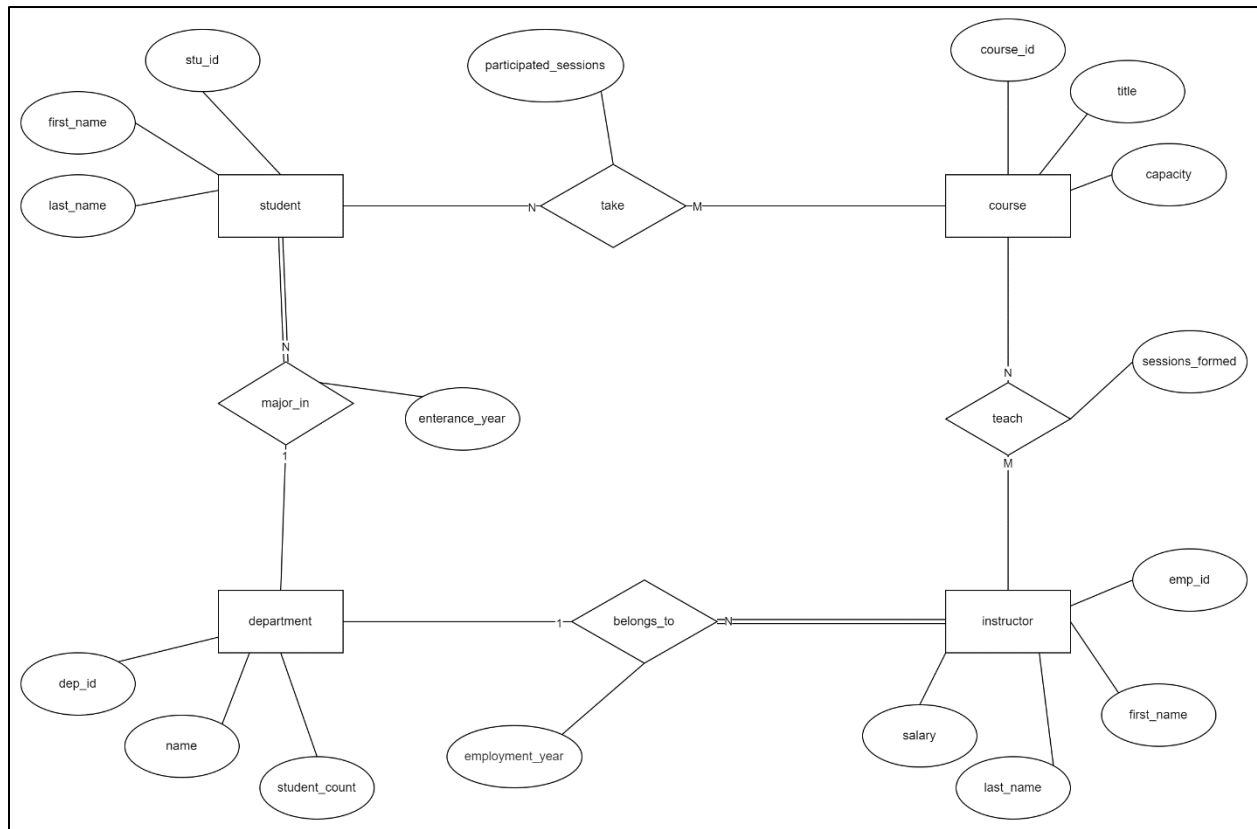


محیط عملیاتی انتخاب شده دانشگاه با چهار موجودیت دانشجو، استاد، گروه و درس می باشد که هر کدام دارای تعدادی صفت هستند. نمودار E-R به شرح زیر می باشد. همانطور که ملاحظه می شود ارتباطات از جنس N به M و 1 به N هستند.



جداول تعریف شده برای نمودار بالا به شرح زیر می باشد. کلید اصلی موجودیت های سمت ۱ باید در جدول سمت N قرار بگیرند. برای ارتباطات N به M هم جدول جدید تعریف شده است:

student

stu_id: varchar(10)	first_name: varchar(40)	last_name: varchar(40)	dep_id: varchar(5)	entrance_year: smallint

department

dep_id: varchar(5)	name: varchar(30)	student_count: smallint

instructor

emp_id: varchar(10)	first_name: varchar(40)	last_name: varchar(40)	salary: int	dep_id: varchar(5)	employment_year: smallint

course

course_id: varchar(5)	title: varchar(50)	capacity: smallint

take

stu_id: varchar(10)	course_id: varchar(5)	participated_sessions: smallint

teach

emp_id: varchar(10)	course_id: varchar(5)	sessions_formed: smallint

در مرحله ی بعد باید وارد ترمینال postgresql شد که با دستور

```
sudo -i -u postgres
```

قابل اجراست. سپس پایگاه داده ی جدیدی به شکل زیر ساخته می شود.

```
postgres=# create database exercise_1;
CREATE DATABASE
postgres=# \l
```

با دستور \l می توان پایگاه داده های ایجاد شده را بررسی کرد.

List of databases					
Name	Owner	Encoding	Collate	Ctype	Access privileges
ch4_bookstore	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	
exercise_1	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	
misagh	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	
postgres	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	
session_2	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	
template0	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres
+					
postgres					postgres=CTc/
template1	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres
+					
postgres					postgres=CTc/
(7 rows)					
(END)					

با دستور \c می توان به یک پایگاه داده متصل شد یا اتصال به پایگاه داده و کاربر وارد شده را بررسی کرد.

```
postgres=# \c exercise_1;
You are now connected to database "exercise_1" as user "postgres".
exercise_1=# \c
You are now connected to database "exercise_1" as user "postgres".
```

در ادامه جدول ها ساخته می شوند. با دستور \d می توان جداول ساخته شده را مشاهده کرد.

```
exercise_1=# create table department(
dep_id varchar(5),
name varchar(30) not null,
student_count smallint check(student_count >= 0),
primary key (dep_id)
);
CREATE TABLE
exercise_1=# \d
List of relations
Schema | Name | Type | Owner
-----+-----+-----+-----
public | department | table | postgres
(1 row)

exercise_1=# create table course(
course_id varchar(5),
title varchar(50) not null,
capacity smallint check(capacity >= 0),
primary key (course_id)
);
CREATE TABLE
exercise_1=#
```

```

exercise_1=# create table instructor(
emp_id varchar(10),
first_name varchar(40) not null,
last_name varchar(40) not null,
dep_id varchar(5) not null,
salary float check(salary >= 0),
employment_year smallint check(employment_year >= 1350),
primary key (emp_id),
constraint dep_fk foreign key(dep_id) references department(dep_id) on delete CASCADE
);
CREATE TABLE
exercise_1=# create table student(
stu_id varchar(10),
first_name varchar(40) not null,
last_name varchar(40) not null,
dep_id varchar(5) not null,
enterance_year smallint check(enterance_year >= 1350),
primary key (stu_id),
constraint dep_fk foreign key(dep_id) references department(dep_id) on delete CASCADE
);
CREATE TABLE
exercise_1=# create table take(
course_id varchar(5),
stu_id varchar(10),
participated_sessions smallint,
constraint course_fk foreign key(course_id) references course(course_id)
on delete CASCADE,
constraint stu_fk foreign key(stu_id) references student(stu_id)
on delete CASCADE);
CREATE TABLE
exercise_1=# create table teach(
emp_id varchar(10),
course_id varchar(5),
sessions_formed smallint,
constraint course_fk foreign key(course_id) references course(course_id)
on delete CASCADE,
constraint emp_fk foreign key(emp_id) references instructor(emp_id)
on delete CASCADE);
CREATE TABLE

```

در ادامه به جداول عمل insert به صورت تکی و گروهی انجام می شود:

```

exercise_1=# insert into course(course_id, title, capacity) values
('09231', 'Computer Networks', 45),
('41982', 'Advanced Programming', 90),
('87013', 'Database Lab', 15),
('87122', 'Electronics 2', 75);
INSERT 0 4
exercise_1=# insert into student(stu_id, first_name, last_name, enterance_year, dep_id) values
('9801226800', 'Seyyed Amirhosein', 'Misaghi', 1398, '12345');
INSERT 0 1
exercise_1=# insert into student(stu_id, first_name, last_name, enterance_year, dep_id) values
('9803416680', 'Mohammad Hosein', 'Hosein Zade', 1398, '11011'),
('9904138732', 'Zahra', 'Dehnavi', 1399, '54321'),
('9741349879', 'Amir', 'Karimi', 1397, '12345');
INSERT 0 3
exercise_1=# insert into department(dep_id, name, student_count) values
('12345', 'Computer Engineering', 1000),
('54321', 'Computer Science', 1500),
('11011', 'Electrical Engineering', 2000);
INSERT 0 3

```

```

exercise_1=# insert into instructor(emp_id, first_name, last_name, salary, dep_id, employmen
t_year) values
('5088989384', 'Mahdi', 'Aminnian', 30000000, '12345', 1388),
('9870871342', 'Bahman', 'Zanj', 29000000, '11011', 1379),
('0001388484', 'Arash', 'Yousefi', 15000000, '12345', 1395),
('0138781191', 'Kamran', 'Mallai', 42000000, '54321', 1400);
INSERT 0 4

exercise_1=# insert into take(stu_id, course_id, participated_sessions) values
('9801226800', '09231', 10),
('9904138732', '41982', 9),
('9801226800', '87013', 3),
('9803416680', '87122', 5);
INSERT 0 4

exercise_1=# insert into teach(emp_id, course_id, sessions_formed) values
('0001388484', '87013', 5),
('9870871342', '87122', 13),
('5088989384', '09231', 14);
INSERT 0 3

```

بررسی نقض شروط تعريف شده بصورت فوق می باشد:

```

exercise_1=# insert into student values ('9801226800', 'Seyyed Amirreza', 'Misaghi', '11011'
, 1398);
ERROR: duplicate key value violates unique constraint "student_pkey"
DETAIL: Key (stu_id)=(9801226800) already exists.

exercise_1=# insert into student values ('9801226100', 'Seyyed Amirreza', 'Misaghi', '11111'
, 1398);
ERROR: insert or update on table "student" violates foreign key constraint "dep_fk"
DETAIL: Key (dep_id)=(11111) is not present in table "department".

exercise_1=# insert into instructor values('1111111111', 'foo', 'bar', '12345', -1.2, 1399);
ERROR: new row for relation "instructor" violates check constraint "instructor_salary_check
"
DETAIL: Failing row contains (1111111111, foo, bar, 12345, -1.2, 1399).

exercise_1=# insert into instructor values('1111111111', 'foo', 'bar', '12345', 1.2, 1209);
ERROR: new row for relation "instructor" violates check constraint "instructor_employment_y
ear_check"
DETAIL: Failing row contains (1111111111, foo, bar, 12345, 1.2, 1209).

```

در ادامه جست و جو های ساده، سپس پیشرفته تری صورت گرفته است:

```

exercise_1=# select emp_id from instructor;
   emp_id
-----
 5088989384
 9870871342
 0001388484
 0138781191
(4 rows)

exercise_1=# select first_name, last_name
exercise_1=# from student
exercise_1=# where dep_id='12345';
   first_name   | last_name
-----+-----
 Seyyed Amirhosein | Misaghi
      Amir        | Karimi
(2 rows)

```

```
exercise_1=# select first_name as esm, last_name as famili
exercise_1-# from instructor
exercise_1-# where salary >= 20000000;
```

```
   esm   | famili
-----+-----
Mahdi   | Aminnian
Bahman  | Zanj
Kamran  | Mallai
(3 rows)
```

```
exercise_1=# select * from student
where entrance_year >= 1398
order by last_name desc;
```

```
   stu_id   | first_name   | last_name | dep_id | entrance_year
-----+-----+-----+-----+-----
9801226800 | Seyyed Amirhosein | Misaghi   | 12345  | 1398
9803416680 | Mohammad Hosein   | Hosein Zade | 11011  | 1398
9904138732 | Zahra            | Dehnavi   | 54321  | 1399
(3 rows)
```

```
exercise_1=# select first_name, last_name from
exercise_1-# student
exercise_1-# inner join take
exercise_1-# on student.stu_id=take.stu_id;
```

```
   first_name   | last_name
-----+-----
Seyyed Amirhosein | Misaghi
Zahra           | Dehnavi
Seyyed Amirhosein | Misaghi
Mohammad Hosein   | Hosein Zade
(4 rows)
```

```
exercise_1=# select distinct first_name, last_name from
instructor
inner join teach
on instructor.emp_id=teach.emp_id;
```

```
   first_name | last_name
-----+-----
Mahdi        | Aminnian
Arash        | Yousefi
Bahman       | Zanj
(3 rows)
```

استفاده از توابع رشته ای موجود در postgresql که شامل length, split_part, reverse, upper, substring می باشند، در ادامه آورده شده است:

```
exercise_1=# select * from student
where length(last_name) >=7;
```

```
   stu_id   | first_name   | last_name | dep_id | entrance_year
-----+-----+-----+-----+-----
9801226800 | Seyyed Amirhosein | Misaghi   | 12345  | 1398
9803416680 | Mohammad Hosein   | Hosein Zade | 11011  | 1398
9904138732 | Zahra            | Dehnavi   | 54321  | 1399
(3 rows)
```

```
exercise_1=# select upper(first_name)
exercise_1-# from student
exercise_1-# where last_name='Misaghi';
      upper
```

```
-----
SEYYED AMIRHOSEIN
(1 row)
```

```

exercise_1=# select split_part(first_name, 'a', 1)
exercise_1-# from instructor;
split_part
-----
M
B
Ar
K
(4 rows)

```

```

exercise_1=# select reverse(title) as reversed_susy_title
exercise_1-# from course;
reversed_susy_title
-----
skrowteN retupmoC
gnimmargorP decnavdA
baL esabataD
2 scinortceE
(4 rows)

```

```

exercise_1=# select substring(ins1.first_name, 2, 4) as substr, ins2.first_name
from instructor as ins1
inner join instructor as ins2
on ins1.emp_id=ins2.emp_id;
substr | first_name
-----+-----
ahdi   | Mahdi
ahma   | Bahman
rash   | Arash
amra   | Kamran
(4 rows)

```

```

exercise_1=# select title from
(select course_id from
(select stu_id from student where stu_id like '98%') as tb1
inner join take
on take.stu_id=tb1.stu_id) as tb2
inner join course
on tb2.course_id=course.course_id;
title
-----
Computer Networks
Database Lab
Electronics 2
(3 rows)

```

می توان عمل update جدول را به صورت یک delete + insert تعبیر کرد. عضو اول جدول برداشته شده، تغییر کرده و دوباره در جدول درج می شود. جدول قبل و بعد از دستور update در دو مثال قابل مشاهده است.

```
exercise_1=# select * from student;
```

stu_id	first_name	last_name	dep_id	enterance_year
9801226800	Seyyed Amirhosein	Misaghi	12345	1398
9803416680	Mohammad Hosein	Hosein Zade	11011	1398
9904138732	Zahra	Dehnavi	54321	1399
9741349879	Amir	Karimi	12345	1397

(4 rows)

```
exercise_1=# update student
set first_name='Lale'
where stu_id='9904138732';
UPDATE 1
exercise_1=# select * from student;
```

stu_id	first_name	last_name	dep_id	enterance_year
9801226800	Seyyed Amirhosein	Misaghi	12345	1398
9803416680	Mohammad Hosein	Hosein Zade	11011	1398
9741349879	Amir	Karimi	12345	1397
9904138732	Lale	Dehnavi	54321	1399

(4 rows)

```
exercise_1=# select * from instructor;
```

emp_id	first_name	last_name	dep_id	salary	employment_year
5088989384	Mahdi	Aminnian	12345	30000000	1388
9870871342	Bahman	Zanj	11011	29000000	1379
0001388484	Arash	Yousefi	12345	15000000	1395
0138781191	Kamran	Mallai	54321	42000000	1400

(4 rows)

```
exercise_1=# update instructor
exercise_1=# set salary=31000000
exercise_1=# where dep_id='12345';
UPDATE 2
exercise_1=# select * from instructor;
```

emp_id	first_name	last_name	dep_id	salary	employment_year
9870871342	Bahman	Zanj	11011	29000000	1379
0138781191	Kamran	Mallai	54321	42000000	1400
5088989384	Mahdi	Aminnian	12345	31000000	1388
0001388484	Arash	Yousefi	12345	31000000	1395

(4 rows)

درباره ی عمل delete نکته ی حائز اهمیت این است که اگر تاپل های جداول اصلی قرار است حذف شوند، باید در هنگام تعریف جدول قید on delete CASCADE برایشان تعریف شود. این قید بدین گونه عمل می کند که بعد از درخواست حذف یک تاپل که صفت کلید آن به عنوان کلید خارجی در سایر جداول استفاده شد، تاپل های موجود در آن جداول با این کلید خارجی را هم حذف می کند؛ اما حذف کردن از جداول روابط بدون مانع می باشد.

```
exercise_1=# select * from instructor;
  emp_id | first_name | last_name | dep_id | salary | employment_year
-----+-----+-----+-----+-----+-----
  9870871342 | Bahman    | Zanj      | 11011  | 29000000 | 1379
  0138781191 | Kamran    | Mallai    | 54321  | 42000000 | 1400
  5088989384 | Mahdi     | Aminnian  | 12345  | 31000000 | 1388
  0001388484 | Arash     | Yousefi   | 12345  | 31000000 | 1395
(4 rows)

exercise_1=# delete from instructor
exercise_1=# where first_name='Bahman' and last_name='Zanj';
DELETE 1
exercise_1=# select * from instructor;
  emp_id | first_name | last_name | dep_id | salary | employment_year
-----+-----+-----+-----+-----+-----
  0138781191 | Kamran    | Mallai    | 54321  | 42000000 | 1400
  5088989384 | Mahdi     | Aminnian  | 12345  | 31000000 | 1388
  0001388484 | Arash     | Yousefi   | 12345  | 31000000 | 1395
(3 rows)

exercise_1=# select teach from instructor;
ERROR: column "teach" does not exist
LINE 1: select teach from instructor;
           ^
exercise_1=# select * from teach;
  emp_id | course_id | sessions_formed
-----+-----+-----
  0001388484 | 87013    | 5
  5088989384 | 09231    | 14
(2 rows)
```

```

exercise_1=# select * from student;
 stu_id | first_name | last_name | dep_id | entrance_year
-----+-----+-----+-----+-----
 9801226800 | Seyyed Amirhosein | Misaghi | 12345 | 1398
 9803416680 | Mohammad Hosein | Hosein Zade | 11011 | 1398
 9741349879 | Amir | Karimi | 12345 | 1397
 9904138732 | Lale | Dehnavi | 54321 | 1399
(4 rows)

exercise_1=# delete from student
exercise_1=# where stu_id='9801226800';
DELETE 1
exercise_1=# select * from student;
 stu_id | first_name | last_name | dep_id | entrance_year
-----+-----+-----+-----+-----
 9803416680 | Mohammad Hosein | Hosein Zade | 11011 | 1398
 9741349879 | Amir | Karimi | 12345 | 1397
 9904138732 | Lale | Dehnavi | 54321 | 1399
(3 rows)

exercise_1=# select * from take;
 course_id | stu_id | participated_sessions
-----+-----+-----
 41982 | 9904138732 | 9
 87122 | 9803416680 | 5
(2 rows)

```

• Table Space ها چیستند؟

Table Space ها در PostgreSQL محل هایی هستند که شامل فایل های اطلاعات پایگاه داده هستند. این فایل های اطلاعات فضایی هستند که اشیائی مثل جداول، دیدها و ... در آنها ذخیره می شوند. در PostgreSQL هر کدام از این اشیاء فایل اطلاعات خودشان را دارند. PostgreSQL از فایل سیستم سیستم عامل به عنوان فضای ذخیره سازی استفاده می کند.