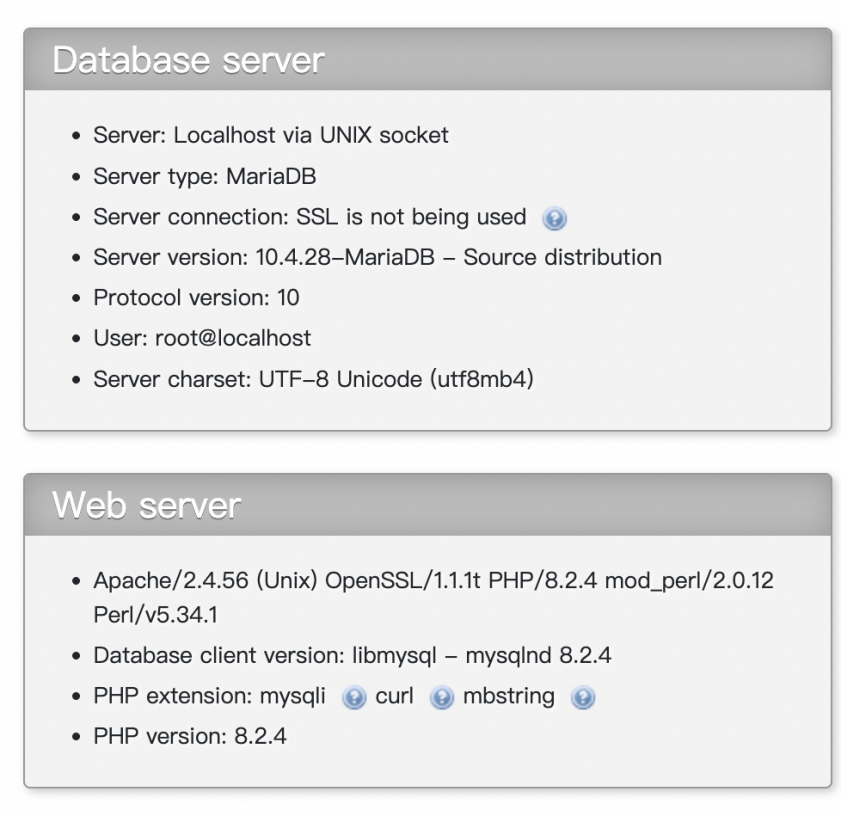
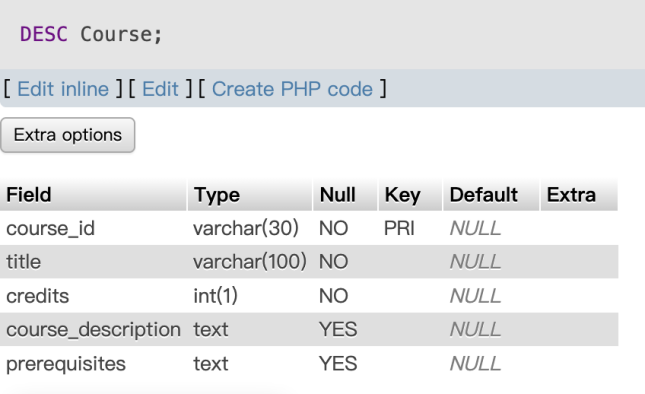
REPORT

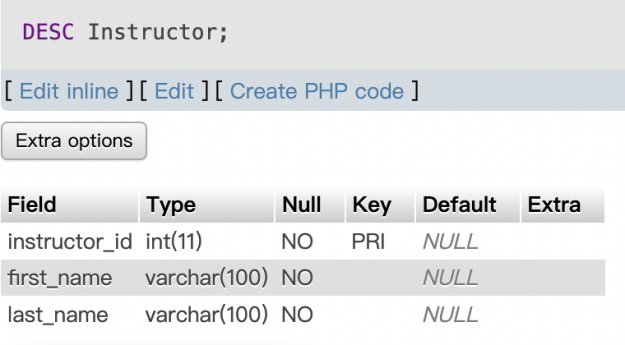
1. A) database server:   
    

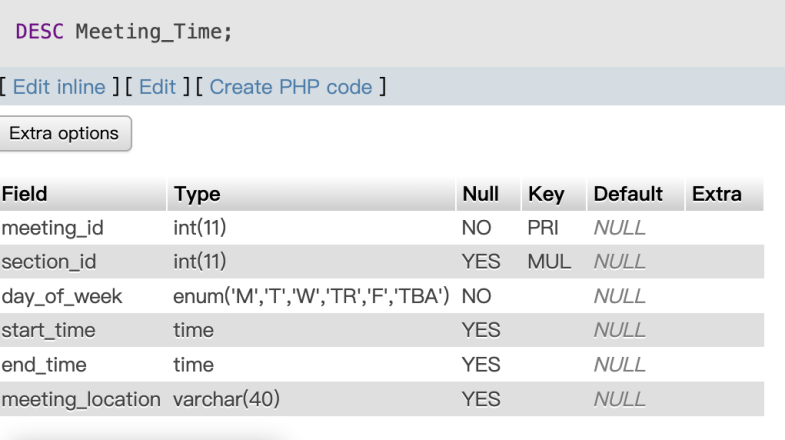
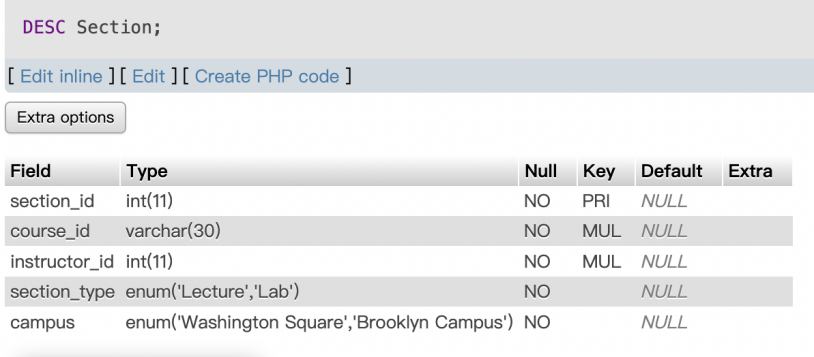
All query runs on the web version of XAMPP

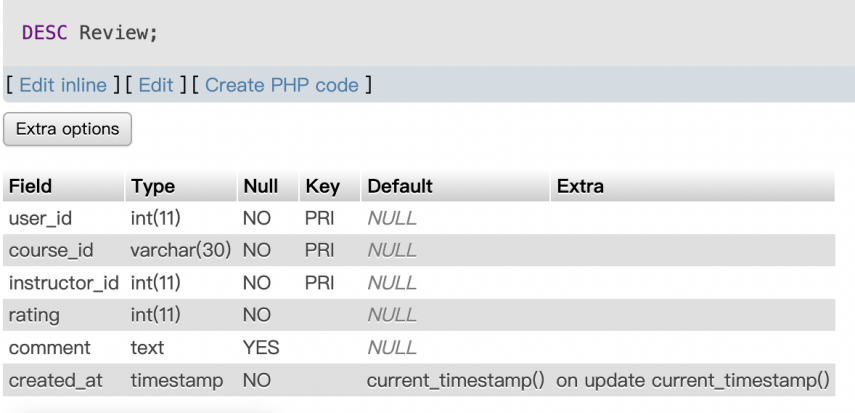
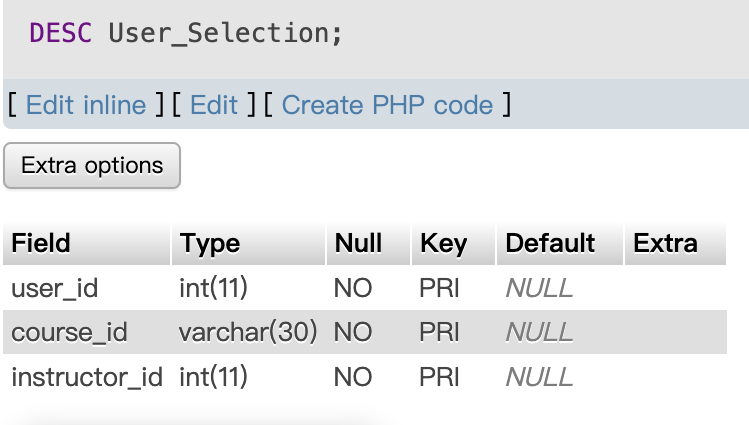
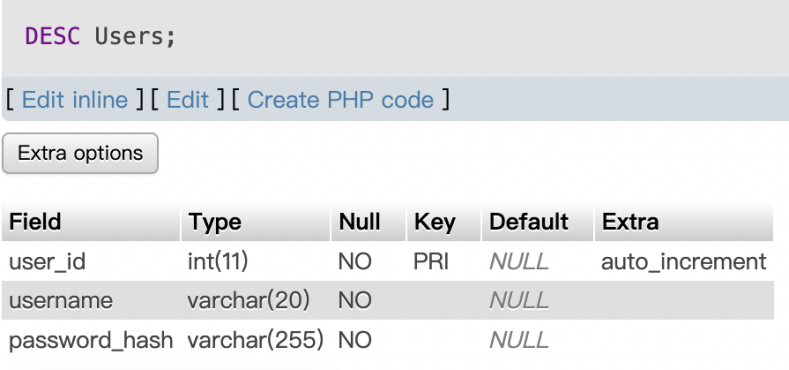
1. A) Table schema

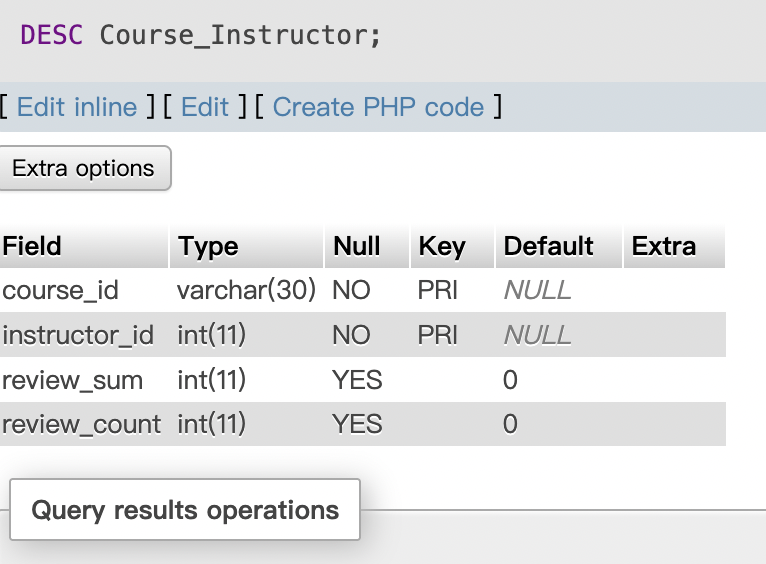
\* Our schema design has been modified due to changes in overall structure and design. The schemas shown in this report are the updated tables in our current design.



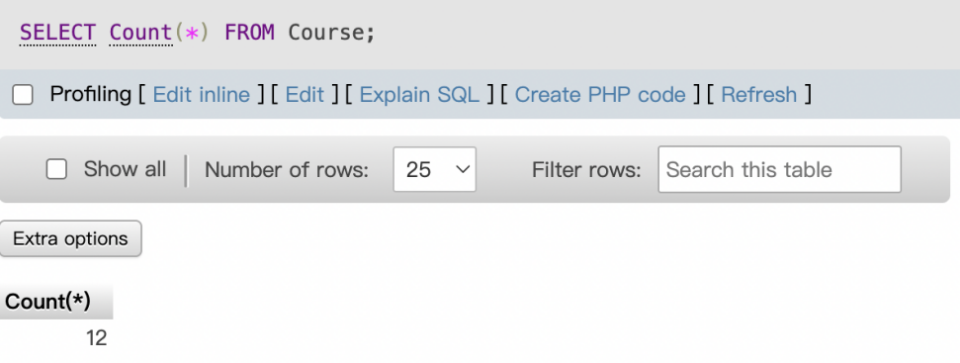


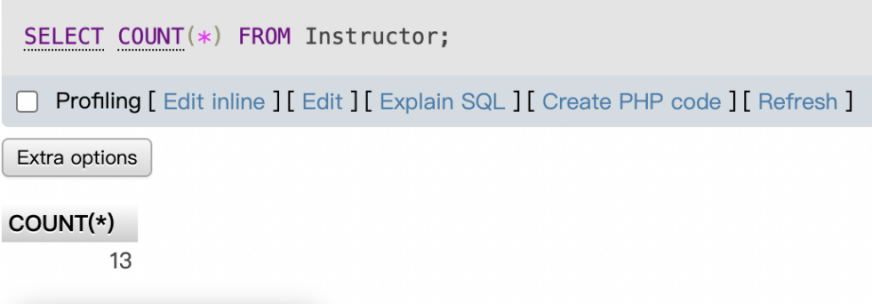


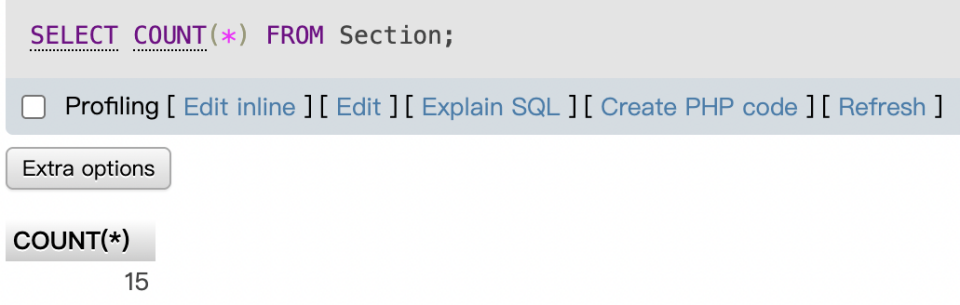


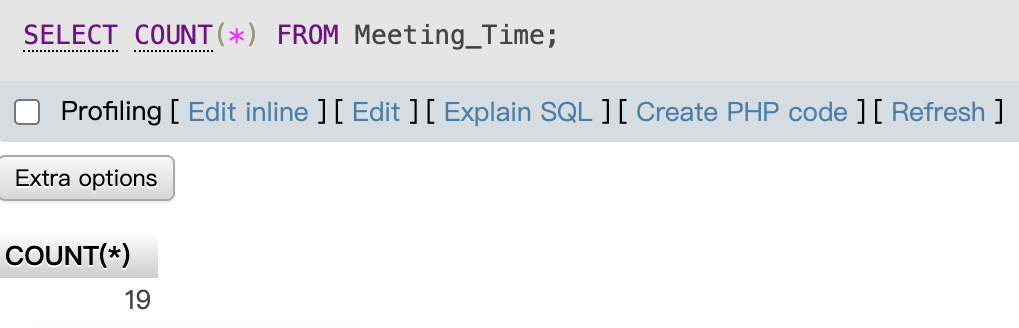


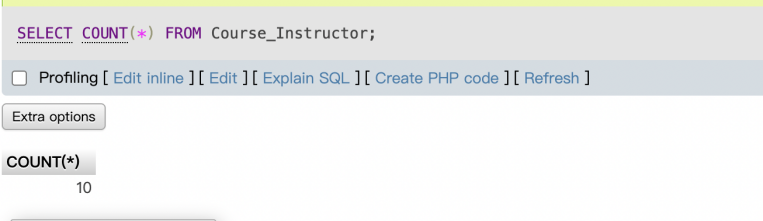
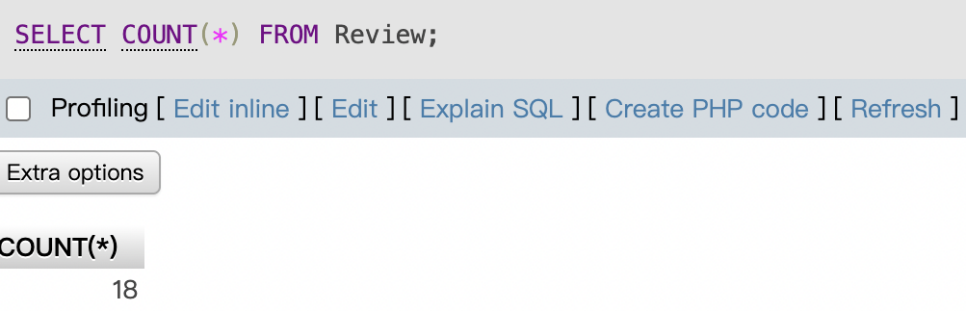
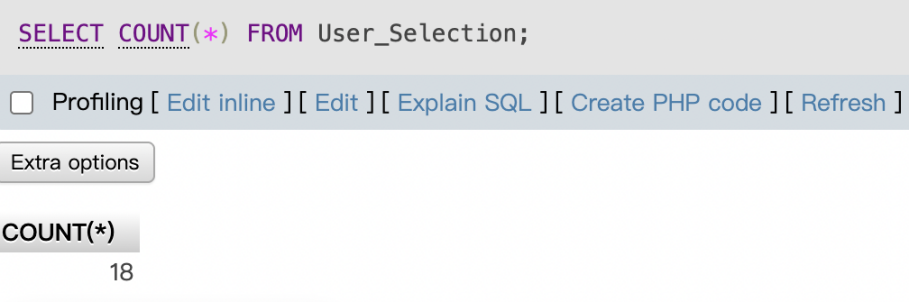
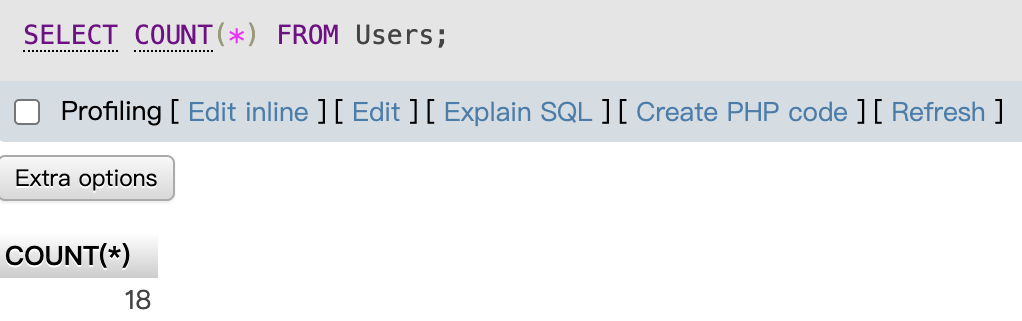
B)

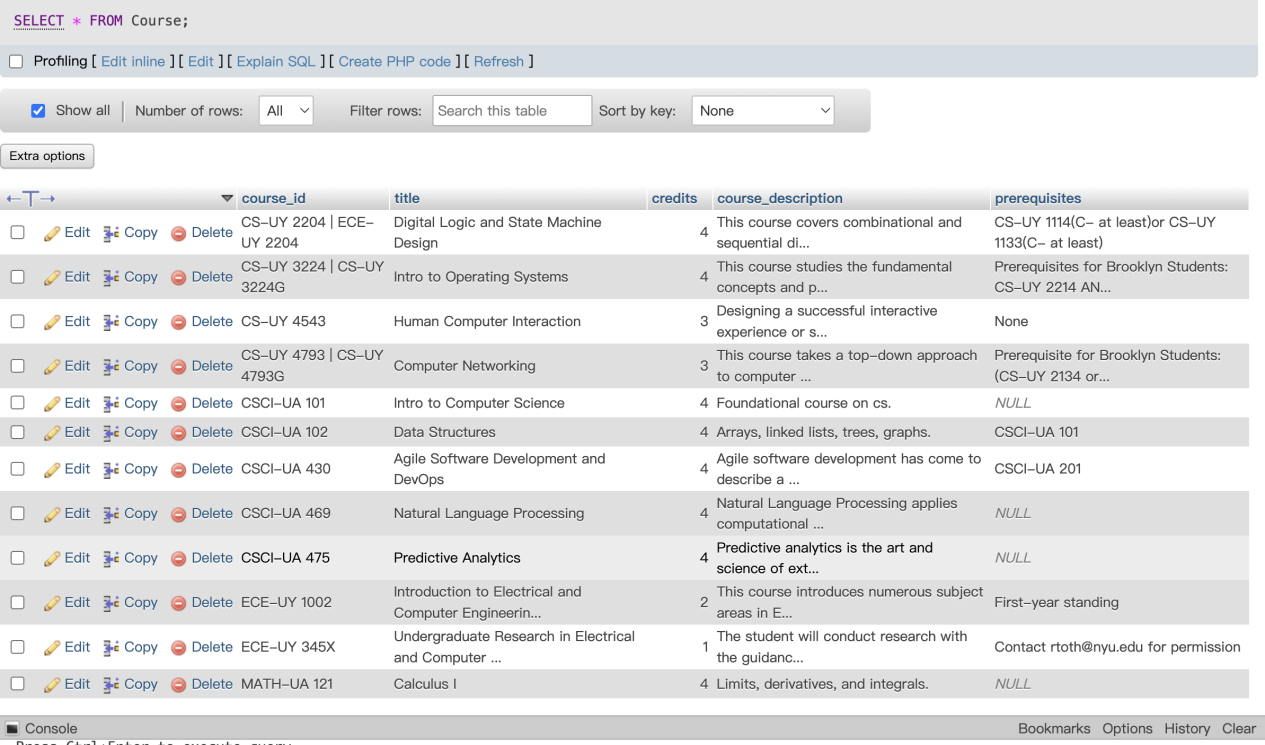


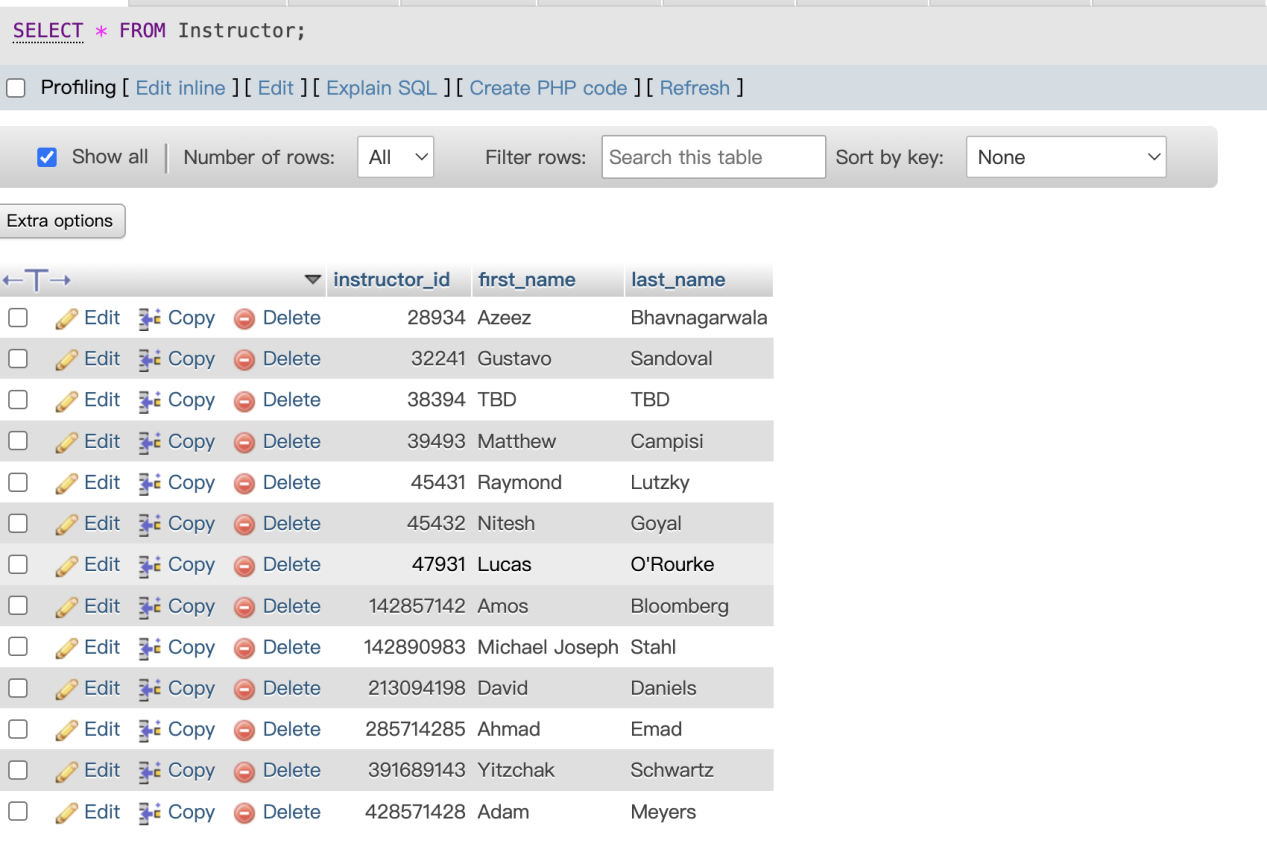


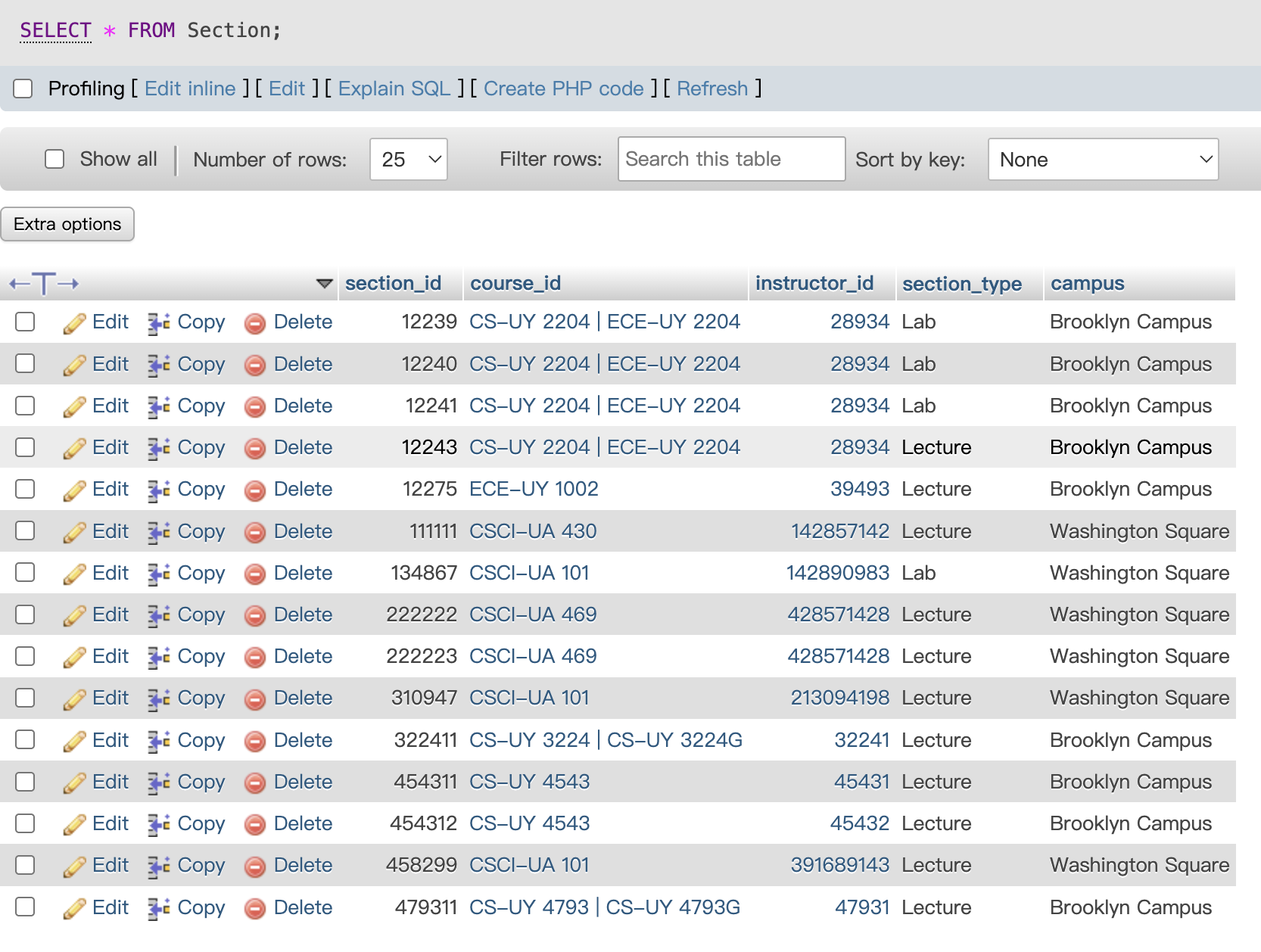


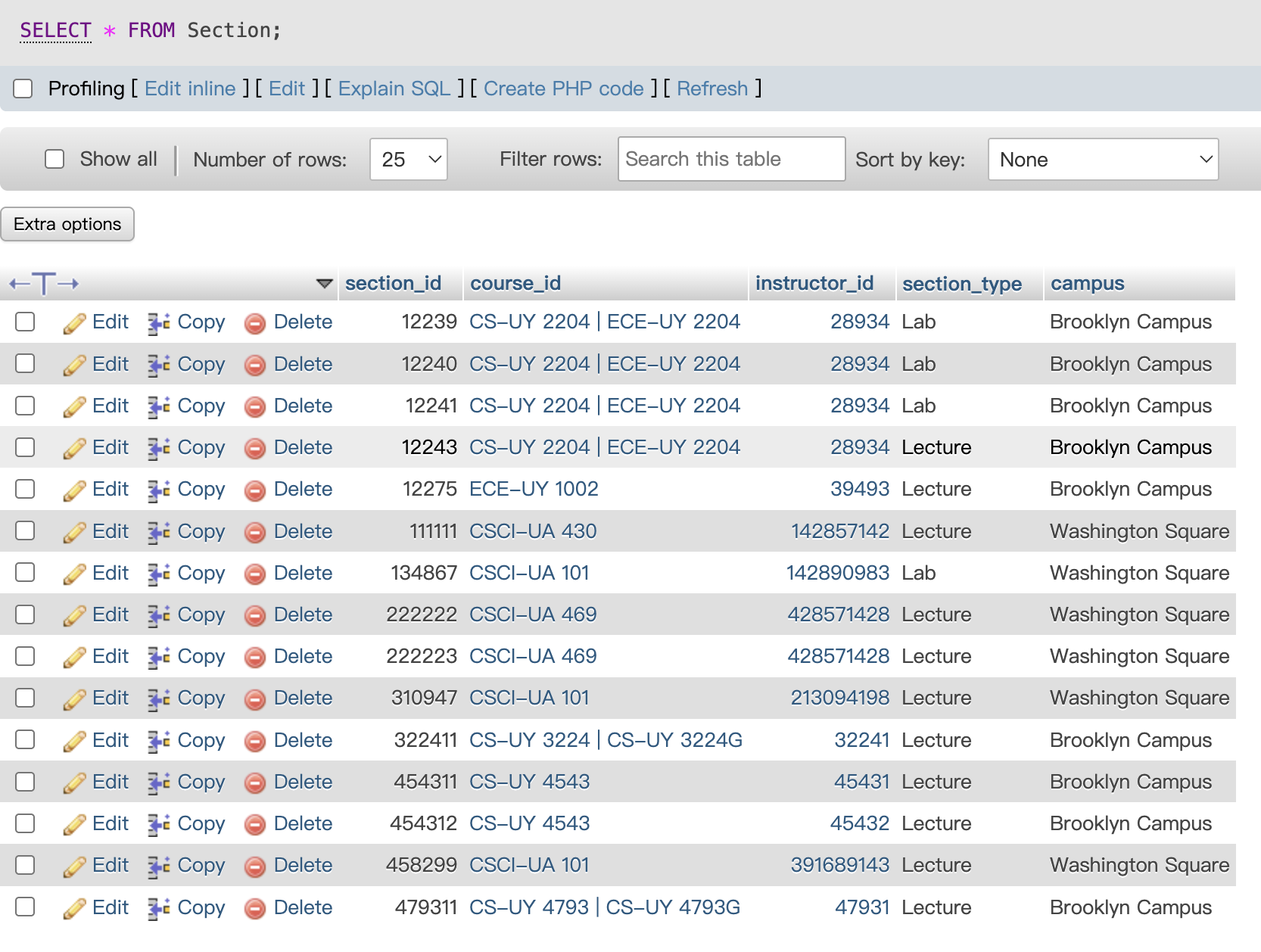




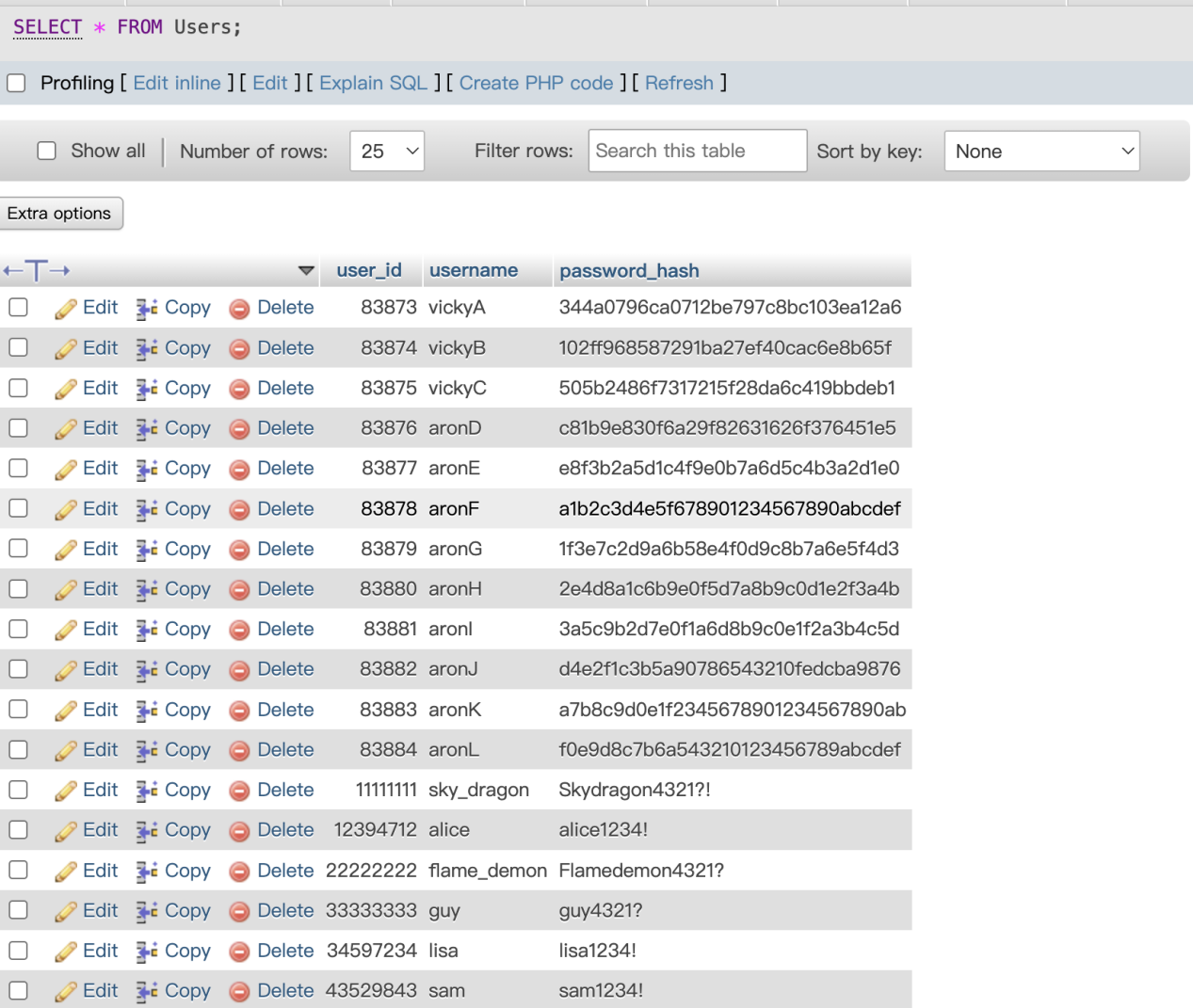




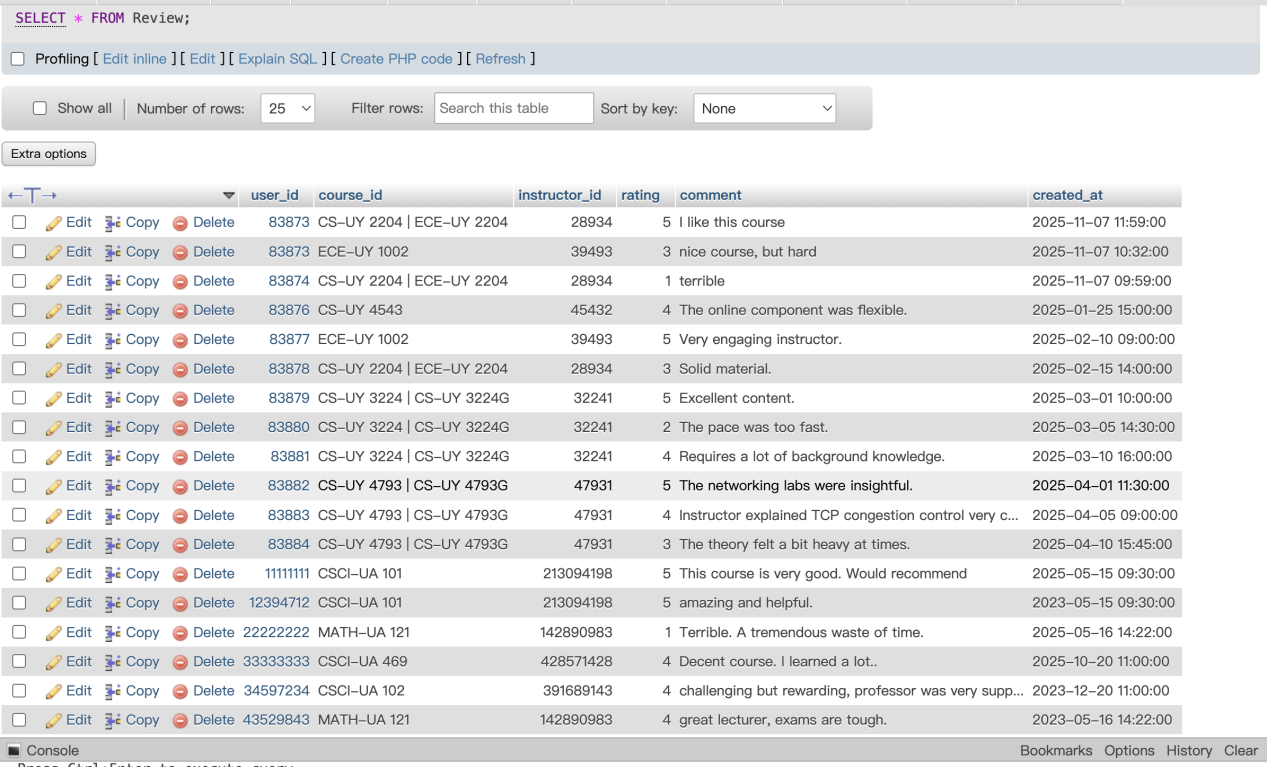




截屏2025-11-09 14.54.35









1. Commands

A)

--CREATE table

create Table Course(

course\_id VARCHAR(30),

title VARCHAR(100) NOT NULL,

credits INT(1) NOT NULL,

course\_description TEXT,

prerequisites TEXT,

Primary Key (course\_id)

);

CREATE TABLE Instructor (

instructor\_id INT PRIMARY KEY,

first\_name VARCHAR(100) NOT NULL,

last\_name VARCHAR(100) NOT NULL

);

create Table Section(

section\_id INT,

course\_id VARCHAR(30) NOT NULL,

instructor\_id INT NOT NULL,

section\_type ENUM('Lecture', 'Lab') NOT NULL,

campus ENUM('Washington Square', 'Brooklyn Campus') NOT NULL,

Primary Key (section\_id),

FOREIGN KEY (course\_id) REFERENCES Course(course\_id),

FOREIGN KEY (instructor\_id) REFERENCES Instructor(instructor\_id)

);

CREATE TABLE Users (

user\_id INT AUTO\_INCREMENT NOT NULL,

username VARCHAR(20) NOT NULL,

password\_hash VARCHAR(255) NOT NULL,

PRIMARY KEY(user\_id)

);

CREATE TABLE Review (

user\_id INT,

course\_id VARCHAR(30),

instructor\_id INT,

rating INT NOT NULL CHECK (rating >= 0 AND rating <= 5),

comment TEXT,

created\_at TIMESTAMP NOT NULL,

PRIMARY KEY (user\_id, course\_id, instructor\_id),

FOREIGN KEY(user\_id) REFERENCES Users(user\_id)

);

CREATE TABLE User\_Selection (

user\_id INT,

course\_id VARCHAR(30),

instructor\_id INT,

FOREIGN KEY(user\_id) REFERENCES Users(user\_id),

FOREIGN KEY(course\_id) REFERENCES Course(course\_id),

FOREIGN KEY(instructor\_id) REFERENCES Instructor(instructor\_id),

PRIMARY KEY (user\_id, course\_id, instructor\_id)

);

create Table Meeting\_Time(

meeting\_id INT,

section\_id INT,

day\_of\_week ENUM('M', 'T', 'W', 'TR', 'F', 'TBA') NOT NULL,

start\_time TIME,

end\_time TIME,

meeting\_location VARCHAR(40),

Primary Key (meeting\_id),

FOREIGN KEY (section\_id) REFERENCES Section(section\_id)

);

-- new Table to track average review

create Table Course\_Instructor(

course\_id VARCHAR(30),

instructor\_id INT,

review\_sum INT DEFAULT 0,

review\_count INT DEFAULT 0,

PRIMARY KEY (course\_id, instructor\_id),

FOREIGN KEY (course\_id, instructor\_id)

)

--INSERT DATA

INSERT INTO Users (user\_id, username, password\_hash) VALUES

(12394712, 'alice', 'alice1234!'),

(43529843, 'sam', 'sam1234!'),

(34597234, 'lisa', 'lisa1234!');

INSERT INTO Instructor (instructor\_id, first\_name, last\_name) VALUES

(213094198, 'David', 'Daniels'),

(142890983, 'Michael Joseph', 'Stahl'),

(391689143, 'Yitzchak', 'Schwartz');

INSERT INTO Course (course\_id, title, credits, course\_description, prerequisites) VALUES

('CSCI-UA 101', 'Intro to Computer Science', 4, 'Foundational course on cs.', NULL),

('CSCI-UA 102', 'Data Structures', 4, 'Arrays, linked lists, trees, graphs.', 'CSCI-UA 101'),

('MATH-UA 121', 'Calculus I', 4, 'Limits, derivatives, and integrals.', NULL);

INSERT INTO Section (section\_id, course\_id, instructor\_id, section\_type, campus) VALUES

(310947, 'CSCI-UA 101', 213094198, 'Lecture', 'Washington Square'),

(134867, 'CSCI-UA 101', 142890983, 'Lab', 'Washington Square'),

(458299, 'CSCI-UA 101', 391689143, 'Lecture', 'Washington Square');

INSERT INTO Meeting\_Time (meeting\_id, section\_id, day\_of\_week, start\_time, end\_time, meeting\_location) VALUES

(3001, 310947, 'W', '11:00:00', '12:15:00', 'SILV 406'),

(3002, 134867, 'W', '12:30:00', '13:45:00', 'TISC LC3'),

(3003, 458299, 'TR', '08:00:00', '09:15:00', 'SILV 206');

INSERT INTO User\_Selection (user\_id, course\_id, instructor\_id) VALUES

(12394712, 'CSCI-UA 101', 213094198),

(43529843, 'CSCI-UA 102', 142890983),

(34597234, 'MATH-UA 121', 391689143);

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at) VALUES

(12394712, 'CSCI-UA 101', 213094198, 5, 'amazing and helpful.', '2023-05-15 09:30:00'),

(43529843, 'MATH-UA 121', 142890983, 4, 'great lecturer, exams are tough.', '2023-05-16 14:22:00'),

(34597234, 'CSCI-UA 102', 391689143, 4, 'challenging but rewarding, professor was very supportive.', '2023-12-20 11:00:00');

INSERT INTO Users (user\_id, username, password\_hash) VALUES

(11111111, 'sky\_dragon', 'Skydragon4321?!'),

(22222222, 'flame\_demon', 'Flamedemon4321?'),

(33333333, 'guy', 'guy4321?');

INSERT INTO Instructor (instructor\_id, first\_name, last\_name) VALUES

(142857142, 'Amos', 'Bloomberg'),

(428571428, 'Adam', 'Meyers'),

(285714285, 'Ahmad', 'Emad');

INSERT INTO Course (course\_id, title, credits, course\_description, prerequisites) VALUES

('CSCI-UA 430', 'Agile Software Development and DevOps', 4, "Agile software development has come to describe a specific approach and toolset that allow for the requirements of a software project to change as a project progresses without disrupting schedules, budgets, and responsibilities. The field of DevOps, a portmanteau of development and operations has introduced further processes and infrastructure to automate many of the tasks required in such development. Together, Agile's methodology and DevOps' automation have increased the speed, robustness, and scalability with which software is developed today. Upon completion of this course, students will understand the core methodologies, technologies, and tools used in the software industry today.", 'CSCI-UA 201'),

('CSCI-UA 469', 'Natural Language Processing', 4, 'Natural Language Processing applies computational and linguistic knowledge to the processing of natural languages (English, Chinese, Spanish, Japanese). Applications include: machine translation, information extraction, information retrieval, and others. On the one hand, the class will include the modeling and representation of linguistic phenomena. On the other, it will cover methods for applying this knowledge using both manual rules and machine learning. Sample topics include: formal languages, hidden Markov models, part of speech tagging, vector-based methods, shallow and full parsing, semantic role labeling, information extraction and machine translation. Students will complete programming

assignments (POS-tagging, Information Extraction, etc.) and group final projects.', NULL),

('CSCI-UA 475', 'Predictive Analytics ', 4, 'Predictive analytics is the art and science of extracting useful information from historical data and present data for the purpose of predicting future trends. In this course, students will be introduced to the phases of the analytics life-cycle and will gain an understanding of a variety of tools and machine learning algorithms for analyzing data and discovering forward insights. Several techniques will be introduced including: data preprocessing techniques, data reduction algorithms, data clustering algorithms, data classification algorithms, uplifting algorithms, association rules, data mining algorithms, recommender systems, and more. This course aims to provide students with skills of the new generation of data scientists that will allow them to structure, analyze and derive useful insights from data that could help make better decisions.', NULL);

INSERT INTO Section (section\_id, course\_id, instructor\_id, section\_type, campus) VALUES

(111111, 'CSCI-UA 430',142857142, 'Lecture', 'Washington Square'),

(222222, 'CSCI-UA 469', 428571428, 'Lecture', 'Washington Square'),

(222223, 'CSCI-UA 469', 428571428, 'Lecture', 'Washington Square');

INSERT INTO Meeting\_Time (meeting\_id, section\_id, day\_of\_week, start\_time, end\_time, meeting\_location) VALUES

(1001, 111111, 'TR', '12:30:00', '13:45:00', 'Online'),

(1002, 222222, 'TR', '11:00:00', '12:15:00', 'Silv 405'),

(1003, 222223, 'TR', '14:00:00', '15:15:00', 'Silv 408');

INSERT INTO User\_Selection (user\_id, course\_id, instructor\_id) VALUES

(22222222, 'CSCI-UA 469', 428571428),

(22222222, 'CSCI-UA 430',142857142),

(11111111, 'CSCI-UA 430', 142857142);

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at) VALUES

(11111111, 'CSCI-UA 101', 213094198, 5, 'This course is very good. Would recommend', '2025-05-15 09:30:00'),

(22222222, 'MATH-UA 121', 142890983, 1, 'Terrible. A tremendous waste of time.', '2025-05-16 14:22:00'),

(33333333, 'CSCI-UA 469', 428571428, 4,'Decent course. I learned a lot..', '2025-10-20 11:00:00');

INSERT INTO Course (course\_id, title, credits, course\_description, prerequisites)

VALUES

('CS-UY 2204 | ECE-UY 2204', 'Digital Logic and State Machine Design', 4, 'This course covers combinational and sequential digital circuits. Topics: Introduction to digital systems. Number systems and binary arithmetic. Switching algebra and logic design. Error detection and correction. Combinational integrated circuits, including adders. Timing hazards. Sequential circuits, flipflops, state diagrams and synchronous machine synthesis. Programmable Logic Devices, PLA, PAL and FPGA. Finite-state machine design. Memory elements. ', 'CS-UY 1114(C- at least)or CS-UY 1133(C- at least)'),

('ECE-UY 345X', 'Undergraduate Research in Electrical and Computer Engineering', 1, 'The student will conduct research with the guidance of a faculty member. A written report is required. This course may be repeated for up to a maximum of 6 credits.', 'Contact rtoth@nyu.edu for permission'),

('ECE-UY 1002', 'Introduction to Electrical and Computer Engineering', 2, 'This course introduces numerous subject areas in Electrical and Computer Engineering (power systems, electronics, computer networking, microprocessors, digital logic, embedded systems, communications, feedback control, and signal processing). ', 'First-year standing');

INSERT INTO Instructor (instructor\_id, first\_name, last\_name)

VALUES

(28934, 'Azeez', 'Bhavnagarwala'),

(38394, 'TBD', 'TBD'),

(39493, 'Matthew', 'Campisi');

INSERT INTO Section (section\_id, course\_id, instructor\_id, section\_type, campus)

VALUES

(12243, 'CS-UY 2204 | ECE-UY 2204', 28934, 'Lecture', 'Brooklyn Campus'),

(12239, 'CS-UY 2204 | ECE-UY 2204', 28934, 'Lab', 'Brooklyn Campus'),

(12240, 'CS-UY 2204 | ECE-UY 2204', 28934, 'Lab', 'Brooklyn Campus'),

(12241, 'CS-UY 2204 | ECE-UY 2204', 28934, 'Lab', 'Brooklyn Campus'),

(12275, 'ECE-UY 1002', 39493, 'Lecture', 'Brooklyn Campus');

INSERT INTO Meeting\_Time (meeting\_id, section\_id, day\_of\_week, start\_time, end\_time, meeting\_location)

VALUES

(73483, 12243, 'T', '14:00:00', '15:50:00', 'Jacobs Hall 6 Metrotech RM 474'),

(273984, 12243, 'T', '14:00:00', '15:50:00', 'Jacobs Hall 6 Metrotech RM 474'),

(172384, 12239, 'W', '14:00:00', '16:50:00', 'Jacobs Hall 6 Metrotech RM 227'),

(238949, 12240, 'TR', '17:00:00', '19:50:00', 'Jacobs Hall 6 Metrotech RM 227'),

(738648, 12241, 'F', '14:00:00', '16:50:00', 'Jacobs Hall 6 Metrotech RM 227'),

(253673, 12275, 'T', '14:00:00', '15:20:00', '5 MetroTech Center Room AUD'),

(253674, 12275, 'TR', '14:00:00', '15:20:00', '5 MetroTech Center Room AUD');

INSERT INTO Users (user\_id, username, password\_hash)

VALUES

(83873, 'vickyA', '344a0796ca0712be797c8bc103ea12a6'),

(83874, 'vickyB', '102ff968587291ba27ef40cac6e8b65f'),

(83875, 'vickyC', '505b2486f7317215f28da6c419bbdeb1');

INSERT INTO User\_Selection (user\_id, course\_id, instructor\_id)

VALUES

(83873, 'CS-UY 2204 | ECE-UY 2204', 28934),

(83873, 'ECE-UY 1002', 39493),

(83874, 'ECE-UY 345X', 38394);

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at)

VALUES

(83873, 'CS-UY 2204 | ECE-UY 2204', 28934, 5, 'I like this course', '2025-11-07 11:59:00'),

(83874, 'CS-UY 2204 | ECE-UY 2204', 28934, 1, 'terrible', '2025-11-07 09:59:00'),

(83873, 'ECE-UY 1002', 39493, 3, 'nice course, but hard', '2025-11-07 10:32:00');

INSERT INTO Course (course\_id, title, credits, course\_description, prerequisites)

VALUES

('CS-UY 4543', 'Human Computer Interaction', 3,

'Designing a successful interactive experience or software system takes more than technical savvy and vision--it also requires a deep understanding of how to serve people''s needs and desires through the experience of the system, and knowledge about how to weave this understanding into the development process. This course introduces key topics and methods for creating and evaluating human-computer interfaces/digital user experiences. Students apply these practices to a system of their choosing. (I encourage application to prototype systems that students are currently working on in other contexts, at any stage of development). The course builds toward a final write-up and presentation in which students detail how they tackled HCI/user experience design and evaluation of their system, and results from their investigations. Some experience creating/participating in the production of interactive experiences/software is recommended.', 'None'

);

INSERT INTO Instructor (instructor\_id, first\_name, last\_name)

VALUES

(45431, 'Raymond', 'Lutzky'),

(45432, 'Nitesh', 'Goyal');

INSERT INTO Section (section\_id, course\_id, instructor\_id, section\_type, campus)

VALUES

(454311, 'CS-UY 4543', 45431, 'Lecture', 'Brooklyn Campus'),

(454312, 'CS-UY 4543', 45432, 'Lecture', 'Brooklyn Campus');

INSERT INTO Meeting\_Time (meeting\_id, section\_id, day\_of\_week, start\_time, end\_time, meeting\_location)

VALUES

(4543111, 454311, 'TBA', NULL, NULL, 'Online'),

(4543121, 454312, 'W', '17:00:00', '19:30:00', 'Jacobs Hall, 6 Metrotech Room 674');

INSERT INTO Users (user\_id, username, password\_hash)

VALUES

(83876, 'aronD', 'c81b9e830f6a29f82631626f376451e5'),

(83877, 'aronE', 'e8f3b2a5d1c4f9e0b7a6d5c4b3a2d1e0'),

(83878, 'aronF', 'a1b2c3d4e5f678901234567890abcdef');

INSERT INTO User\_Selection (user\_id, course\_id, instructor\_id)

VALUES

(83876, 'CS-UY 4543', 45432),

(83877, 'ECE-UY 1002', 39493),

(83878, 'CS-UY 2204 | ECE-UY 2204', 28934);

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at)

VALUES

(83876, 'CS-UY 4543', 45432, 4, 'The online component was flexible.', '2025-01-25 15:00:00'),

(83877, 'ECE-UY 1002', 39493, 5, 'Very engaging instructor.', '2025-02-10 09:00:00'),

(83878, 'CS-UY 2204 | ECE-UY 2204', 28934, 3, 'Solid material.', '2025-02-15 14:00:00');

INSERT INTO Course (course\_id, title, credits, course\_description, prerequisites)

VALUES

('CS-UY 3224 | CS-UY 3224G', 'Intro to Operating Systems', 4,

'This course studies the fundamental concepts and principles of operating systems. Batch, spooling and multiprogramming systems are introduced. The parts of an operating system are described in terms of their functions, structure and implementation. Basic policies for allocating resources are discussed.',

'Prerequisites for Brooklyn Students: CS-UY 2214 AND (CS-UY 2134 or CS-UY 1134) AND (CS-UY 2124 or CS-UY 1124) (C- or better). | Prerequisite for Abu Dhabi Students: (ENGR-UH 3510 or CS-UH 1050) (C- or better) AND (CS-UH 2010 or ENGR-UH 3511) | Prerequisites for Shanghai Students: CSCI-SHU 210 (C- or better) AND CENG-SHU 202'

);

INSERT INTO Users (user\_id, username, password\_hash)

VALUES

(83879, 'aronG', '1f3e7c2d9a6b58e4f0d9c8b7a6e5f4d3'),

(83880, 'aronH', '2e4d8a1c6b9e0f5d7a8b9c0d1e2f3a4b'),

(83881, 'aronI', '3a5c9b2d7e0f1a6d8b9c0e1f2a3b4c5d');

INSERT INTO User\_Selection (user\_id, course\_id, instructor\_id)

VALUES

(83879, 'CS-UY 3224 | CS-UY 3224G', 32241),

(83880, 'CS-UY 3224 | CS-UY 3224G', 32241),

(83881, 'CS-UY 3224 | CS-UY 3224G', 32241);

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at)

VALUES

(83879, 'CS-UY 3224 | CS-UY 3224G', 32241, 5, 'Excellent content.', '2025-03-01 10:00:00'),

(83880, 'CS-UY 3224 | CS-UY 3224G', 32241, 2, 'The pace was too fast.', '2025-03-05 14:30:00'),

(83881, 'CS-UY 3224 | CS-UY 3224G', 32241, 4, 'Requires a lot of background knowledge.', '2025-03-10 16:00:00');

INSERT INTO Course (course\_id, title, credits, course\_description, prerequisites)

VALUES

('CS-UY 4793 | CS-UY 4793G', 'Computer Networking', 3,

'This course takes a top-down approach to computer networking. After an overview of computer networks and the Internet, the course covers the application layer, transport layer, network layer and link layers. Topics at the application layer include client-server architectures, P2P architectures, DNS and HTTP and Web applications. Topics at the transport layer include multiplexing, connectionless transport and UDP, principles or reliable data transfer, connection-oriented transport and TCP and TCP congestion control. Topics at the network layer include forwarding, router architecture, the IP protocol and routing protocols including OSPF and BGP. Topics at the link layer include multiple-access protocols, ALOHA, CSMA/CD, Ethernet, CSMA/CA, wireless 802.11 networks and link-layer switches. The course includes simple quantitative delay and throughput modeling, socket programming and network application development and Ethereal labs.',

'Prerequisite for Brooklyn Students: (CS-UY 2134 or CS-UY 1134) and (CS-UY 2124 or CS-UY 1124) (C- or better) | Prerequisite for Abu Dhabi Students: ENGR-UH 3510 or CS-UH 1050 (C- or better) | Prerequisite for Shanghai Students: CSCI-SHU 210 (C- or better)'

);

INSERT INTO Instructor (instructor\_id, first\_name, last\_name)

VALUES

(47931, 'Lucas', 'O''Rourke');

INSERT INTO Section (section\_id, course\_id, instructor\_id, section\_type, campus)

VALUES

(479311, 'CS-UY 4793 | CS-UY 4793G', 47931, 'Lecture', 'Brooklyn Campus');

INSERT INTO Meeting\_Time (meeting\_id, section\_id, day\_of\_week, start\_time, end\_time, meeting\_location)

VALUES

(4793111, 479311, 'M', '17:00:00', '18:20:00', '2 MetroTech Center Room 801'),

(4793112, 479311, 'W', '17:00:00', '18:20:00', '2 MetroTech Center Room 801');

INSERT INTO Users (user\_id, username, password\_hash)

VALUES

(83882, 'aronJ', 'd4e2f1c3b5a90786543210fedcba9876'),

(83883, 'aronK', 'a7b8c9d0e1f2345678901234567890ab'),

(83884, 'aronL', 'f0e9d8c7b6a543210123456789abcdef');

INSERT INTO User\_Selection (user\_id, course\_id, instructor\_id)

VALUES

(83882, 'CS-UY 4793 | CS-UY 4793G', 47931),

(83883, 'CS-UY 4793 | CS-UY 4793G', 47931),

(83884, 'CS-UY 4793 | CS-UY 4793G', 47931);

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at)

VALUES

(83882, 'CS-UY 4793 | CS-UY 4793G', 47931, 5, 'The networking labs were insightful.', '2025-04-01 11:30:00'),

(83883, 'CS-UY 4793 | CS-UY 4793G', 47931, 4, 'Instructor explained TCP congestion control very clearly.', '2025-04-05 09:00:00'),

(83884, 'CS-UY 4793 | CS-UY 4793G', 47931, 3, 'The theory felt a bit heavy at times.', '2025-04-10 15:45:00');

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at)

VALUES

(83882, 'CSCI-UA 430', 142857142, 5, 'generous in grade, nice prof', '2025-05-01 18:30:00'),

(83883, 'CSCI-UA 430', 142857142, 4, 'somewhat okay', '2025-03-05 08:20:00');

B)

DELIMITER $$

CREATE OR REPLACE PROCEDURE RegisterNewUser(

IN input\_username VARCHAR(20),

IN input\_password\_hash VARCHAR(255)

)

BEGIN

IF EXISTS (SELECT 1 FROM User WHERE username = input\_username) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Username already exists.';

ELSE

INSERT INTO Users (username, password\_hash)

VALUES (input\_username, input\_password\_hash);

SELECT LAST\_INSERT\_ID() AS new\_user\_id;

END IF;

END $$

DELIMITER ;

DELIMITER $$

CREATE OR REPLACE PROCEDURE CheckUserLogin(

IN input\_username VARCHAR(20),

IN input\_password\_hash VARCHAR(255)

)

BEGIN

DECLARE found\_user\_id INT;

SELECT user\_id INTO found\_user\_id

FROM Users

WHERE username = input\_username AND password\_hash = input\_password\_hash

LIMIT 1;

SELECT found\_user\_id AS authenticated\_user\_id;

END $$

DELIMITER ;

DELIMITER $$

CREATE OR REPLACE PROCEDURE AddUserSelection(

IN p\_user\_id INT,

IN p\_course\_id VARCHAR(30),

IN p\_instructor\_id INT

)

BEGIN

DECLARE existing\_count INT DEFAULT 0;

SELECT COUNT(\*)

INTO existing\_count

FROM User\_Selection

WHERE user\_id = p\_user\_id

AND course\_id = p\_course\_id

AND instructor\_id = p\_instructor\_id;

IF existing\_count = 0 THEN

INSERT INTO User\_Selection (user\_id, course\_id, instructor\_id)

VALUES (p\_user\_id, p\_course\_id, p\_instructor\_id);

SELECT 'Successful' AS result;

ELSE

SELECT 'Already exist' AS result;

END IF;

END $$

DELIMITER ;

DELIMITER $$

DELIMITER $$

CREATE OR REPLACE PROCEDURE RemoveUserSelection(

IN p\_user\_id INT,

IN p\_course\_id VARCHAR(30),

IN p\_instructor\_id INT

)

BEGIN

DECLARE existing\_count INT DEFAULT 0;

SELECT COUNT(\*)

INTO existing\_count

FROM User\_Selection

WHERE user\_id = p\_user\_id

AND course\_id = p\_course\_id

AND instructor\_id = p\_instructor\_id;

IF existing\_count > 0 THEN

DELETE FROM User\_Selection

WHERE user\_id = p\_user\_id

AND course\_id = p\_course\_id

AND instructor\_id = p\_instructor\_id;

SELECT 'Successful';

ELSE

SELECT 'Not Found';

END IF;

END$$

CREATE OR REPLACE PROCEDURE SearchReview(

IN course\_id VARCHAR(30),

IN instructor\_id INT

)

BEGIN

SELECT

r.rating,

r.comment,

r.created\_at,

(

SELECT title

FROM Course c

WHERE c.course\_id = r.course\_id

) AS title,

(

SELECT credits

FROM Course c

WHERE c.course\_id = r.course\_id

) AS credits,

(

SELECT course\_description

FROM Course c

WHERE c.course\_id = r.course\_id

) AS course\_description,

(

SELECT CONCAT(i.first\_name, ' ', i.last\_name)

FROM Instructor i

WHERE i.instructor\_id = r.instructor\_id

) AS instructor\_name

FROM Review r

WHERE r.course\_id = p\_course\_id

AND r.instructor\_id = p\_instructor\_id

ORDER BY r.created\_at DESC;

END$$

DELIMITER ;

DELIMITER $$

CREATE OR REPLACE PROCEDURE PostReview(

IN user\_id INT,

IN course\_id VARCHAR(30),

IN instructor\_id INT,

IN rating INT,

IN comment TEXT,

IN created\_at TIMESTAMP

)

BEGIN

INSERT INTO Review (user\_id, course\_id, instructor\_id, rating, comment, created\_at)

VALUES (user\_id, course\_id, instructor\_id, rating, comment, created\_at);

END$$

DELIMITER ;

DELIMITER $$

CREATE OR REPLACE PROCEDURE SearchCourse(

IN u\_course\_id VARCHAR(30),

IN u\_course\_title VARCHAR(255),

IN u\_instructor\_name VARCHAR(200)

)

BEGIN

SELECT

c.title,

c.credits,

c.course\_description,

c.prerequisites,

s.section\_type,

s.campus,

m.day\_of\_week,

m.start\_time,

m.end\_time,

m.meeting\_location,

i.first\_name,

i.last\_name

FROM Course c

INNER JOIN Section s ON c.course\_id = s.course\_id

INNER JOIN Instructor i ON s.instructor\_id = i.instructor\_id

INNER JOIN Meeting\_Time m ON s.section\_id = m.section\_id

WHERE (u\_course\_id IS NULL OR u\_course\_id = '' OR c.course\_id LIKE CONCAT('%', u\_course\_id, '%'))

AND (u\_course\_title IS NULL OR u\_course\_title = '' OR c.title LIKE CONCAT('%', u\_course\_title, '%'))

AND (u\_instructor\_name IS NULL OR u\_instructor\_name = '' OR CONCAT(i.first\_name, ' ', i.last\_name) LIKE CONCAT('%', u\_instructor\_name, '%'))

ORDER BY c.course\_id, i.instructor\_id, s.section\_id, m.day\_of\_week;

END$$

DELIMITER ;

Using of Trigger:

DELIMITER $$

CREATE OR REPLACE TRIGGER review\_insert\_upsert\_trigger

AFTER INSERT ON review

FOR EACH ROW

BEGIN

INSERT INTO course\_instructor

(course\_id, instructor\_id, review\_sum, review\_count)

VALUES

(NEW.course\_id, NEW.instructor\_id, NEW.rating, 1)

ON DUPLICATE KEY UPDATE

review\_sum = review\_sum + NEW.rating,

review\_count = review\_count + 1;

END$$

DELIMITER ;