



UNIVERSITETI I TIRANËS  
Fakulteti i Shkencave të Natyrës

# LEARNING MANAGEMENT SYSTEM (LMS)



Lënda: “Databazë nën Oracle”

Punoi:

Imelda Malçellari

Ina Bektashi

Irisa Jani

Jona Malaveci

Matias Poshnjari

Mikaela Tasi

Bachelor, Informatikë, B1  
**Viti akademik 2023-2024**





**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

***Përmbajtja***

<u>1.1.Përshkrimi i problemit.....</u>	<u>2</u>
<u>1.2.Modelimi i diagramës ER/EER.....</u>	<u>3</u>
<u>1.2.1.Shpjegimi i modelit ER/EER.....</u>	<u>4</u>
<u>2.1.Modelimi i Skemës Relacionale.....</u>	<u>10</u>
<u>2.1.1.Shpjegimi i diagramës Racionale.....</u>	<u>11</u>
<u>2.2.Tabelat.....</u>	<u>12</u>
<u>2.3.SQL Queries.....</u>	<u>13</u>
<u>2.3.1. Create.....</u>	<u>13</u>
<u>2.3.2. Insert.....</u>	<u>19</u>
<u>2.3.3. Select.....</u>	<u>29</u>
<u>2.3.4. Update.....</u>	<u>32</u>
<u>2.3.4. Delete.....</u>	<u>36</u>



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

#### ❖ Përshkrimi i problemit

Qëllimi i një Sistemi Menaxhimi Studimi (Learning Management System - LMS) është ofrimi i mësimdhënies dhe trajnimeve që përmban kursi në mënyrë sa më të organizuar, të lehta për tu aksesuar si dhe efikase. Kjo bëhet nga një sistem i vetëm, i cili bën të mundur ruajtjen dhe menaxhimin në kohë reale të materialeve të mësimin dhe personave të cilët janë regjistruar në secilin prej kurseve. Gjithashtu sistemi bën të mundur monitorimin e cdo individi që është regjistruar në një prej kurseve të ofruara, ku cdo informacion që lidhet me progresin e individit ruhet për të pasur një pasqyrë sa më të qartë të arritjeve në fund të përfundimit të kursit. Mbi të gjitha sistemi ruan informacione në varësi të statusit të individit që identifikohet duke pasur larmishmëri për cdo lloj user-i.

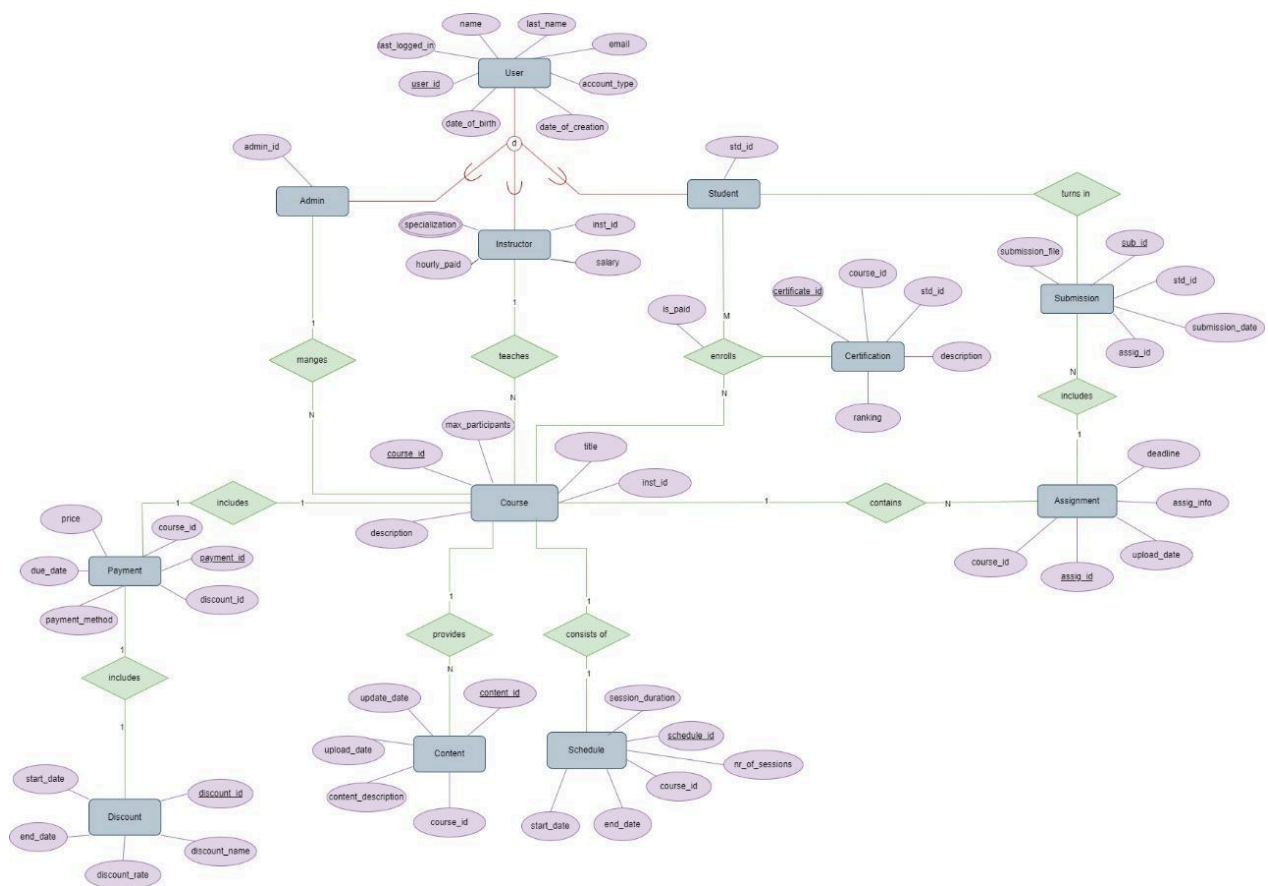
Sistemi është i ndërtuar për cdo individ i cili kërkon të zgjerojë njohuri të reja në gjuhë të ndryshme të programimit. Këta individë mund të jenë student të cdo moshe. Sistemi është i pershtatshëm për tu përdorur edhe nga persona të cilët kanë aftësi mësimdhënie dhe që dëshirojnë të trajnojnë individë të tjerë duke ofruar eksperiencën e tyre. Gjithashtu përdoret nga individë specifike të cilët kanë nën menaxhim një nga qendrat mësimore të sistemit.

Problemi që kjo databazë zgjidh, është ruajtja e informacionit pa pasur mungesë, ku secili prej user-ave të këtij sistemi mund të ketë lehtësisht akses mbi këto informacione. Informacionet mund të jenë të tipeve të ndryshme si: oraret e seancave mësimore dhe kohëzgjatja e tyre, detyra të ndryshme, progresi i pjesëmarrësve të kurseve, materiale të ndryshme të ngarkuara në sistem, certifikimet, pagesa, uljet për cdo kurs.



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

❖ **Modelimi i diagrames ER/EER**





**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

❖ **Shpjegimi i modelit ER/EER**

**Entiteti: User**

*Attribute:*

- user\_id
- first\_name
- last\_name
- account\_type
- email
- date\_of\_birth
- date\_of\_creation
- last\_loggedIn

**Entiteti: Instructor**

*Attribute:*

- inst\_id
- specialization
- hourly\_paid
- salary

**Entiteti: Student**

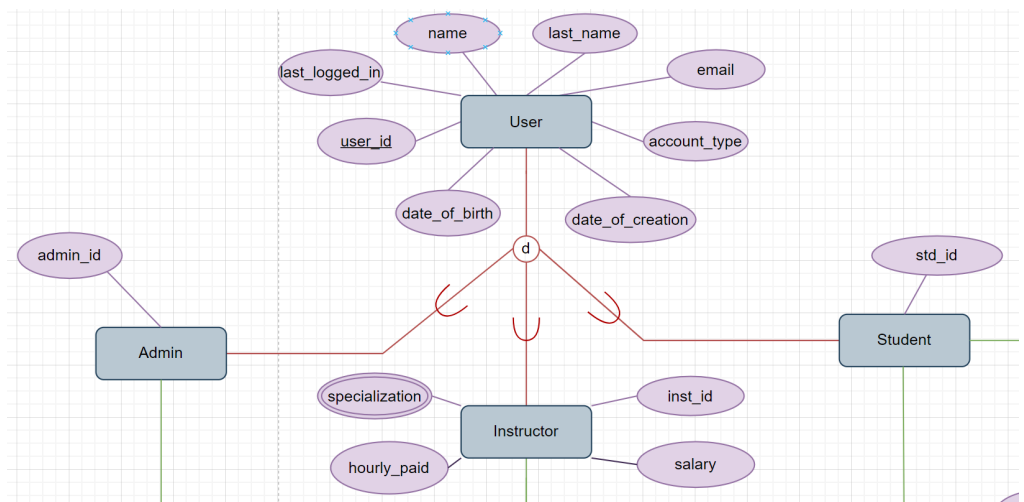
*Attribute:*

- std\_id

**Entiteti: Admin**

*Attribute:*

- admin\_id





## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

Entiteti User shërben si superklasë për të gjitha llojet e përdoruesve (instructor, student, admin) dhe përmban atributet e përbashkëta.

Entitetet Admin, Instructor dhe Student janë entitete të specializuara nga entiteti User. Pra ato përmbajnë atributet e entitetit User dhe atributet e deklaruar nën to.

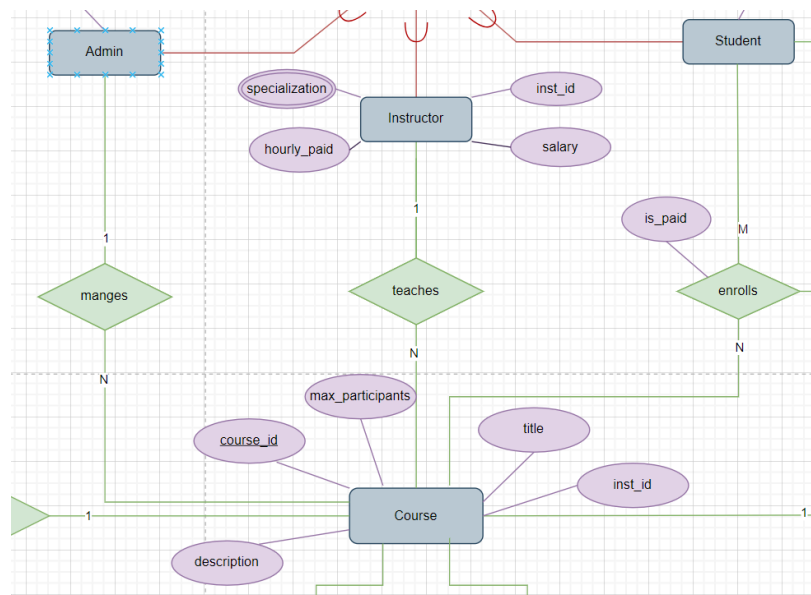
Ky lloj specializim është i tipit disjoint total, pra nënkupton që entiteti user mund të jetë Admin, Instructor ose Student por nuk mund ti përkasë më shumë se një tipi në të njëjtën kohë.

#### **Entiteti: Course**

##### *Attribute:*

- course\_id
- title
- description
- max\_participants
- inst\_id

Entiteti "Course" në këtë bazë të të dhënave shërben për të ruajtur informacionin e lidhur me kurset që ofrohen në platformën e LMS.



Lidhjet në entitetin Course janë si më poshtë:

1. Admin ka një lidhje “manages” me Course, të tipit 1:N, pra një admin ka shumë kurse që menaxhon në këtë LMS.
2. Instructor ka një lidhje “teaches” me Course, të tipit 1:N, pra një instruktur ka disa kurse të cilat i zhvillon.
3. Student ka një lidhje “enrolls” me Course, të tipit 1:N, pra një student mund të regjistrohet në shumë kurse.



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

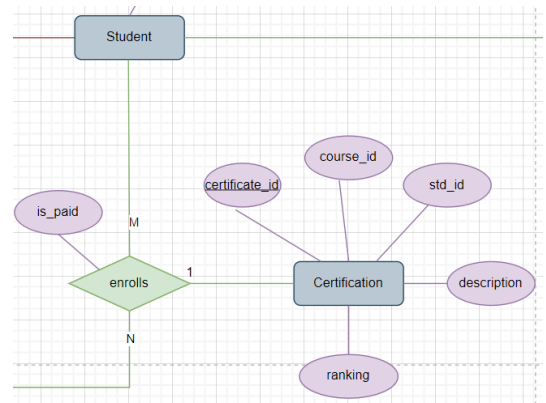
#### ***Entiteti: Certification***

##### *Atribute:*

- certification\_id
- course\_id
- std\_id
- description
- ranking

Entiteti Certification ruan informacion për cdo regjistrim që bëhet nga studenti për cdo kurs.

Certification lidhet me “enrolls”, duke shprehur që për cdo regjistrim që krijohet nga lidhjet e entiteteve Student dhe Course do krijohet një Certifikatë për atë regjistrim.

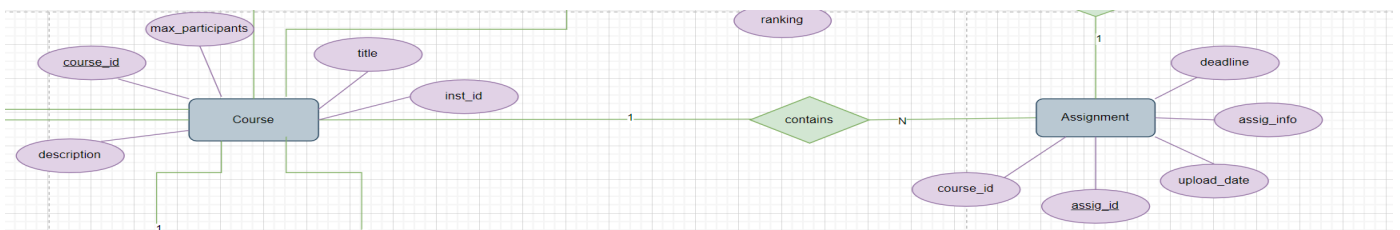


#### ***Entiteti: Assignment***

Entiteti Assignment lidhet me relation “contains” 1:N me Course.

##### *Atribute:*

- assig\_id
- deadline
- assig\_info
- upload\_date
- course\_id (Foreign Key)





# UNIVERSITETI I TIRANËS

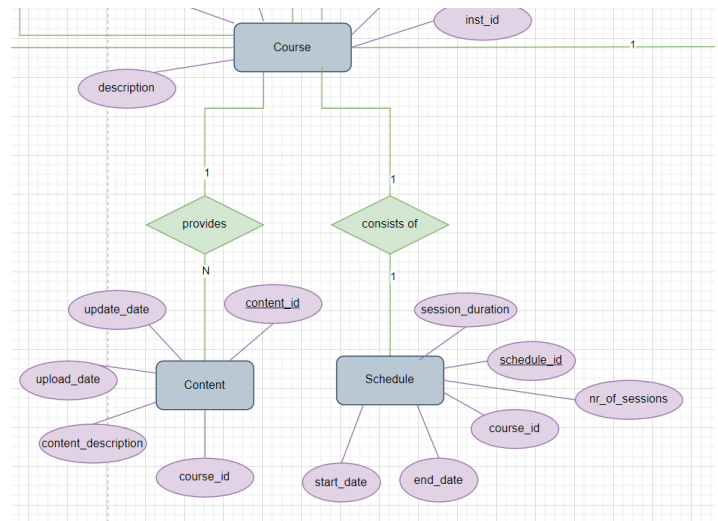
## Fakulteti i Shkencave të Natyrës

### Entiteti: Schedule

Entiteti Course ka një lidhje “consists of” 1:N me entitetin Schedule.

*Attribute:*

- schedule\_id
- start\_date
- end\_date
- nr\_of\_sessions
- session\_duration
- course\_id

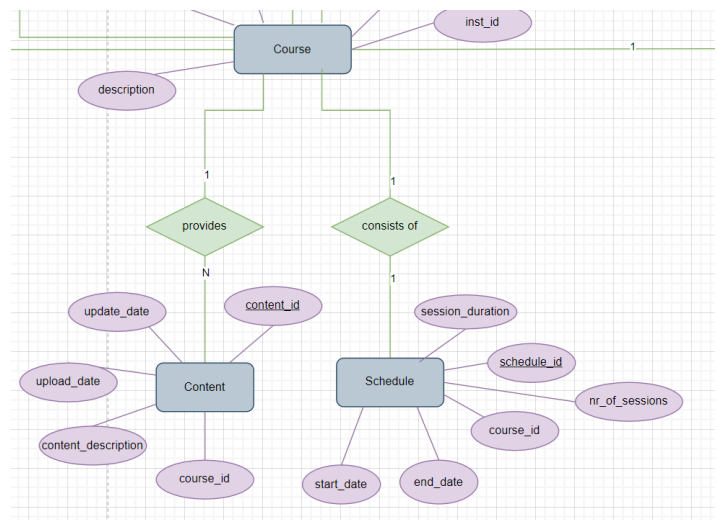


### Entiteti: Content

Entiteti Course ka lidhje “provides” 1: N me entitetin Content.

*Attribute:*

- content\_id
- content\_description
- course\_id
- upload\_date
- update\_date







## UNIVERSITETI I TIRANËS

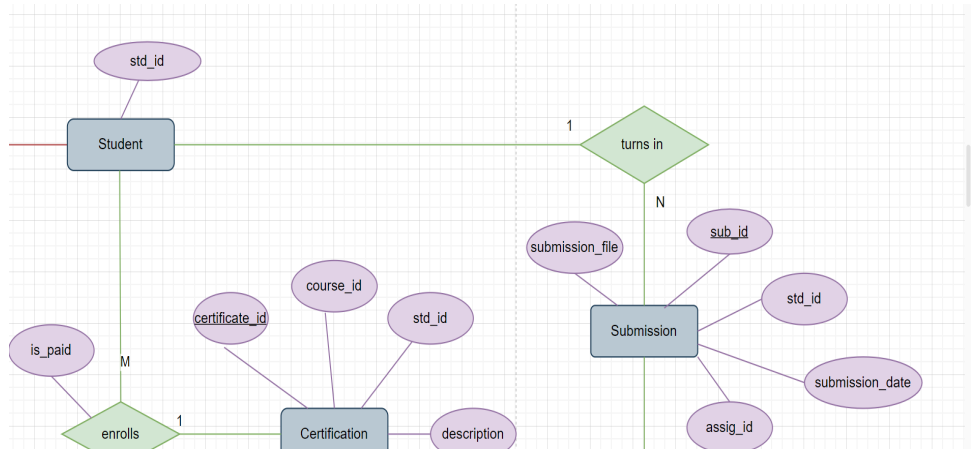
### Fakulteti i Shkencave të Natyrës

#### **Entiteti: Submission**

Entiteti student ka lidhje “turns in” 1:N Submission

*Attribute:*

- sub\_id
- submission\_file
- std\_id
- submission\_date
- assig\_id

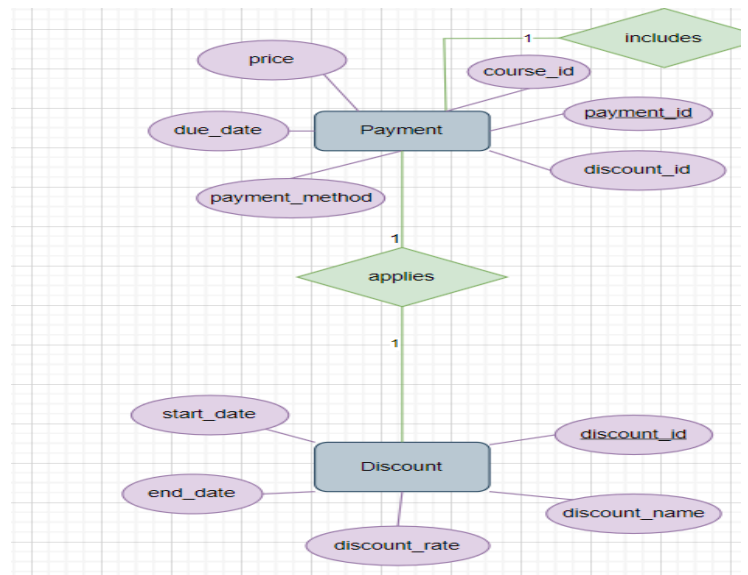


#### **Entiteti: Discount**

Entiteti payment ka lidhje “applies” 1:1 me Discount.

*Attribute:*

- discount\_id
- discount\_name
- discount\_rate
- start\_date
- end\_date





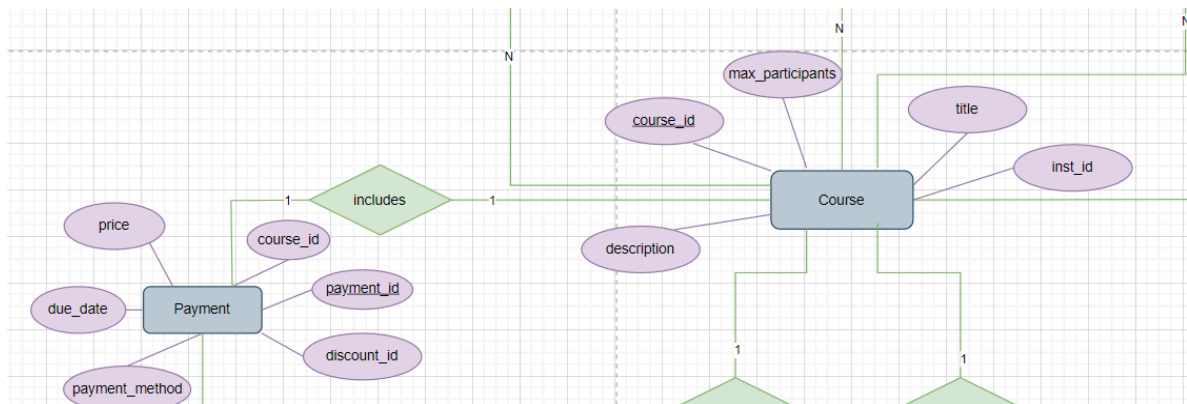
**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

***Entiteti: Payment***

Entiteti Course ka lidhje include 1:1 me entitetin payment.

*Attribute:*

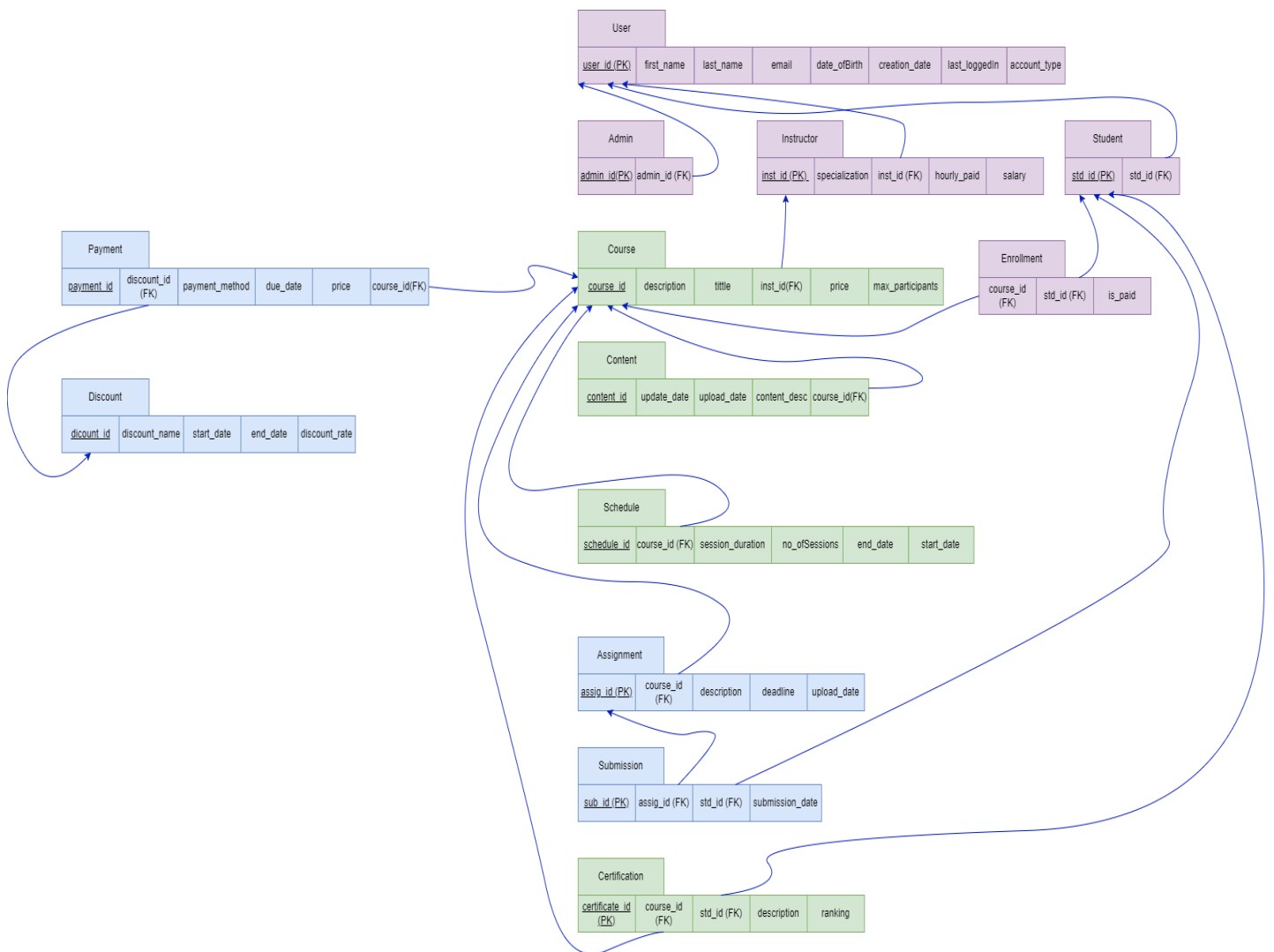
- payment\_id
- discount\_id
- course\_id
- due\_date
- payment\_method





**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

❖ **Modelimi i Skemës Relacionale**





## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

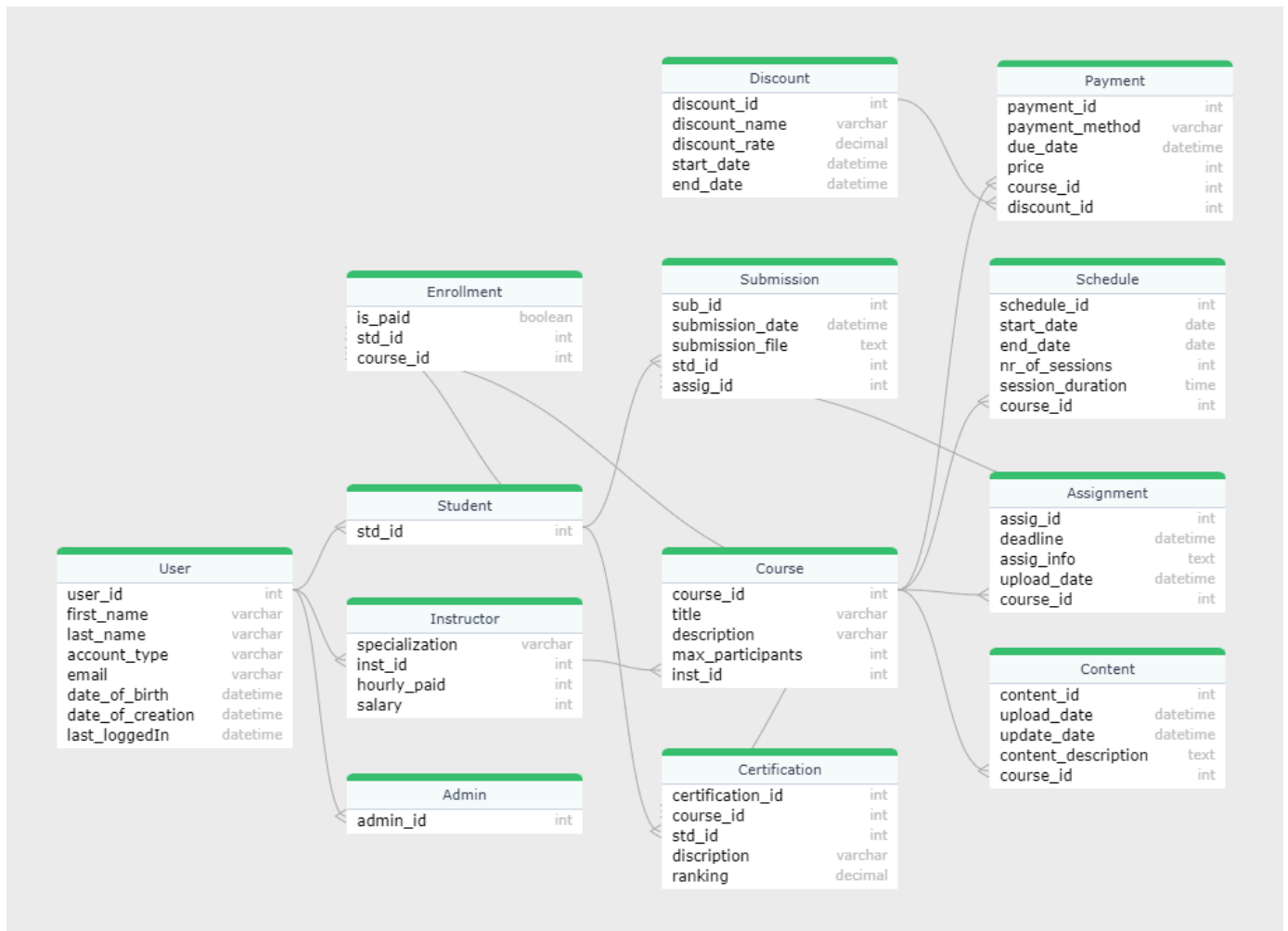
#### ❖ Shpjegimi i diagramës Racionale

1. Identifikojmë entitetet nga diagrama EER. Në rastin tonë kemi entitete të regullta, superklasë dhe nënklasa. Kemi ndertuar nje relacione te vecante R për entitetet që përfshijnë të gjithë atributet E. Gjithashtu kemi zgjedhur nje çelës te E si çelës primar te R.
2. Nuk kemi entitete te dobëta ne këtë diagramë EER.
3. Krijuam relacione te lidhjeve binare te tipit 1:1 duke zgjedhur te ruajmë nje çelës te huaj tek nje prej tabelave që ndodhen në këtë lidhje.
4. Krijuam relacione te lidhjeve binare te tipit 1:N. Ku për cdo lidhje binare R te tipit 1:N identifikuam relacionin S që përfaqëson tipin e entitetit të anës N të lidhjes. Me pas përfshimë si çelës të jashtëm në S çelsin primar te relacionit T që perfaqeson tipin tjetër të entitetit që merr pjesë në relacionin R.
5. Krijuam relacione te lidhjeve binare R te tipit N:M. Me pas krijuam nje relacion te ri S që perfaqëson R dhe përfshimë si çeles te jashtëm në S, çelsat primarë të relacioneve që marrin pjesë në lidhje dhe kombinimet e tyre do të formojnë çelsin primar të S.



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

❖ **Tabelat**





**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

❖ **SQL Queries**

➤ **CREATE**

```
CREATE DATABASE LMS;
```

```
use LMS;
```

```
CREATE TABLE IF NOT EXISTS User (  
    user_id INT AUTO_INCREMENT PRIMARY KEY,  
    first_name VARCHAR(16) NOT NULL,  
    last_name VARCHAR(16),  
    account_type VARCHAR(16) NOT NULL,  
    email VARCHAR(32),  
    date_of_birth DATETIME,  
    date_of_creation DATETIME NOT NULL,  
    last_loggedIn DATETIME NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS Instructor (  
    specialization VARCHAR(64),  
    inst_id INT,  
    FOREIGN KEY (inst_id) REFERENCES User(user_id) ON DELETE CASCADE );
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

```
CREATE TABLE IF NOT EXISTS Student (  
    std_id INT,  
    FOREIGN KEY (std_id) references User (user_id) ON DELETE CASCADE;  
);  
  
CREATE TABLE IF NOT EXISTS Admin(  
    admin_id INT PRIMARY KEY,  
    FOREIGN KEY (admin_id) references User (user_id)  
);  
  
CREATE TABLE IF NOT EXISTS Course (  
    course_id INT AUTO_INCREMENT PRIMARY KEY,  
    title VARCHAR(64) NOT NULL,  
    description VARCHAR(1024),  
    max_participants INT,  
    inst_id INT,  
    FOREIGN KEY (inst_id) REFERENCES Instructor(inst_id) ON DELETE  
    CASCADE  
);
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

```
CREATE TABLE IF NOT EXISTS Enrollment (  
    is_paid BOOLEAN,  
    std_id INT,  
        FOREIGN KEY (std_id) REFERENCES student(std_id) ON DELETE  
CASCADE,  
    course_id INT,  
    FOREIGN KEY (course_id) REFERENCES course(course_id) ON DELETE  
CASCADE  
);
```

```
CREATE TABLE IF NOT EXISTS Certification (  
    certification_id INT AUTO_INCREMENT PRIMARY KEY,  
    course_id INT,  
    std_id INT,  
    description VARCHAR(512),  
    ranking DECIMAL(5,2) NOT NULL,  
    FOREIGN KEY (course_id) REFERENCES Course(course_id) ON DELETE SET  
NULL,  
    FOREIGN KEY (std_id) REFERENCES Student(std_id) ON DELETE SET NULL  
);
```





**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

```
CREATE TABLE IF NOT EXISTS Assignment(
    assig_id INT PRIMARY KEY AUTO_INCREMENT,
    deadline DATETIME NOT NULL,
    assig_info TEXT NOT NULL,
    upload_date DATETIME NOT NULL,
    course_id INT,
    FOREIGN KEY (course_id) REFERENCES Course(course_id) ON DELETE
    CASCADE
);
```

```
CREATE TABLE IF NOT EXISTS Submission(
    sub_id INT PRIMARY KEY AUTO_INCREMENT,
    submission_date DATETIME NOT NULL,
    submission_file TEXT NOT NULL,
    std_id INT NOT NULL,
    assig_id INT NOT NULL,
    FOREIGN KEY (std_id) REFERENCES Student(std_id) ON DELETE CASCADE,
    FOREIGN KEY ( assig_id) REFERENCES Assignment( assig_id) ON DELETE
    CASCADE
);
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

```
CREATE TABLE IF NOT EXISTS Schedule(  
    schedule_id INT AUTO_INCREMENT PRIMARY KEY,  
    start_date date,  
    end_date date,  
    nr_of_sessions INT,  
    session_duration TIME,  
    course_id INT,  
    FOREIGN KEY(course_id) references Course(course_id) ON DELETE  
    CASCADE  
);
```

```
CREATE TABLE IF NOT EXISTS Content (  
    content_id INT AUTO_INCREMENT PRIMARY KEY,  
    upload_date datetime NOT NULL,  
    update_date datetime NOT NULL,  
    content_description TEXT NOT NULL,  
    course_id INT,  
    FOREIGN KEY(course_id) references Course(course_id) ON DELETE  
    CASCADE  
);
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

```
CREATE TABLE IF NOT EXISTS Discount (  
    discount_id INT AUTO_INCREMENT PRIMARY KEY,  
    discount_name VARCHAR(32),  
    discount_rate DECIMAL(5,2) NOT NULL DEFAULT 0,  
    start_date DATETIME,  
    end_date DATETIME  
);  
  
CREATE TABLE IF NOT EXISTS Payment (  
    payment_id INT AUTO_INCREMENT PRIMARY KEY,  
    payment_method VARCHAR(16) NOT NULL,  
    due_date DATETIME NOT NULL,  
    price INT NOT NULL,  
    course_id INT,  
    discount_id INT,  
    FOREIGN KEY (course_id) REFERENCES Course(course_id) ON DELETE  
    CASCADE,  
    FOREIGN KEY (discount_id) REFERENCES Discount(discount_id) ON DELETE  
    SET NULL  
);
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

#### ➤ INSERT

```
INSERT INTO User (first_name, last_name, account_type, email,  
date_of_birth, date_of_creation, last_loggedIn)
```

```
VALUES
```

```
    ('John', 'Doe', 'admin', 'john.doe@example.com', '1990-01-15  
12:00:00', '2022-01-01 08:00:00', '2022-01-05 15:30:00'),  
  
    ('Jane', 'Smith', 'student', 'jane.smith@example.com', '2006-01-01  
10:30:00', '2022-01-02 09:15:00', '2022-01-06 14:45:00'),  
  
    ('Michael', 'Johnson', 'student', 'michael.j@example.com',  
    '1997-01-01 15:45:00', '2022-01-03 13:30:00', '2022-01-07  
11:00:00'),  
  
    ('Emily', 'Williams', 'student', 'emily.w@example.com', '1994-01-01  
08:20:00', '2022-01-04 17:45:00', '2022-01-08 10:20:00'),  
  
    ('Sophia', 'Anderson', 'student', 'sophia.a@example.com', '1997-01-01  
14:10:00', '2022-01-05 09:30:00', '2022-01-09 12:45:00'),  
  
    ('Adam', 'Brown', 'student', 'adam.b@example.com', '1995-01-01  
16:45:00', '2022-01-10 11:15:00', '2022-01-10 16:30:00'),  
  
    ('Mia', 'Miller', 'student', 'mia.m@example.com', '2003-01-01  
11:00:00', '2022-01-11 14:00:00', '2022-01-11 15:45:00'),  
  
    ('Ethan', 'Clark', 'student', 'ethan.c@example.com', '1997-01-01  
13:15:00', '2022-01-12 08:30:00', '2022-01-12 12:15:00'),  
  
    ('Ava', 'Davis', 'student', 'ava.d@example.com', '1994-01-01  
09:00:00', '2022-01-13 10:45:00', '2022-01-13 14:30:00'),  
  
    ('Oliver', 'Garcia', 'student', 'oliver.g@example.com', '1997-01-01  
14:30:00', '2022-01-14 12:00:00', '2022-01-14 16:15:00'),
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

('Amelia', 'Hill', 'student', 'amelia.h@example.com', '1996-01-01 10:15:00', '2022-01-15 15:45:00', '2022-01-15 17:30:00'),

('Lucas', 'Lee', 'student', 'lucas.l@example.com', '1997-01-01 08:30:00', '2022-01-16 13:00:00', '2022-01-16 14:45:00'),

('Emma', 'Smith', 'student', 'emma.smith@example.com', '2004-01-01 00:00:00', '2022-01-22 08:00:00', '2022-01-22 12:00:00'),

('James', 'Johnson', 'student', 'james.johnson@example.com', '1997-01-01 00:00:00', '2022-01-20 08:00:00', '2022-01-20 12:00:00'),

('Sophia', 'Brown', 'student', 'sophia.brown@example.com', '1995-01-01 00:00:00', '2022-01-26 08:00:00', '2022-01-26 12:00:00'),

('Oliver', 'Davis', 'student', 'oliver.davis@example.com', '1994-01-01 00:00:00', '2022-01-21 08:00:00', '2022-01-21 12:00:00'),

('Ava', 'Miller', 'student', 'ava.miller@example.com', '1997-01-01 00:00:00', '2022-01-27 08:00:00', '2022-01-27 12:00:00'),

('Liam', 'Wilson', 'student', 'liam.wilson@example.com', '1996-01-01 00:00:00', '2022-01-25 08:00:00', '2022-01-25 12:00:00'),

('Isabella', 'Moore', 'student', 'isabella.moore@example.com', '1994-01-01 00:00:00', '2022-01-23 08:00:00', '2022-01-23 12:00:00'),

('Mia', 'Taylor', 'student', 'mia.taylor@example.com', '1997-01-01 00:00:00', '2022-01-24 08:00:00', '2022-01-24 12:00:00'),

('Ella', 'Nelson', 'instructor', 'ella.n@example.com', '1998-07-22 12:45:00', '2022-01-17 09:30:00', '2022-01-17 11:15:00'),



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

```
('Jackson', 'Owens', 'instructor', 'jackson.o@example.com',  
  '1989-09-01 09:30:00', '2022-01-18 10:45:00', '2022-01-18  
  12:30:00'),  
  
('Avery', 'Evans', 'instructor', 'avery.e@example.com', '1990-04-14  
  11:45:00', '2022-01-19 14:15:00', '2022-01-19 16:00:00');
```

-- Inserting data into Instructor table with computer science-related specializations

```
INSERT INTO Instructor (specialization, inst_id)  
SELECT 'Software Development', user_id  
FROM User  
WHERE account_type = 'instructor' AND first_name = 'Ella';  
  
INSERT INTO Instructor (specialization, inst_id)  
SELECT 'Data Structures and Algorithms', user_id  
FROM User  
WHERE account_type = 'instructor' AND first_name = 'Jackson';
```

```
INSERT INTO Instructor (specialization, inst_id)  
SELECT 'Web Development', user_id  
FROM User  
WHERE account_type = 'instructor' AND first_name = 'Avery';
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

-- Inserting data into Student table

INSERT INTO Student (std\_id)

VALUES

(2),  
(3),  
(4),  
(5),  
(6),  
(7),  
(8),  
(9),  
(10),  
(11),  
(12),  
(13),  
(14),  
(15),  
(16),  
(17),  
(18),  
(19),  
(20);



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

-- Inserting data into Admin table

```
INSERT INTO Admin (admin_id)
```

```
VALUES
```

```
(1);
```

-- Inserting data into Course table for Computer Science students

```
INSERT INTO Course (title, description, max_participants, inst_id)
```

```
VALUES
```

```
('Introduction to Programming', 'Embark on a journey into the world  
of programming with our Introduction to Programming course.
```

This beginner-friendly program is crafted for those stepping into the realm of coding for the first time. Covering essential concepts such as variables, control structures, and functions, this course provides a solid foundation in programming logic.

Delve into hands-on exercises and projects designed to reinforce your understanding of key programming principles.

Whether you are looking to kickstart a career in software development or gain a valuable skill set, this course is the gateway to unlocking the fascinating universe of coding.', 10, 21),

```
('Data Structures and Algorithm', 'Dive into the core of computer  
science with our Data Structures and Algorithms course.
```

This advanced program is tailored for developers seeking to enhance their problem-solving skills and optimize code efficiency.

Explore a comprehensive range of data structures, from arrays to trees, and algorithms such as sorting and searching.





## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

Learn how to analyze algorithmic complexities and make informed design choices.

Through practical exercises and real-world applications, you will gain proficiency in creating scalable and efficient solutions.

Elevate your programming prowess and be well-equipped to tackle complex challenges in software development with this in-depth exploration of data structures and algorithms.', 15, 22),

('Java', 'Explore the fundamentals of Java programming with our comprehensive course.

From basic syntax to advanced concepts like multithreading and exception handling, this course is designed for beginners and aspiring developers.

Learn to design and implement Java applications, work with object-oriented principles, and gain hands-on experience with real-world projects.

Whether youre aiming to kickstart your programming journey or enhance your skills, this Java course offers a solid foundation for building robust and scalable applications.', 8, 21),

('Web Development', 'Embark on a journey into the dynamic world of Web Development with our comprehensive course.

This program is crafted for both beginners and intermediate learners, providing a solid foundation in front-end and back-end technologies.

Dive into HTML, CSS, and JavaScript to build responsive and interactive user interfaces.

Explore server-side scripting with frameworks like Node.js and Express, coupled with database management using technologies like MongoDB.



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

Delve into the intricacies of RESTful API design and asynchronous programming.

Through hands-on projects, you will develop the skills needed to create modern, scalable web applications.

Whether you aspire to be a full-stack developer or build your web presence, this course equips you with the tools for success in the ever-evolving field of Web Development.', 9, 23);

--- Inserting data into Enrollment table for Computer Science students

```
INSERT INTO Enrollment (is_paid, std_id, course_id)
```

```
VALUES
```

```
(true, 2, 1),
```

```
(true, 3, 1),
```

```
(false, 4, 1),
```

```
(true, 5, 1),
```

```
(false, 6, 2),
```

```
(false, 7, 2),
```

```
(false, 8, 2),
```

```
(true, 9, 2),
```

```
(true, 10, 3),
```

```
(false, 11, 3),
```

```
(true, 12, 3),
```

```
(false, 13, 3),
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

```
(true,14, 3),
(true,15, 3),
(false,16, 4),
(true,17, 4),
(false,18, 4),
(true,19, 4),
(true,20, 4);

-- Inserting data into Discount table
INSERT INTO Discount (discount_name, discount_rate, start_date,
end_date)
VALUES
    ('Student Discount', 0.10, '2022-01-15 00:00:00', '2022-02-15
00:00:00'),
    ('Holiday Discount', 0.25, '2022-12-01 00:00:00', '2022-12-31
00:00:00'),
    ('Summer Sale', 0.30, '2022-06-01 00:00:00', '2022-08-31 00:00:00');

-- Inserting data into Payment table
INSERT INTO Payment (payment_method, due_date, price, course_id,
discount_id)
VALUES
    ('Cash', '2022-07-10 00:00:00', 500, 1, 1),
    ('Bank Transfer', '2022-03-12 00:00:00', 600, 2, 3),
    ('Cash', '2022-04-15 00:00:00', 350, 3, 2),
    ('Bank Transfer', '2022-03-18 00:00:00', 550, 4, 1);
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

-- Inserting data into Certification table

```
INSERT INTO Certification (course_id, std_id, discription, ranking)
```

```
VALUES
```

```
(1, 2, 'Introduction to Programming', 4.8),  
(1, 3, 'Introduction to Programming', 4.9),  
(1, 4, 'Introduction to Programming', 4.7),  
(1, 5, 'Introduction to Programming', 4.5),  
(2, 6, 'Data Structures and Algorithm', 4.6),  
(2, 7, 'Data Structures and Algorithm', 4.8),  
(2, 8, 'Data Structures and Algorithm', 4.9),  
(2, 9, 'Data Structures and Algorithm', 4.7),  
(3, 10, 'Java', 4.5),  
(3, 11, 'Java', 4.6),  
(3, 12, 'Java', 4.8),  
(3, 13, 'Java', 4.9),  
(3, 14, 'Java', 4.7),  
(3, 15, 'Java', 4.5),  
(4, 16, 'Web Development', 4.6),  
(4, 17, 'Web Development', 5.0),  
(4, 18, 'Web Development', 4.1),  
(4, 19, 'Web Development', 4.5),  
(4, 20, 'Web Development', 4.9);
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

-- Inserting data into Assignment table

```
INSERT INTO Assignment (deadline, assig_info, upload_date, course_id)
VALUES
    ('2022-06-15 23:59:59', 'Assignment file1.txt', '2022-01-15
12:00:00', 1),
    ('2022-02-25 23:59:59', 'Assignment file2.txt', '2022-02-01
10:00:00', 2),
    ('2022-03-20 23:59:59', 'Assignment file3.txt', '2022-02-15
14:30:00', 3),
    ('2022-02-23 23:59:59', 'Assignment file4.txt', '2022-03-01
11:45:00', 4),
    ('2022-03-15 23:59:59', 'Assignment file5.txt', '2022-04-01
09:15:00', 4),
    ('2022-06-05 23:59:59', 'Assignment file6.txt', '2022-05-01
13:30:00', 3),
    ('2022-03-10 23:59:59', 'Assignment file7.txt', '2022-06-01
15:45:00', 2),
    ('2022-07-05 23:59:59', 'Assignment file8.txt', '2022-07-01
12:00:00', 1);
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

-- Inserting data into Submission table

```
INSERT INTO Submission (submission_date, submission_file, std_id,  
    assig_id)
```

VALUES

```
('2022-06-12 23:55:00', 'Submitted file1.txt', 5, 1),  
( '2022-02-23 23:50:00', 'Submitted file2.txt', 6, 2),  
( '2022-03-19 23:45:00', 'Submitted file3.txt', 7, 3),  
( '2022-02-21 23:40:00', 'Submitted file4.txt', 8, 4),  
( '2022-03-14 23:35:00', 'Submitted file5.txt', 9, 5),  
( '2022-06-01 23:30:00', 'Submitted file6.txt', 10, 6),  
( '2022-03-09 23:25:00', 'Submitted file7.txt', 11, 7),  
( '2022-07-03 23:20:00', 'Submitted file8.txt', 12, 8),  
( '2022-06-04 23:15:00', 'Submitted file9.txt', 13, 6),  
( '2022-02-24 23:10:00', 'Submitted file10.txt', 14, 2),  
( '2022-02-21 23:05:00', 'Submitted file11.txt', 6, 4),  
( '2022-03-13 23:00:00', 'Submitted file12.txt', 2, 5),  
( '2023-06-04 22:55:00', 'Submitted file13.txt', 8, 6),  
( '2023-07-04 22:50:00', 'Submitted file14.txt', 10, 8),  
( '2023-02-23 22:45:00', 'Submitted file15.txt', 20, 2);
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

-- Inserting data into Schedule table

```
INSERT INTO Schedule (start_date, end_date, nr_of_sessions,  
    session_duration, course_id)
```

VALUES

```
('2022-06-10', '2022-08-10', 10, '02:00:00', 1),  
( '2022-02-12', '2022-04-12', 12, '01:30:00', 2),  
( '2022-03-15', '2022-06-15', 8, '02:30:00', 3),  
( '2022-02-18', '2022-04-18', 15, '01:00:00', 4);
```

-- Inserting data into Content table

```
INSERT INTO Content (upload_date, update_date, content_description,  
    course_id)
```

VALUES

```
('2022-06-11 09:00:00', '2022-07-08 15:30:00', 'Content file1.txt',  
1),  
( '2022-02-13 10:30:00', '2022-02-13 09:15:00', 'Content file1.txt',  
2),  
( '2022-03-18 08:20:00', '2022-03-18 17:45:00', 'Content file1.txt',  
3),  
( '2022-02-22 14:10:00', '2022-02-24 09:30:00', 'Content file1.txt',  
4),  
( '2022-05-25 16:45:00', '2022-05-25 15:45:00', 'Content file2.txt',  
3),  
( '2022-03-28 13:15:00', '2022-03-28 12:15:00', 'Content file2.txt',  
4),
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

```
('2022-03-04 09:00:00', '2022-03-04 10:45:00', 'Content file2.txt',  
2),  
( '2022-07-05 10:45:00', '2022-08-07 14:30:00', 'Content file2.txt',  
1),  
( '2022-08-08 08:30:00', '2022-08-08 10:45:00', 'Content file3.txt',  
1),  
( '2022-03-26 16:30:00', '2022-03-26 14:45:00', 'Content file3.txt',  
3),  
( '2022-03-25 12:00:00', '2022-03-25 16:15:00', 'Content file3.txt',  
2),  
( '2022-03-14 17:30:00', '2022-03-14 13:00:00', 'Content file3.txt',  
4);
```

#### ➤ SELECT

*1. Selektto emrat dhe adresat e emailit të të gjithë instruktorëve.*

```
SELECT CONCAT(first_name, " ", last_name) AS full_name, email FROM  
User  
  
JOIN Instructor ON user_id = inst_id;
```

*2. Selektto kurset me përshkrimet e tyre që kanë më shumë se 5 regjistrime.*

```
SELECT title, COUNT(title) AS enrollments_per_course FROM Course  
JOIN Enrollment ON Course.course_id = Enrollment.course_id  
JOIN Student ON Enrollment.std_id = Student.std_id  
GROUP BY title HAVING enrollments_per_course < 5;
```





## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

*3. Listo datat e dorëzimit dhe emrat e detyrave të paraqitura nga një student specifik.*

```
SELECT submission_date , assig_info
FROM assignment INNER JOIN submission
ON assignment.assig_id = submission.assig_id
WHERE std_id = 16;
```

*4. Selektot datat e fillimit dhe mbarimit të kurseve së bashku me numrin e seancave.*

```
SELECT title, start_date, end_date, nr_of_sessions
FROM schedule INNER JOIN course
ON schedule.course_id = course.course_id;
```

*5. Selektot emrat e studentëve që nuk janë regjistruar në asnjë kurs.*

```
SELECT first_name, last_name
FROM User
WHERE user_id IN (
    SELECT std_id
    FROM Student
    WHERE std_id NOT IN (
        SELECT std_id
        FROM Enrollment
    )
);
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

*6. Selektotitullin ekursit dhe të gjitha përshkrimet e përmbajtjes së tyre.*

```
SELECT Course.title, Content.content_description
FROM Course
JOIN Content ON Course.course_id = Content.course_id;
```

*7. Listoni emrat e plotë të studentëve, të cilët kanë një notë certifikimi më të lartë se 4.5.*

```
SELECT CONCAT(first_name, ' ', last_name) AS student_name FROM
    certification c
INNER JOIN user u
ON c.std_id = u.user_id AND c.ranking > 4.5;
SELECT CONCAT(u.first_name, ' ', u.last_name) AS student_name
FROM user u

WHERE u.user_id IN (
    SELECT std_id
    FROM certification c
    WHERE c.ranking > 4.5
);
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

*8. Selektro kursin, çmimin origjinal, emrin e zbritjes, perqindjen e zbritjes dhe çmimin përfundimtar pasi të jetë aplikuar zbritja  $\geq 25\%$*

```
SELECT
    C.title AS Title,
    P.price AS 'Original Price',
    D.discount_name AS 'Discount Name',
    D.discount_rate AS 'Discount Rate',
    P.price - (P.price * D.discount_rate) AS 'Final Price'
FROM
    Course C
JOIN
    Payment P ON C.course_id = P.course_id
JOIN
    Discount D ON P.discount_id = D.discount_id
WHERE
    D.discount_rate  $\geq$  0.25;
```

*9. Selektro përdoruesin që është log-uar i fundit.*

```
SELECT CONCAT(U.first_name, ' ', U.last_name) AS 'Last Logged In User'
FROM User U
ORDER BY last_loggedIn DESC
LIMIT 1;
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

*10. Gjeni regjistrimet e mbetura për kurs.*

```
SELECT
    title,
    max_participants - COUNT(Enrollment.std_id) AS remaining_enrolls
FROM
    Course
LEFT JOIN
    Enrollment ON Course.course_id = Enrollment.course_id
GROUP BY
    title, max_participants
HAVING
    max_participants - COUNT(Enrollment.std_id) > 4;
```

**➤ UPDATE**

*1. Zvogëlo kohëzgjatjen e sesionit të të gjitha kurseve "Java" me 1 orë.*

```
UPDATE Schedule
JOIN Course ON Schedule.course_id = Course.course_id
SET session_duration = SUBTIME(session_duration, "01:00:00")
WHERE title = "Java";
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

***2.Ndrysho afatin e një detyre specifike.***

```
UPDATE assigment  
SET deadline = "2022-04-20 00:00:00"  
WHERE assig_id = 4;
```

***3.Përditëso përshkrimin e një kursi me një përshkrim të ri të kursit.***

```
UPDATE course  
SET description = "Welcome to our comprehensive Web Development  
Course,  
where innovation meets instruction to transform you into a proficient  
web developer.  
This course is meticulously designed for both beginners and  
intermediate learners,  
offering a dynamic and hands-on approach to mastering the intricacies  
of web development."  
WHERE course_id = 4;
```

***4.Ndrysho datën e pagesës për një kurs specifik.***

```
UPDATE Payment  
SET due_date = '2022-04-10'  
WHERE course_id = 3;
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

*5. Ndrysho çmimin e një kursi specifik.*

```
UPDATE Payment
SET price = 600
WHERE course_id = 4;
```

*6. Ndrysho datat e fillimit dhe mbarimit të një kursi .*

```
UPDATE schedule
SET start_date = '2022-06-08', end_date = '2022-08-08'
WHERE course_id = 1;
```

*7. Ndrysho përqindjen për një discount specifik.*

```
UPDATE discount
SET discount_rate = 0.40
WHERE discount_name = 'Summer Sale' AND discount_id = 3;
```

*8. Përditëso adresën e emailit të instruktorëve me specializim në "Strukturat e të dhënave dhe Algoritmet".*

```
UPDATE User
SET email = 'new_email@example.com'
WHERE user_id IN (
    SELECT I.inst_id
    FROM Instructor I
    WHERE I.specialization = 'Data Structures and Algorithms'
);
```



**UNIVERSITETI I TIRANËS**

**Fakulteti i Shkencave të Natyrës**

*9. Përditëso datën e përfundimit për një kurs që i përket një instruktori që ka një specializim në 'Zhvillimin e Softuerit' dhe ka më shumë se 5 studentë të regjistruar.*

```
UPDATE Schedule
SET end_date = '2022-07-15'
WHERE course_id = (
    SELECT course_id
    FROM Course
    WHERE inst_id = (
        SELECT inst_id
        FROM Instructor
        WHERE specialization = 'Software Development')
    AND course_id IN (
        SELECT course_id
        FROM Enrollment
        GROUP BY course_id
        HAVING COUNT(std_id) > 5
    )
);
```



## UNIVERSITETI I TIRANËS

### Fakulteti i Shkencave të Natyrës

**10. Përditëso tabelën e instruktorëve duke shtuar dy kolona të reja: *hourly\_paid* dhe *paga* duke u dhënë atyre disa vlera të paracaktuara.**

```
ALTER TABLE instructor
ADD COLUMN hourly_paid DOUBLE;
ALTER TABLE instructor
ADD COLUMN salary DOUBLE;
UPDATE instructor
SET hourly_paid = 12;
UPDATE instructor
SET salary = 24960;
```

**11. Rritja e pagesës në orë me 5% për instruktorin që ka dhënë më shumë kurse.**

```
CREATE TRIGGER b_hp_update
BEFORE UPDATE ON instructor
FOR EACH ROW
SET NEW.salary = (NEW.hourly_paid * 2080);
CREATE VIEW Inst_history AS
SELECT inst_id, COUNT(inst_id) AS courses_given FROM Course
GROUP BY inst_id;

UPDATE Instructor
JOIN Inst_history ON Instructor.inst_id = Inst_history.inst_id
SET hourly_paid = hourly_paid * 0.05
WHERE courses_given = (SELECT MAX(courses_given) FROM Inst_history );
```





**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

➤ **DELETE**

*1.Fshi një zbritje specifike*

```
DELETE FROM discount  
WHERE discount_id = 2
```

*2.Hiq një kurs nga baza e të dhënave së bashku me të dhënat e tij përkatëse.*

```
DELETE FROM course  
WHERE course.id = course.id AND course.id = 2;
```

*3.Fshi një instruktor specifik nga baza e të dhënave.*

```
DELETE FROM Instructor  
WHERE inst_id = 22;
```

*4.Hiq një student me një emër specifik.*

```
DELETE FROM User  
WHERE first_name = 'Jane' AND account_type = 'student';
```

*5.Fshi të gjitha zbritjet që kanë mbaruar.*

```
DELETE discount FROM Discount  
JOIN Payment ON Payment.discount_id = Discount.discount_id  
WHERE end_date < NOW();
```



**UNIVERSITETI I TIRANËS**  
**Fakulteti i Shkencave të Natyrës**

***6.Fshi një content specifik të përmbajtjes për një kurs.***

```
DELETE FROM content  
WHERE course_id = 3 AND content_id = 3;
```

***7.Hiq një përdoruesi nga baza e të dhënave.***

```
DELETE FROM user  
WHERE user_id = 2;
```

***8.Çregjistro studentët që nuk e kanë paguar kursin pas 14 ditësh nga fillimi i kursit.***

```
DELETE enrollment FROM Enrollment  
JOIN Course ON Enrollment.course_id = Course.course_id  
JOIN Schedule ON Course.course_id = Schedule.course_id  
WHERE is_paid = false AND DATEDIFF(NOW(), start_date) > 14;
```

***9.Fshi të gjitha detyrat e kurseve që kanë përfunduar.***

```
DELETE assignment FROM Assignment  
JOIN Course ON Assignment.course_id = Course.course_id  
JOIN Schedule ON Course.course_id = Schedule.course_id  
WHERE Schedule.end_date < NOW();
```