment_score column to the Students table to run this query.
11
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
course_name num_students_enrolled average_score
Database Fundamentals 1 NULL
SQL Mastery 1 NULL
Computer Architecture 1 NULL
Digital Literacy Proficiency 1 NULL
Principles of Program and Design 1 NULL
Network Architecture 1 NULL
Programming with Python 1 NULL
Robotics Development 1 NULL
Cloud Fundamentals 1 NULL
Core Web Development 1 NULL
Result 1
Read Only
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 22:26:08 SELECT C.course_name, COUNT(E.student_id) AS num_students_enrolled, AVG(S.assessment_score) AS average_score FROM Courses C LE... 10 row(s) returned 0.000 sec / 0.000 sec
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.


MySQL Workbench
Local instance MySQL80
File Edit View Query Database Server Tools Scripting Help
Navigator
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Information
No object selected
Object Info Session
ctu_db
students-course-enrollments
student-reports
classify-students
course-average-score
SQL File 8*
Limit to 1000 rows
-- Query to retrieve student information and their enrolled courses
SELECT
CONCAT(S.first_name, ' ', S.last_name) AS student_name,
S.email_address,
C.course_name
FROM Students S
INNER JOIN Enrollments E ON S.student_id = E.student_id
INNER JOIN Courses C ON E.course_id = C.course_id
WHERE S.enrolled_flag = 1
ORDER BY student_name;
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
student_name
email_address
course_name
Bokang Monye
bokang@monye.com
Robotics Development
John Doe
john@doe.com
Database Fundamentals
Johnson Nate
johnson@nate.com
Computer Architecture
Lauriston Barends
lauriston@barends.com
Principles of Program and Design
Lebogang Kalamore
lebogang@kalamore.com
Core Web Development
Mac Tastic
Mac@tastic.com
SQL Mastery
Melvin Abrahams
Melvin@abrahams.com
Cloud Fundamentals
Michael Phelps
michael@phelps.com
Digital Literacy Proficiency
Orefemetse Bothoko
orefemetse@bothoko.com
Programming with Python
Pieter Cordier
pieter@cordier.com
Network Architecture
Result Grid
Form Editor
Field Types
Query Stats
Result 1
Read Only
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.
Context Help Snippets
Output
Action Output
Time Action Message Duration / Fetch
1 22:31:53 SELECT CONCAT(S.first_name, ' ', S.last_name) AS student_name, S.email_address, C.course_name FROM Students S INNER JOIN Enrollment... 10 row(s) returned 0.016 sec / 0.000 sec

Completed Declaration of Authenticity

I Leduma Tlotliso Moshoeshoe hereby

(FULL NAME)

declare that the contents of this assignment PRG522 are entirely my work except for the following documents: (List the documents and page numbers of work in this portfolio that were generated in a group)

Activity	Date

Signature:  Date: 22-09-2023