PART 1 : Basic structure of Tables

STUDENTS TABLE

student_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

student_first_name (Adı) VARCHAR

student_last_name (Soyadı) VARCHAR

student_gender (Cinsiyet) VARCHAR

student_birth_date (Doğum tarihi) DATE

student_address (Adres) VARCHAR

student_email (E-posta) VARCHAR

student_phone (Telefon numarası) VARCHAR

student_enrollment_date (Kayıt tarihi) DATE

department_id (Bölüm numarası) FOREIGN KEY INT REF DEPARTMENT TABLE

event_id (event numarası) FOREIGN KEY INT REF EVENTS TABLE

DEPARTMENTS TABLE

department_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

department_name VARCHAR

faculty_id FOREIGN KEY INT REF FACULTY TABLE

FACULTIES TABLE

faculty_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

faculty_name VARCHAR

COURSES TABLE

course_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

course_code VARCHAR

course_name VARCHAR

course_description VARCHAR

course_credit INT

course_hours VARCHAR

department_id FOREIGN KEY INT REF DEPARTMENT TABLE

sub_course_id FOREIGN KEY INT REF SUB_COURSES TABLE

STUDENT_COURSES TABLE

student_course_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

student_id FOREIGN KEY INT REF STUDENT TABLE

course_id FOREIGN KEY INT REF COURSE TABLE

semester VARCHAR

year INT

INSTRUCTORS TABLE

instructor_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

instructor_first_name VARCHAR

instructor_last_name VARCHAR

instructor_birth_date DATE

instructor_gender VARCHAR

instructor_email VARCHAR

instructor_phone VARCHAR

department_id FOREIGN KEY INT REF DEPARTMENT TABLE

INSTRUCTORS_COURSES TABLE

instructor_courses_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

course_id FOREIGN KEY INT REF COURSE TABLE

instructor_id FOREIGN KEY INT REF INSTRUCTOR TABLE

COURSE_SCHEDULE TABLE

course_schedule_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

course_id FOREIGN KEY INT REF COURSE TABLE

instructor_id FOREIGN KEY INT REF INSTRUCTOR TABLE

room VARCHAR

day_of_week VARCHAR

start_time DATE

end_time DATE

EVENTS TABLE

event_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

event_title VARCHAR

event_description VARCHAR

event_date DATE

event_location VARCHAR

GRADES TABLE

grade_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

student_id FOREIGN KEY INT REF STUDENT TABLE

course_id FOREIGN KEY INT REF COURSE TABLE

midterm_grade INT

final_grade INT

average_grade INT

Quiz INT

Homework INT

SUB_COURSES TABLE

id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

code VARCHAR

name VARCHAR

description VARCHAR

credit INT

hours VARCHAR

SUB_COURSE_INSTRUCTORS TABLE

instructor_sub_courses_id SERIAL(PL/pgSQL)=> INT PRIMARY KEY

sub_course_id FOREIGN KEY INT REF SUB_COURSE TABLE

instructor_id FOREIGN KEY INT REF INSTRUCTOR TABLE

PART 2: SQL Queries for creating tables

1- FACULTIES TABLE

```
CREATE TABLE faculties (
  id SERIAL PRIMARY KEY,
  name VARCHAR(50)
);
```

2- DEPARTMENTS TABLE

```
CREATE TABLE departments (
  id SERIAL PRIMARY KEY,
  name VARCHAR(50),
```

faculty_id INT

ADD FOREIGN KEY (faculty_id) REFERENCES faculties(id);

);

3- STUDENTS TABLE

CREATE TABLE students (

id SERIAL PRIMARY KEY,

first_name VARCHAR(50),

last_name VARCHAR(50),

gender CHAR(1),

birth_date DATE,

address VARCHAR(255),

email VARCHAR(100),

phone VARCHAR(20),

enrollment_date DATE,

department_id INT REFERENCES departments(id)

event_id INT REFERENCES events(id)

);

4- COURSES TABLE

CREATE TABLE courses (

id SERIAL PRIMARY KEY,

code VARCHAR(20),

name VARCHAR(50),

description TEXT,

credit INT,

```
hours INT,
department_id INT REFERENCES departments(id)
);

ALTER TABLE courses ADD sub_course_id INT;
ALTER TABLE courses ADD FOREIGN KEY (sub_course_id) REFERENCES sub_courses(id);
```

5- INSTRUCTORS TABLE

```
CREATE TABLE instructors (
id SERIAL PRIMARY KEY,
first_name VARCHAR(50),
last_name VARCHAR(50),
birth_date DATE,
gender CHAR(1),
email VARCHAR(100),
phone VARCHAR(20),
department_id INT REFERENCES departments(id)
);
```

6- COURSE SCHEDULE TABLE

```
CREATE TABLE course_schedule (
id SERIAL PRIMARY KEY,
course_id INT,
instructor_id INT,
room VARCHAR(20),
day_of_week INT,
start_time TIME,
```

end_time TIME,

FOREIGN KEY (course_id) REFERENCES courses(id),

FOREIGN KEY (instructor_id) REFERENCES instructors(id)

);

7- GRADES TABLE

CREATE TABLE grades (

id SERIAL PRIMARY KEY,

student_id INT,

course_id INT,

midterm_grade INT,

final_grade INT,

average_grade INT,

FOREIGN KEY (student_id) REFERENCES students(id),

FOREIGN KEY (course_id) REFERENCES courses(id)

);

ALTER TABLE grades ADD Quiz int

ALTER TABLE grades ADD Homework int

8- STUDENT_COURSES TABLE

CREATE TABLE student_courses (

id SERIAL PRIMARY KEY,

student_id INT,

course_id INT,

semester INT,

year INT,

FOREIGN KEY (student_id) REFERENCES students(id),

FOREIGN KEY (course_id) REFERENCES courses(id)

);

9- COURSE_INSTRUCTORS TABLE

CREATE TABLE course_instructors (

course_id INT,

instructor_id INT,

PRIMARY KEY (course_id, instructor_id),

FOREIGN KEY (course_id) REFERENCES courses(id),

FOREIGN KEY (instructor_id) REFERENCES instructors(id)

);

10- EVENTS TABLE

CREATE TABLE events (

id SERIAL PRIMARY KEY,

title VARCHAR(255),

description TEXT,

event_date DATE,

location VARCHAR(255));

11- SUB_COURSES TABLE

CREATE TABLE sub_courses(

id SERIAL PRIMARY KEY,

code VARCHAR(20),

name VARCHAR(20),

description TEXT,

credit INT,

hours INT

);

12- SUB_COURSE_INSTRUCTORS TABLE

CREATE TABLE sub_course_instructors (

sub_course_id INT,

instructor_id INT,

PRIMARY KEY (sub_course_id, instructor_id),

FOREIGN KEY (sub_course_id) REFERENCES sub_courses(id),

FOREIGN KEY (instructor_id) REFERENCES instructors(id)

);

PART 3: SQL Queries for inserting data

INSERT INTO students(first_name,last_name,gender,birth_date,address,email,phone,enrollment_date, department_id,event_id) VALUES

('jon','snow','m','2000-05-19','a city b street', 'kinginthenorth@gmail.com', '25637255','2022-12-12',1,1),

('arya','stark','f','2004-04-02','c city d street','nonamegirl@gmail.com','11225566', '2023-02-14',2,1),

('tanjiro','kamado','m','2000-10-19','e city f street','hinokamikagura@gmail.com','77785269','2021-01-10',1,1),

('jesse','pinkman','m','1990-03-22', 'g city h street', 'whatsup@gmail.com','44485694','2020-04-24',3,2),

('dean','winchester','m','1992-01-01','t city u street','familybusiness@gmail.com','78912332','2018-09-15',1,3)

INSERT INTO faculties(name) VALUES('engineering')

INSERT INTO departments(name,faculty_id) VALUES

('computer engineering',1),

('machine engineering',1),

('chemical engineering',1)

INSERT INTO events(title,description,event_date,location) VALUES

('meeting','meeting with student','2023-11-11','school'),

('cooking','cook something','2022-04-03','caravan'),

('hunting','kill the demons','2020-06-22','abandoned house')

INSERT INTO
instructors(first_name,last_name,birth_date,gender,email,phone,department_id) VALUES

('Ygritte','Nothing','1990-01-14','f','youknownothing@gmail.com','55517895',1),

('Joseph','Ledet','1980-03-15','m','bestteacherever@gmail.com','25677418',1),

('Syrio','Forel','1977-03-13','m','nottoday@gmail.com','47458965',2),

('No Face','Man','1988-05-24','m','valarmorghulis@gmail.com','78945622',2),

('Walter','White','1975-02-25','m','saymyname@gmail.com','99548222',3),

('Saul','Goodman','1985-08-27','m','bettercallsaul@gmail.com','36544712',3),

('Sakonji','Urokodaki','1960-02-01','m','masterurokodaki@gmail.com','78955522',1),

('John','Winchester','1978-07-17','m','givemecolt@gmail.com','45788221',1),

('Bobby','Singer','1975-12-11','m','ohboys@gmail.com','96587412',1)

INSERT INTO courses(code,name,description,credit,hours,department_id) VALUES

('CS200','programming','javascript programming',5,2,1),

('CS202','database','DB management',6,2,1),

('MS101','Intro to Machine','machine 101 course',4,2,2),

('MS102','Advanced Machine','machine 102 course',6,2,2),

('KM101','Chemical Materials','intro chemical materials',4,2,3),

('KM106','Chemical Experiments','intro chemical experiments',3,2,3),

('CS303','Monster Organization','Hunting wild monsters',5,2,1),

('CS304','Weapon Architecture','Gun care',5,2,1)

----------+

INSERT INTO course_instructors VALUES

(1,1),

(1,7),

(2,2),

(3,3),

(4,4),

(5,5),

(6,6),

(7,8),

(8,9)

INSERT INTO
course_schedule(course_id,instructor_id,room,day_of_week,start_time,end_time)
VALUES

(1,1,'d302',1,'08:00','10:00'),

(2,2,'d301',1,'08:00','10:00'),

(3,3,'m222',2,'09:00','11:00'),

(4,4,'m333',2,'09:00','11:00'),

(1,7,'c111',3,'10:00','12:00'),

(5,5,'g123',4,'13:00','15:00'),

(6,6,'g321',4,'13:00','15:00'),

(2,2,'d555',5,'08:00','10:00'),

(7,8,'d666',5,'08:00','10:00'),

(8,9,'d777',5,'08:00','10:00')

INSERT INTO grades(student_id,course_id,midterm_grade,final_grade,average_grade) VALUES

(1,1,40,80,60),

(1,2,30,70,50),

(2,3,60,100,80),

(2,4,40,100,70),

(3,1,10,90,50),

(4,5,50,100,75),

(4,6,100,100,100),

(5,2,20,60,40),

(5,7,100,60,80),

(5,8,40,80,60)

INSERT INTO student_courses(student_id,course_id,semester,year) VALUES

(1,1,1,2022),

(1,2,2,2022),

(2,3,1,2023),

(2,4,2,2023),

(3,1,1,2021),

(4,5,1,2020),

(4,6,2,2020),

(5,2,2,2018),

(5,7,2,2018),

(5,8,2,2018)

PART 4: SQL Query for view

```
CREATE VIEW unnormalized_data AS
SELECT
students.id as student_id,
students.first_name as student_first_name,
students.last_name as student_last_name,
students.gender as student_gender,
students.birth_date as student_birth_date,
students.address as student_address,
students.email as student_email,
students.phone as student_phone,
students.enrollment_date as student_enrollment_date,
departments.id as department_id,
departments.name as department_name,
faculties.id as faculty_id,
faculties.name as faculty_name,
courses.id as course_id,
courses.code as course_code,
courses.name as course_name,
courses.description as course_description,
courses.credit as course_credit,
courses.hours as course_hours,
instructors.id as instructor_id,
instructors.first_name as instructor_first_name,
instructors.last_name as instructor_last_name,
instructors.birth_date as instructor_birth_date,
instructors.gender as instructor_gender,
instructors.email as instructor_email,
instructors.phone as instructor_phone,
course_schedule.id as course_schedule_id,
```

course_schedule.room as course_schedule_room,

course_schedule.day_of_week as course_schedule_day_of_week,

course_schedule.start_time as course_schedule_start_time,

course_schedule.end_time as course_schedule_end_time,

grades.id as grade_id,

grades.midterm_grade as grade_midterm_grade,

grades.final_grade as grade_final_grade,

grades.average_grade as grade_average_grade,

events.id as event_id,

events.title as event_title,

events.description as event_description,

events.event_date as event_date,

events.location as event_location

FROM students

JOIN departments ON students.department_id = departments.id

JOIN faculties ON departments.faculty_id = faculties.id

JOIN student_courses ON students.id = student_courses.student_id

JOIN courses ON student_courses.course_id = courses.id

JOIN course_instructors ON courses.id = course_instructors.course_id

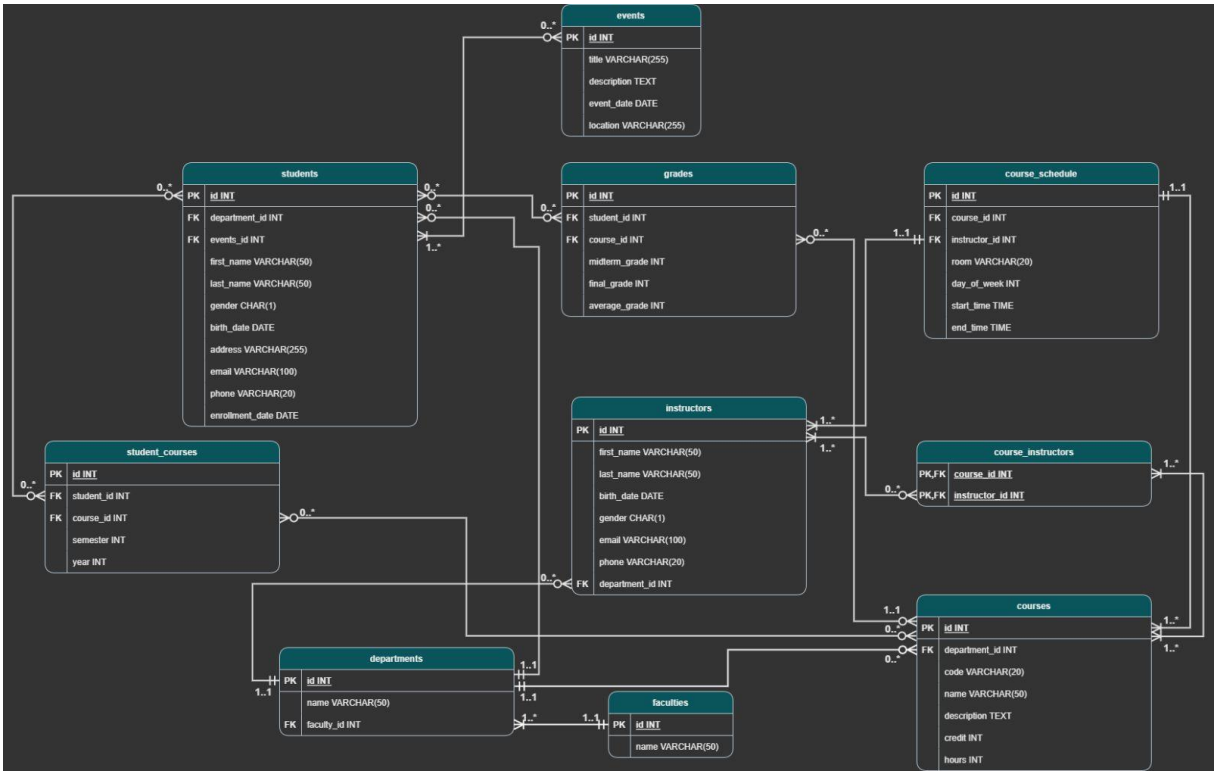
JOIN instructors ON course_instructors.instructor_id = instructors.id

JOIN course_schedule ON courses.id = course_schedule.course_id AND instructors.id = course_schedule.instructor_id

JOIN events ON students.event_id = events.id

JOIN grades ON students.id = grades.student_id AND courses.id = grades.course_id;

PART 5: E-R Diagram Image



PART 6: GROUP MEMBERS

MUSTAFA EMRE OZTURK => 20180808030

ANIL BERKAN ÜZREK => 20200808027

SEVGİ SARICA => 20190808066

BERAT BERKE DEMİR => 20220808610

MUSTAFA MERT CAN => 20220808705