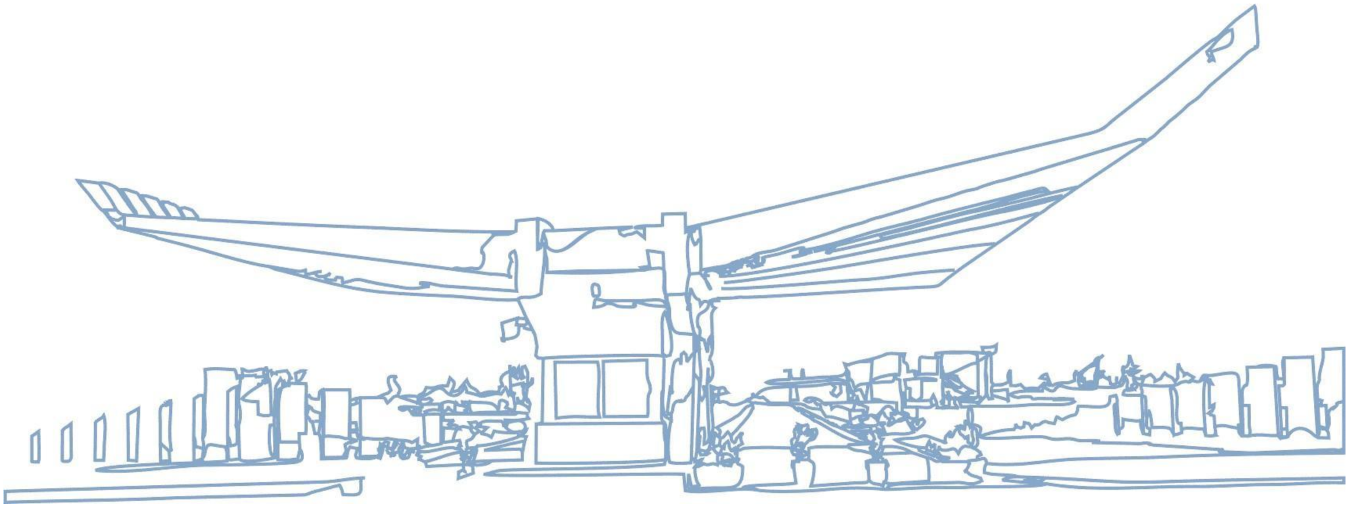
Software Engineering II   2022 -2023

Database Management System

Epoka University Clubs



Team Members:

David Keci

Era Mulla

Erisa Zaimi

Greisi Jaho

Klea Haxhiu

Table of Contents

1. Introduction.

This project is about creating a database about Epoka'a Clubs, which have been a part of a lot of generation's educational life. The main requirement of this project was creating a well-functional database that can be used by our university in the future and fulfill the needed entities and attributes.

Completing this project was a challenging, tiring, collaborative, and interesting journey, which has helped us a lot in different aspects. We learned how to understand the requirements of the client, regardless of the fact that it was the first project in this field. We found a way of communicating every day about this project, not more than ten minutes a day, and confirmed Wednesday as our main meeting day, which lasted two to three hours. The hardest part of this project was modeling the ERD diagram since it was creating the main idea of the club's database. We changed the diagram more than four times to achieve the right one, after researching a lot of information, which was fittable for our project. After finishing the first step, we continued with other steps of the project, until we created a whole SQL functional code.

This project was very helpful because it developed our knowledge, created a strong professional and friendly relationship with each other, managed a not known situation, such as having a real client and communicating with her, and of course, completed each task and requirement from the client.

1.1 Project Overview

The project includes

1. Creating an ER diagram.
2. Creating an RS Schema.
3. Creating SQL files and queries.
4. CREATING & INSERTING DATA.
5. Creating more than 20 Managerial Queries.

2. Planning

2.1. The steps We followed in organizing the project

1. Firstly, we asked questions to the client, who in our case was the university's president to understand her needs.
2. After gathering enough information we organized and analyzed it to help us have a more clear idea about what should be done.
3. Then we designed an ER diagram, which would be very helpful for the other steps to come. We created entities and each of their attributes.
4. Furthermore in the project we created an RS Schema and relationships between each of the tables, using primary and foreign keys.
5. After that we wrote the SQL code and created tables, files, and queries.

6. Another step of the project was to create and insert data into each of the corresponding tables.
7. The last step was to create managerial queries, where we created plenty, using statements and conditions that we have learned such as SELECT, WHERE, DISTINCT, ORDER BY, GROUP BY, HAVING, IN, JOIN, ALIAS, ALTER TABLE, UPDATE, DELETE, etc. We also used Aggregate Functions and Nested Queries

2.2. Description of the upcoming phases

This documentation will provide the needed information and content of all the steps that were mentioned before, also with photos of (ERD,RS,SQL,QUERIES), to give an overall idea of the whole database.This documentation gives answers to questions such as normalization and more.

3. Requirement

3.1. Requirement 1(Productivity Requirement)

When you create a database, you need to know the features of the storage that you want to use for it. Therefore, you must calculate the IOPS and the throughput of your database server.

IOPS(Input/Output Operations Per Second**)** are the number of reads and writes in seconds.

Throughput: Indicates how many MB reads or writes can be done per second. Throughput is calculated using IOPS and BlockSize (Allocation Unit Size) parameters like the below.

Formula:

Throughput=IOPSxBlockSize/1024

*Manufacturers generally assume BlockSize to be 4K when calculating storage throughput. But the recommended SQL Server is 64K. Because SQL Server usually reads and writes 64K.

3.2. Requirement 1 (Error Tolerance Requirement)

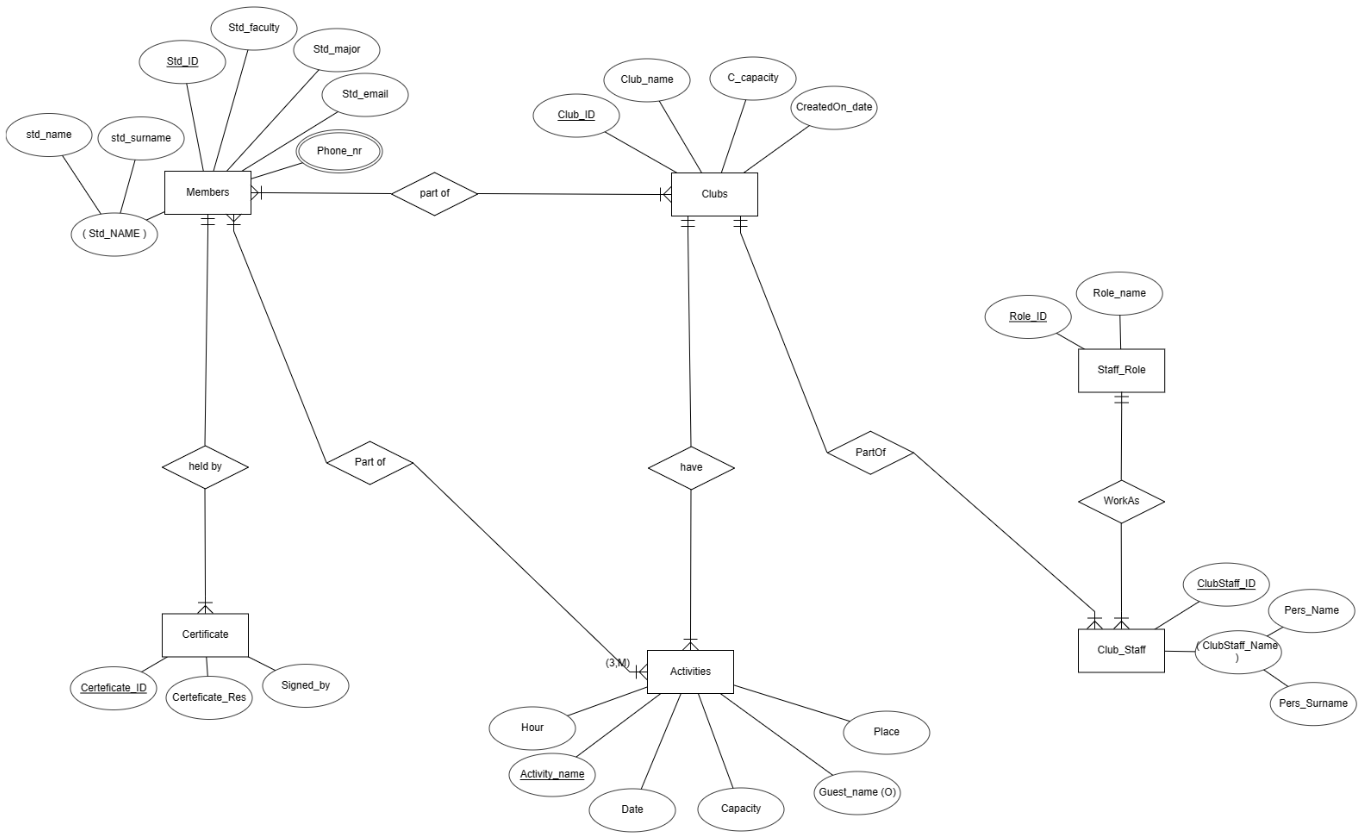
The SQL must be with as few errors as it can, so we can define it as a well-created SQL. Our database error tolerance will be 0.004% and 0.013%. The expected values should be 40.1.

*Normalization of our database:

Our database has many lines of code, and a lot of information and data to be registered, which in some cases they might be the same ones. In this case, we need normalization to avoid the deletion of important data in “Delete queries”, and the update of the wrong data in “Update queries”. Normalization is the reason to avoid these problematic situations.

4. ER Diagram

4.1. Explanation of the ER Diagram



- For each member of the club, we will keep track of the unique student id, student's name, composed by student name and student surname, student faculty, student major, student email, and a multivalued phone number.

- For each club we will keep track of the club's unique id, club name, club capacity, and the date that the club was created.

- For each person working in managing the club considered as staff, we will keep track of the staff's unique id, staff name composed by the name of the person, and surname of the person.

- For each role that a staff member has we will keep track of the role's unique id and role name.

- For each activity that the club will have, we will keep track of the unique name of the activity, the hour, date, place that it was held, the capacity of the members, and the name of the guest which will not always be part of the activity.

- For each certificate that a member will earn, we will keep track of the unique Id of the certificate, the result of the certificate, and the person the signed it.

- One member or more than one can be part of one club or many clubs, and one or many clubs can have one or many members.

- One member or many of them can be part of at least three activities or more than three, and one or many activities can hold one or many members.

- Only one member can earn one or more certificates, and one or more certificates can be earned by only one member.

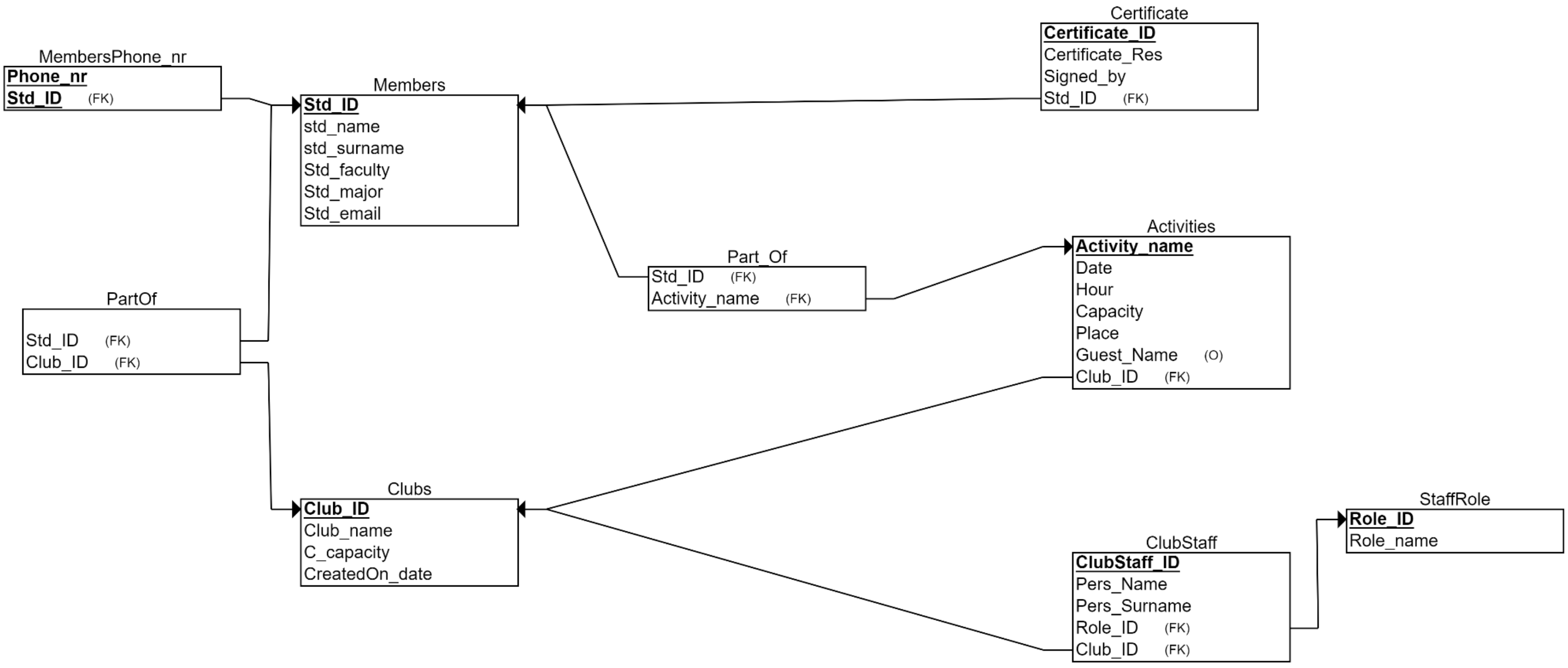
- One club can have one or more activities, and one or more activities can be held by only one club.

- One member of the club's staff or many of them have only one role, and only one role can be convenient for one or many members of the club's staff.

- One member of the club's staff or many of them are part of only one club, and only one club can have one or many members of the club's staff.

5. Relational Schema

5.1 Explanation of Relational Schema



➢ Members with MembersPhone_nr

Because of the multivalued attribute Phone_nr of the entity Members, a MemberPhone_nr relation is created. In this relation multiple rows of different phone numbers belonging to the same employees are possible. Neither column in the MemberPhone_nr relation is unique but each combination of Std_ID and Phone_nr is unique. Therefore, in the MemberPhone_nr relation, Std_ID and Phone_nr form a composite primary key.

➢ Certificate with Members

In this 1-to-M relationship, given that the entity Certificate is on the M side of the 1:M relationship, the resulting relation Certificate has an additional foreign key, column MembersID that corresponds to the primary key of relation Members.
The relation schema contains a line pointing from the foreign key column MembersID(Std_ID) in the relation Certificate to the corresponding primary key column MembersID in the relation Members.

- Activities with Clubs

In this 1-to-M relationship, given that the entity Activities is on the M side of the 1:M relationship, the resulting relation Activities has an additional foreign key, column Club_ID that corresponds to the primary key of relation Clubs. The relation schema contains a line pointing from the foreign key column Club_ID in the relation Activities to the corresponding primary key column Clubs_ID in the relation Clubs.

- ClubStaff with Clubs

In this 1 to-S relationship, given that the entity Club_Staff is on the M side of the 1:M relationship, the resulting relation Club_Staff has an additional foreign key, column Club_ID that corresponds to the primary key of relation Clubs. The relation schema contains a line pointing from the foreign key column Club_ID in the relation Club_Staff to the corresponding primary key column Club_ID in the relation Clubs.

- Club_Staff with Staff_Role

In this 1-to-M relationship, given that the entity Club_Staff is on the M side of the 1: M relationship, the resulting relation Club_Staff has an additional foreign key, column ClubStaff_ID that corresponds to the primary key of relation Staff_Role. The relation schema contains a line pointing from the foreign key column ClubStaff_ID in the relation Club_Staff to the corresponding primary key column ClubStaff_ID in the relation Staff_Role.

- Members with Clubs

This relation represents an M: N relationship, and has two foreign keys, each depicted by the lines pointing from each of the foreign keys to its source. One line points from the foreign key Std_ID to the primary key Std_ID in relation Members. Another line points from the foreign key Club_ID to the primary key Club_ID in relation Clubs. Foreign keys Std_ID and Club_ID are underlined because together they form the composite primary key of the relation PartOf.

- Members with Activities

This relation which represents an M: N relationship, has two foreign keys, each depicted by the lines pointing from each of the foreign keys to its source. One line points from the foreign key Std_ID to the primary key Std_ID in relation Members. Another line points from the foreign key Activity_Name to the primary key Activity_Name in relation Clubs. Foreign keys Std_ID and Activity_Name are underlined because together they form the composite primary key of the relation PartOf.

6. SQL

6.1. Create Tables Explanation

- The SQL command Create Table is used for creating and connecting relational tables. This statement is to create the tables depicted by the relation schema (RS), which we mentioned above. Both the Entity Relation Schema (ERD) and the (RS) help in creating these tables, as they show the connection between these entities and the attributes that each entity holds. We created each table based on the rules that we learned during the DBMS course, and successfully we created the table. This table helped us to organize any type of content, whether it is text or numerical data.

- In every table, we write the name of the table,the same as the name that they have in the ERD schema. We first chose to create table members, as there isn't any foreign key (FK). We wrote the Create Table statement and the name of the table. We did this process for all the tables in the right order. Each of the entities has its own attributes. These will be declared inside each table that we create. After we created members, inside we put the attributes, just like we mentioned before. We declare for each, their data type and the primary key (PK) for that table if there is one. There are tables that contain Foreign Key (FK). We first create the table with the (PK) attribute and then the table with the (FK), as that (FK) refers to the (PK) in the previous table. In this way, we connected all these tables.

```
CREATE TABLE Members
( std_id      char(9)        NOT NULL,
  std_name    varchar(15)    NOT NULL,
  std_surname varchar(15)    NOT NULL,
  std_faculty varchar(35)    NOT NULL,
  std_major   varchar(15)    NOT NULL,
  std_email   varchar(25)    NOT NULL,
  PRIMARY KEY (std_id));
```

```
CREATE TABLE Certificate
(
  certificate_id   char(6)  NOT NULL,
  certificate_res  char(6)  NOT NULL,
  std_id           char(9)  NOT NULL,
  signed_by        varchar(15) NOT NULL,
  PRIMARY key (certificate_id),
  FOREIGN KEY (std_id) references Members
(std_id));
…
```

6.2. Insert Values Explanation

The INSERT part for the table “Members”:

```
INSERT Into Members VALUES('02052106','Era','Mulla','Department of Computer Engineering','Software Engineering','emulla21@epoka.edu.al');
INSERT Into Members VALUES('02052107','Greisi','Jaho','Department of Computer Engineering','Software Engineering','gjaho21@epoka.edu.al');
INSERT Into Members VALUES('02052108','David','Keci','Department of Computer Engineering','Software Engineering','dkeci21@epoka.edu.al');
INSERT Into Members VALUES('02052109','Klea','Haxhiu','Department of Computer Engineering','Software Engineering','khaxhiu21@epoka.edu.al');
INSERT Into Members VALUES('02052110','Erisa','Zaimi','Department of Computer Engineering','Software Engineering','ezaimi21@epoka.edu.al');
INSERT Into Members VALUES('12052206','Klea','Bega','Department of Architecture','Architecture','kbega20@epoka.edu.al');
INSERT Into Members VALUES('12052207','Frenci','Jani','Department of Architecture','Architecture','fjani21@epoka.edu.al');
INSERT Into Members VALUES('12052208','Deni','Diu','Department of Architecture','Architecture','ddiu20@epoka.edu.al');
INSERT Into Members VALUES('12052209','Fiona','Haxhiaj','Department of Architecture','Architecture','fhaxhiaj22@epoka.edu.al');
INSERT Into Members VALUES('12052210','Daniel','Nelaj','Department of Architecture','Architecture','dnelaj22@epoka.edu.al');
INSERT Into Members VALUES('22042006','Berti','Roku','Department of Economics','Economicse','broku20@epoka.edu.al');
INSERT Into Members VALUES('22042007','Adrian','Kalaja','Department of Economics','Economicse','akalaja22@epoka.edu.al');
INSERT Into Members VALUES('22042008','Flamur','Begu','Department of Economics','Economicse','fbegu20@epoka.edu.al');
INSERT Into Members VALUES('22042009','Kristia','Malaj','Department of Economics','Economicse','kmalaj21@epoka.edu.al');
INSERT Into Members VALUES('22042010','Kosta','Fresku','Department of Economics','Economicse','kfresku22@epoka.edu.al');
INSERT Into Members VALUES('32042103','Egli','Soti','Department of Banking and Finance','Banking and Finance','esoti22@epoka.edu.al');
INSERT Into Members VALUES('32042102','Elio','Valari','Department of Banking and Finance','Banking and Finance','evalari22@epoka.edu.al');
INSERT Into Members VALUES('32042101','Irisia','Mihali','Department of Banking and Finance','Banking and Finance','imihali21@epoka.edu.al');
```

```sql
INSERT Into Members VALUES('32042100','Xhesi','Vucaj','Department of Banking and Finance','Banking and Finance','xhvucaj21@epoka.edu.al');
INSERT Into Members VALUES('32042110','Dea','Deriku','Department of Banking and Finance','Banking and Finance','dderiku20@epoka.edu.al');
INSERT Into Members VALUES('42052108','Kejsi','Besimi','Department of Business Administration','Business Administration','kbesimi20@epoka.edu.al');
INSERT Into Members VALUES('42052109','Katia','Mustafaraj','Department of Business Administration','Business Administration','kmustafaraj21@epoka.edu.al');
INSERT Into Members VALUES('42052104','Lise','Betaj','Department of Business Administration','Business Administration','lbetaj22@epoka.edu.al');
INSERT Into Members VALUES('42052107','Saimir','Dallga','Department of Business Administration','Business Administration','sdallga22@epoka.edu.al');
INSERT Into Members VALUES('42052118','Odeta','Ujku','Department of Business Administration','Business Administration','oujku20@epoka.edu.al');
INSERT Into Members VALUES('52042103','Beart','Rakipaj','Department of Business Informatics','Business Informatics','brakipaj22@epoka.edu.al');
INSERT Into Members VALUES('52042102','Frenk','Baroku','Department of Business Informatics','Business Informatics','fbaroku21@epoka.edu.al');
INSERT Into Members VALUES('52042101','Antonela','Gjinaj','Department of Business Informatics','Business Informatics','agjinaj22@epoka.edu.al');
INSERT Into Members VALUES('52042100','Valdrin','Ballo','Department of Business Informatics','Business Informatics','vballo20@epoka.edu.al');
INSERT Into Members VALUES('52042110','Olta','Krena','Department of Business Informatics','Business Informatics','okrena@epoka.edu.al');
INSERT Into Members VALUES('62052240','Denisse','Shaw','Department of Civil Engineering','Civil Engineering','dshaw20@epoka.edu.al');
INSERT Into Members VALUES('62052241','Albert','Prendi','Department of Civil Engineering','Civil Engineering','aprndi20@epoka.edu.al');
INSERT Into Members VALUES('62052242','Denis','Syla','Department of Civil Engineering','Civil Engineering','dsyla20@epoka.edu.al');
INSERT Into Members VALUES('62052243','Klement',' Çela','Department of Civil Engineering','Civil Engineering','kcela20@epoka.edu.al');
INSERT Into Members VALUES('62052244','Denisse','Dervishi','Department of Civil Engineering','Civil Engineering','ddervishi20@epoka.edu.al');
INSERT Into Members VALUES('72052270','Sinan','Kola','Department of Law','Civil Engineering','skola20@epoka.edu.al');
INSERT Into Members VALUES('72052271','Arben','Muco','Department of Law','Civil Engineering','amuco20@epoka.edu.al');
INSERT Into Members VALUES('72052272','Megi','Ruci','Department of Law','Civil Engineering','mruci20@epoka.edu.al');
INSERT Into Members VALUES('72052273','Sara',' Spahiu','Department of Law','Civil
Engineering','sspahiu20@epoka.edu.al');
INSERT Into Members VALUES('72052274','Kjara','Halili','Department of Law','Civil
Engineering','khalili20@epoka.edu.al');
INSERT Into Members VALUES('82052290','Samanta','Cungu','Department of Political Science and
International Relations','Political Science and International Relations','scungu20@epoka.edu.al');
INSERT Into Members VALUES('82052291','Melisa','Tafili','Department of Political Science and
International Relations','Political Science and International Relations','mtafili20@epoka.edu.al');
INSERT Into Members VALUES('82052292','Kejsi','Marku','Department of Political Science and
International Relations','Political Science and International Relations','kmarku20@epoka.edu.al');
INSERT Into Members VALUES('82052293','Alda',' Pjetri','Department of Political Science and
International Relations','Political Science and International Relations','apjetri20@epoka.edu.al');
INSERT Into Members VALUES('82052294','Aldo','Pemaj','Department of Political Science and
International Relations','Political Science and International Relations','apemaj20@epoka.edu.al');
```

Explanation:

- For this part of SQL we have followed the same method of inserting in all tables. The table that we have chosen as an example for the documentation is the one above “Members". For the first row of this table insertions→ (INSERT Into Members VALUES('02052106','Era','Mulla','Department of Computer Engineering','Software Engineering','emulla21@epoka.edu.al');), we have inserted the id of the member as the first value (02052106), followed by the name (Era), surname (Mulla), department (Department of Computer Engineering), major she is taking (Software Engineering) and the email (emulla21@epoka.edu.al). This is the way we have populated the insertion part of the SQL for the table Members and all the other ones.

6.3. Managerial Queries

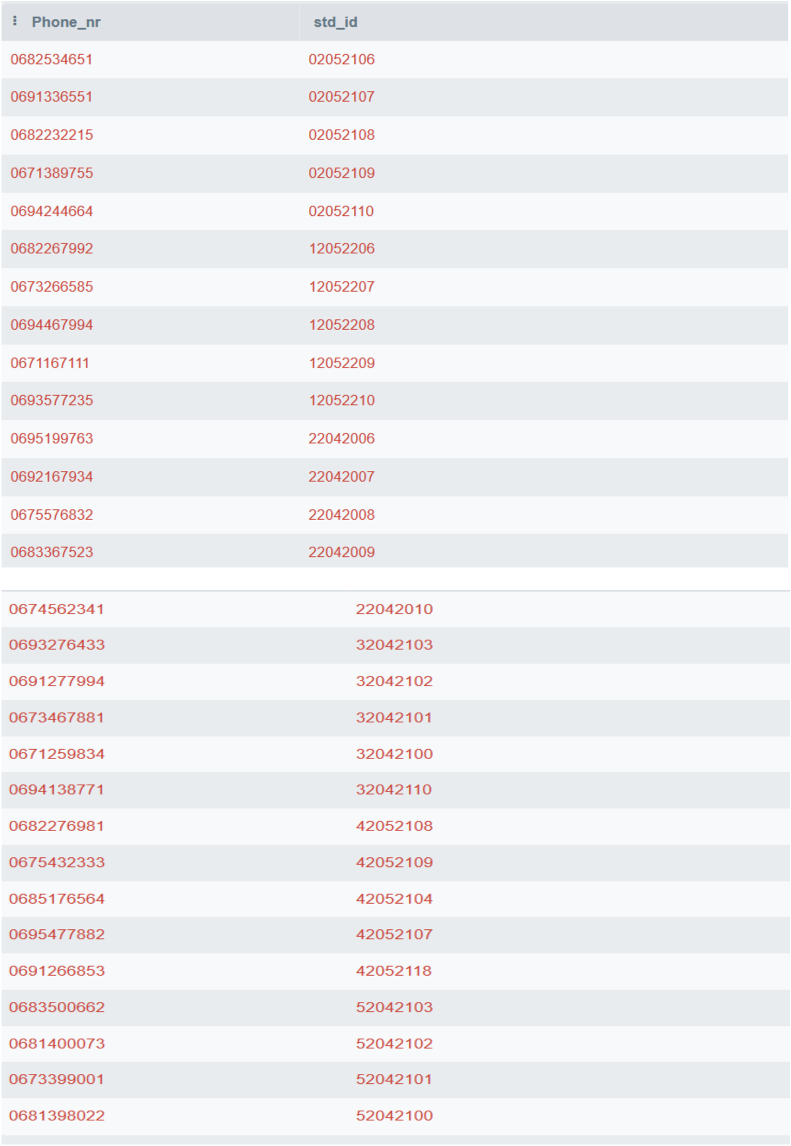
1. ***Select***

a) Retrieve the entire contents of the relation ACTIVITIES.

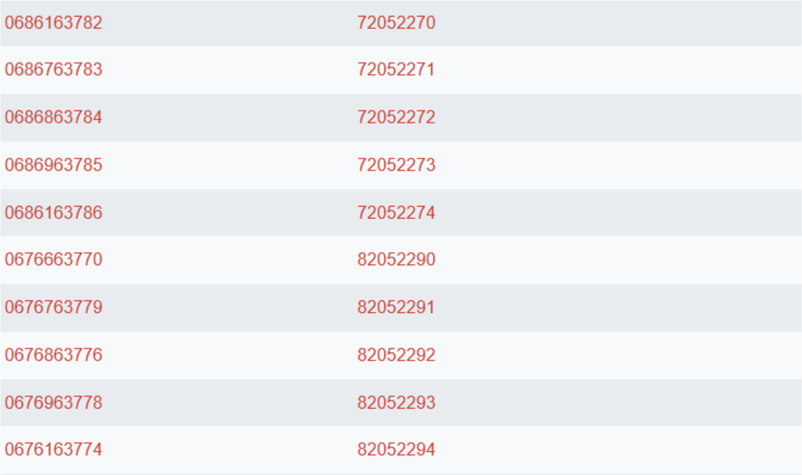
SELECT date, hour, capacity, place, guest_name,club_id
FROM Activities;

b) Retrieve the entire contents of the table MemmbersPhone_nr

SELECT *
FROM MemebersPhone_nr;



Phone_nr	std_id
0682534651	02052106
0691336551	02052107
0682232215	02052108
0671389755	02052109
0694244664	02052110
0682267992	12052206
0673266585	12052207
0694467994	12052208
0671167111	12052209
0693577235	12052210
0695199763	22042006
0692167934	22042007
0675576832	22042008
0683367523	22042009
0674562341	22042010
0693276433	32042103
0691277994	32042102
0673467881	32042101
0671259834	32042100
0694138771	32042110
0682276981	42052108
0675432333	42052109
0685176564	42052104
0695477882	42052107
0691266853	42052118
0683500662	52042103
0681400073	52042102
0673399001	52042101
0681398022	52042100



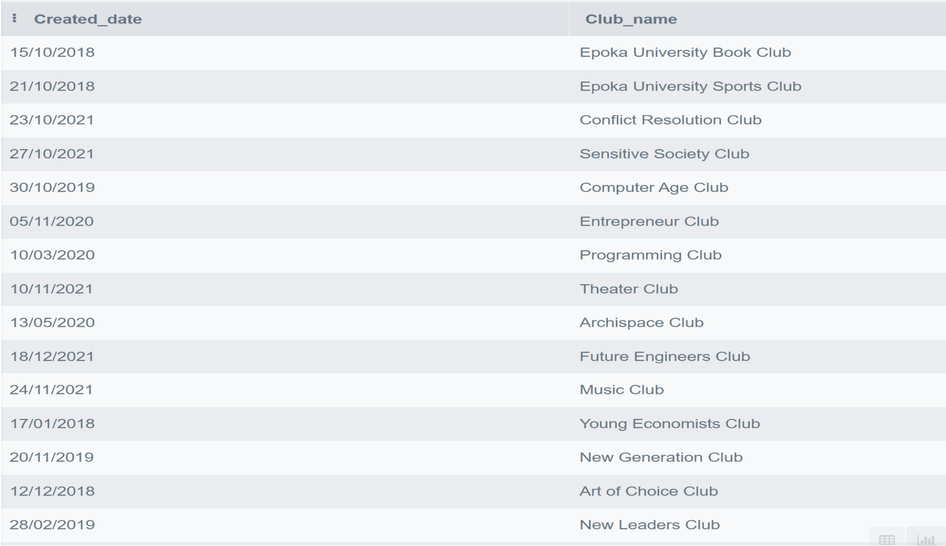
0686163782	72052270
0686763783	72052271
0686863784	72052272
0686963785	72052273
0686163786	72052274
0676663770	82052290
0676763779	82052291
0676863776	82052292
0676963778	82052293
0676163774	82052294

c) Retrieve the student id and who signed the certificate from table Certificate.

```
SELECT std_id,signed_by
FROM Certificate;
```

d) Retrieves the date of creation and the name of each club .

```
Select created_date, club_name
FROM Clubs;
```



Created_date	Club_name
15/10/2018	Epoka University Book Club
21/10/2018	Epoka University Sports Club
23/10/2021	Conflict Resolution Club
27/10/2021	Sensitive Society Club
30/10/2019	Computer Age Club
05/11/2020	Entrepreneur Club
10/03/2020	Programming Club
10/11/2021	Theater Club
13/05/2020	Archispace Club
18/12/2021	Future Engineers Club
24/11/2021	Music Club
17/01/2018	Young Economists Club
20/11/2019	New Generation Club
12/12/2018	Art of Choice Club
28/02/2019	New Leaders Club

e) Retrieves the role id and the role name from table StaffRole .

```
select role_id, role_name
from StaffRole
```

2. ***Where***

a) Retrieve the activity_name, date, and hour for each activity whose club_id is 001.

SELECT activity_name, date, hour

FROM Activities

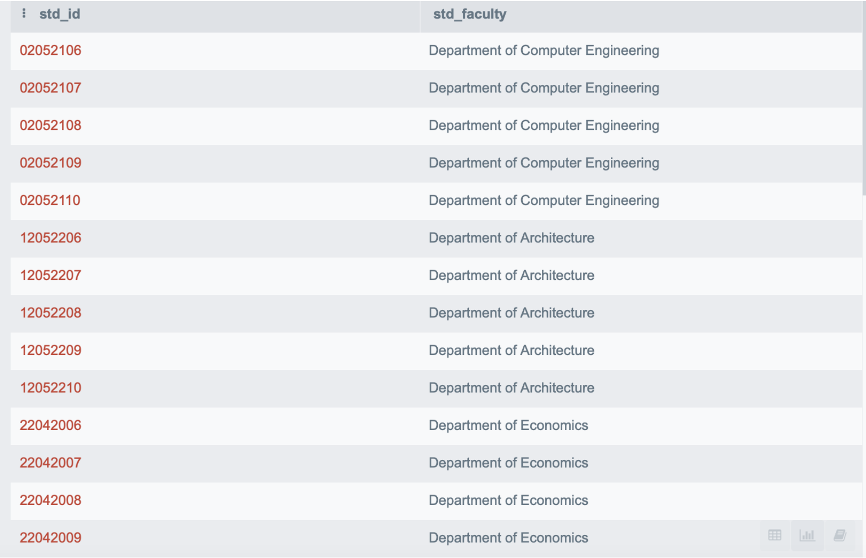
where club_id = '001';

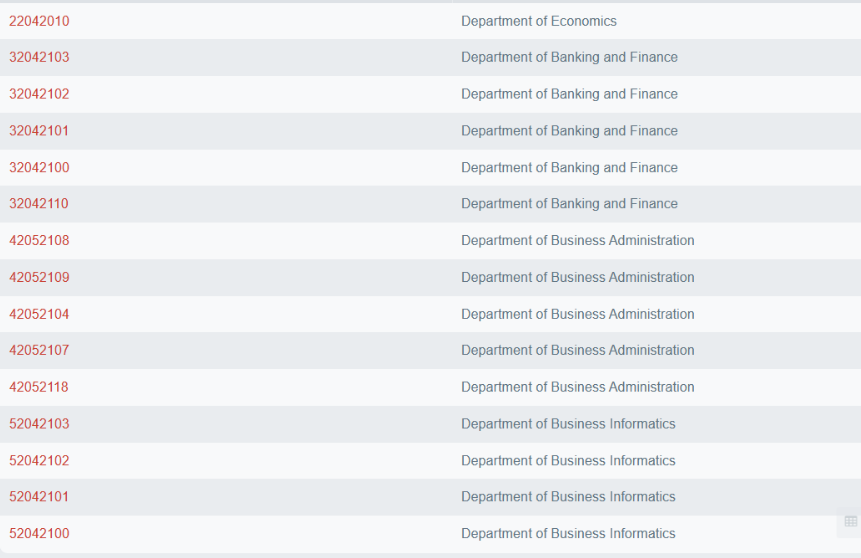
b) Retrieve std_id and std_faculty that have a club_id of 010.

SELECT std_id,std_faculty

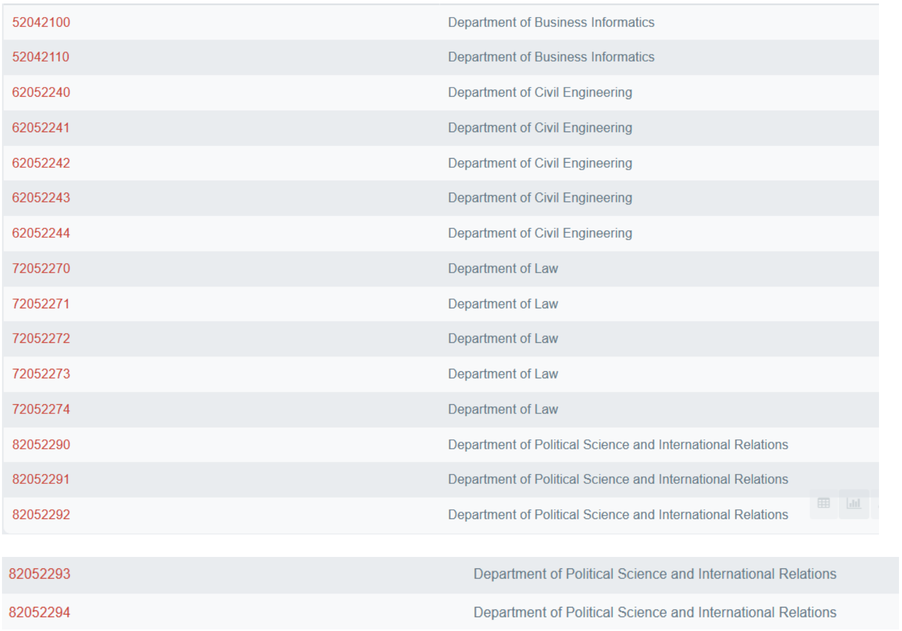
From Members,Clubs

Where club_id="010";





std_id	std_faculty
02052106	Department of Computer Engineering
02052107	Department of Computer Engineering
02052108	Department of Computer Engineering
02052109	Department of Computer Engineering
02052110	Department of Computer Engineering
12052206	Department of Architecture
12052207	Department of Architecture
12052208	Department of Architecture
12052209	Department of Architecture
12052210	Department of Architecture
22042006	Department of Economics
22042007	Department of Economics
22042008	Department of Economics
22042009	Department of Economics
22042010	Department of Economics
32042103	Department of Banking and Finance
32042102	Department of Banking and Finance
32042101	Department of Banking and Finance
32042100	Department of Banking and Finance
32042110	Department of Banking and Finance
42052108	Department of Business Administration
42052109	Department of Business Administration
42052104	Department of Business Administration
42052107	Department of Business Administration
42052118	Department of Business Administration
52042103	Department of Business Informatics
52042102	Department of Business Informatics
52042101	Department of Business Informatics
52042100	Department of Business Informatics



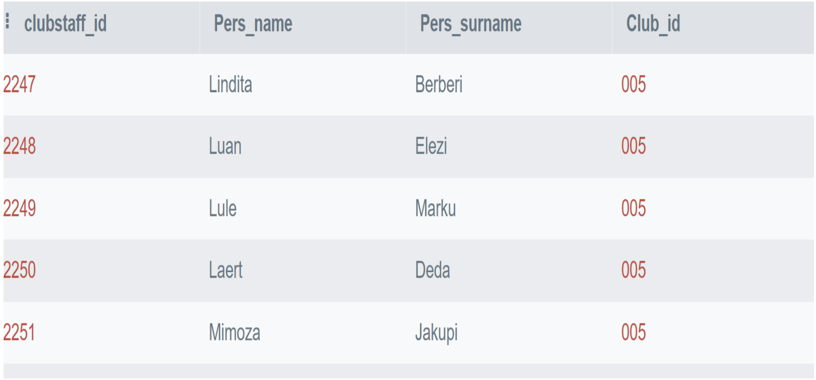
52042100	Department of Business Informatics
52042110	Department of Business Informatics
62052240	Department of Civil Engineering
62052241	Department of Civil Engineering
62052242	Department of Civil Engineering
62052243	Department of Civil Engineering
62052244	Department of Civil Engineering
72052270	Department of Law
72052271	Department of Law
72052272	Department of Law
72052273	Department of Law
72052274	Department of Law
82052290	Department of Political Science and International Relations
82052291	Department of Political Science and International Relations
82052292	Department of Political Science and International Relations
82052293	Department of Political Science and International Relations
82052294	Department of Political Science and International Relations

c) Retrieve clubstaff_id,Pers_name,club_id ,whose club id is 005.

Select clubstaff_id,Pers_name,Pers_surname,club_id

FROM ClubStaff

WHERE club_id="005";



clubstaff_id	Pers_name	Pers_surname	Club_id
2247	Lindita	Berberi	005
2248	Luan	Elezi	005
2249	Lule	Marku	005
2250	Laert	Deda	005
2251	Mimoza	Jakupi	005

3. ***Distinct***

a) Show one instance of all the different guest_name in the relation ACTIVITIES.

SELECT DISTINCT guest_name

FROM Activities;

b) Show one instance of all the different club_id smaller or equal to 003 from ClubStaff.

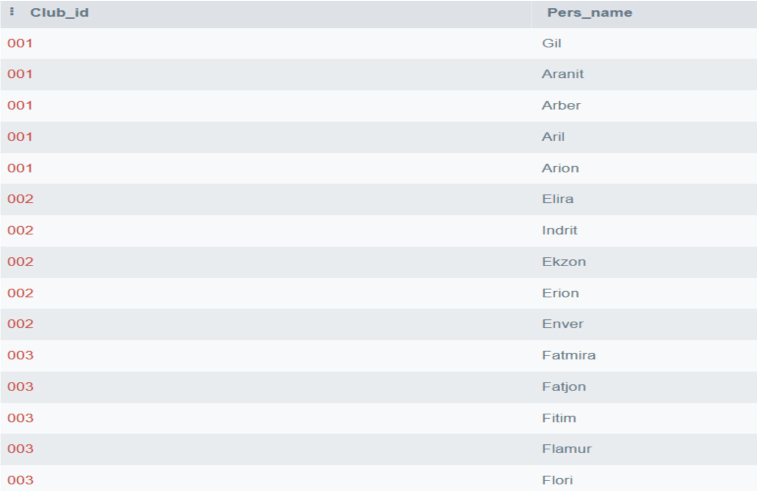
Select DISTINCT club_id,Pers_name

From ClubStaff

WHERE club_id<='003';

c) Show one instance of student id that are in table PartOf.

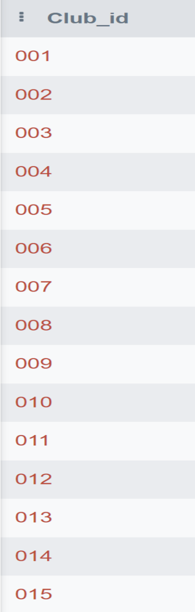
```
select  DISTINCT std_id
from PartOf
```



Club_id	Pers_name
001	Gil
001	Aranit
001	Arber
001	Aril
001	Arion
002	Elira
002	Indrit
002	Ekzon
002	Erion
002	Enver
003	Fatmira
003	Fatjon
003	Fitim
003	Flamur
003	Flori

d) Show one instance of all the different od club id values

```
SELECT  DISTINCT club_id
 FROM PartOf
```

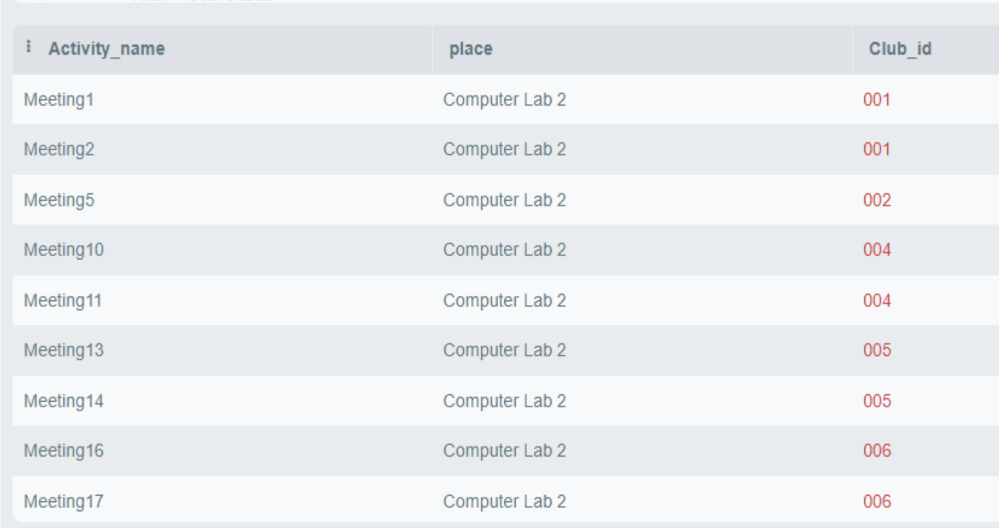


Club_id
001
002
003
004
005
006
007
008
009
010
011
012
013
014
015

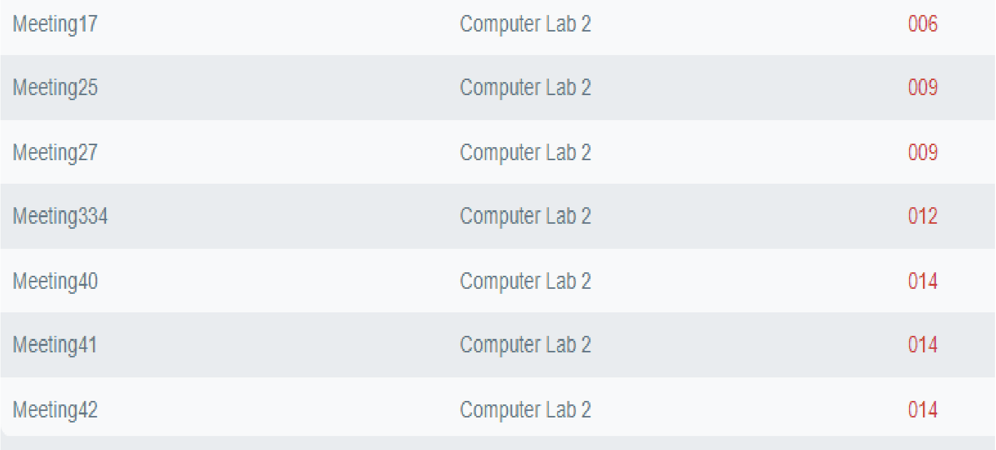
4. ***Order By***

a) Retrieve the activity_name, place, and club_id for each activity that will be held in the Computer Lab 2 place, sorted by club_id.

```
SELECT activity_name, place, club_id
FROM Activities
WHERE place = 'Computer Lab 2'
ORDER By club_id;
```



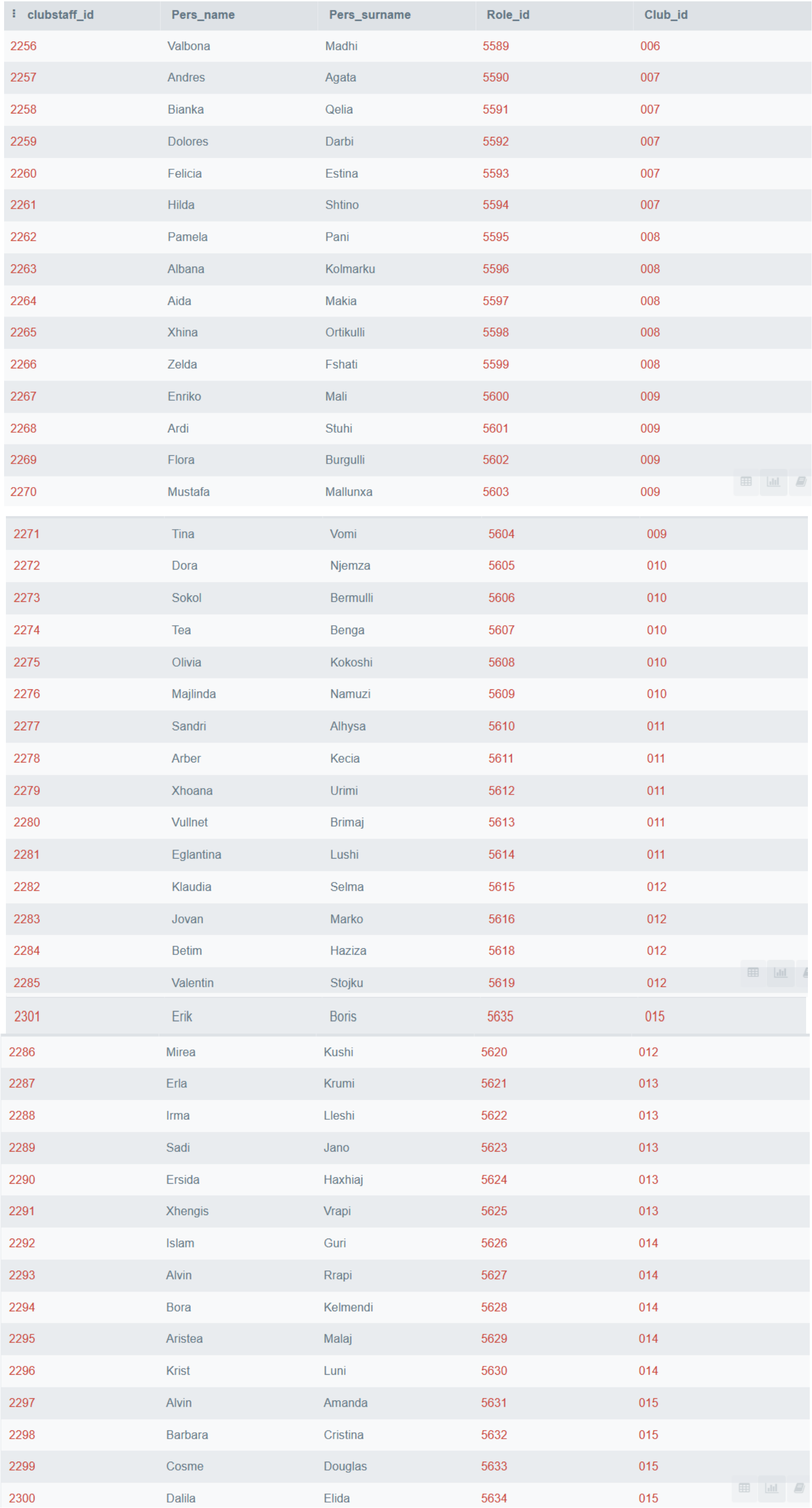
Activity_name	place	Club_id
Meeting1	Computer Lab 2	001
Meeting2	Computer Lab 2	001
Meeting5	Computer Lab 2	002
Meeting10	Computer Lab 2	004
Meeting11	Computer Lab 2	004
Meeting13	Computer Lab 2	005
Meeting14	Computer Lab 2	005
Meeting16	Computer Lab 2	006
Meeting17	Computer Lab 2	006



Meeting17	Computer Lab 2	006
Meeting25	Computer Lab 2	009
Meeting27	Computer Lab 2	009
Meeting334	Computer Lab 2	012
Meeting40	Computer Lab 2	014
Meeting41	Computer Lab 2	014
Meeting42	Computer Lab 2	014

b) Retrieve the clubstaff_id, Pers_name, Pers_surname,role_id,club_id for each club staff if the club clubstaff_id is greater than 2255, sorted by club_id

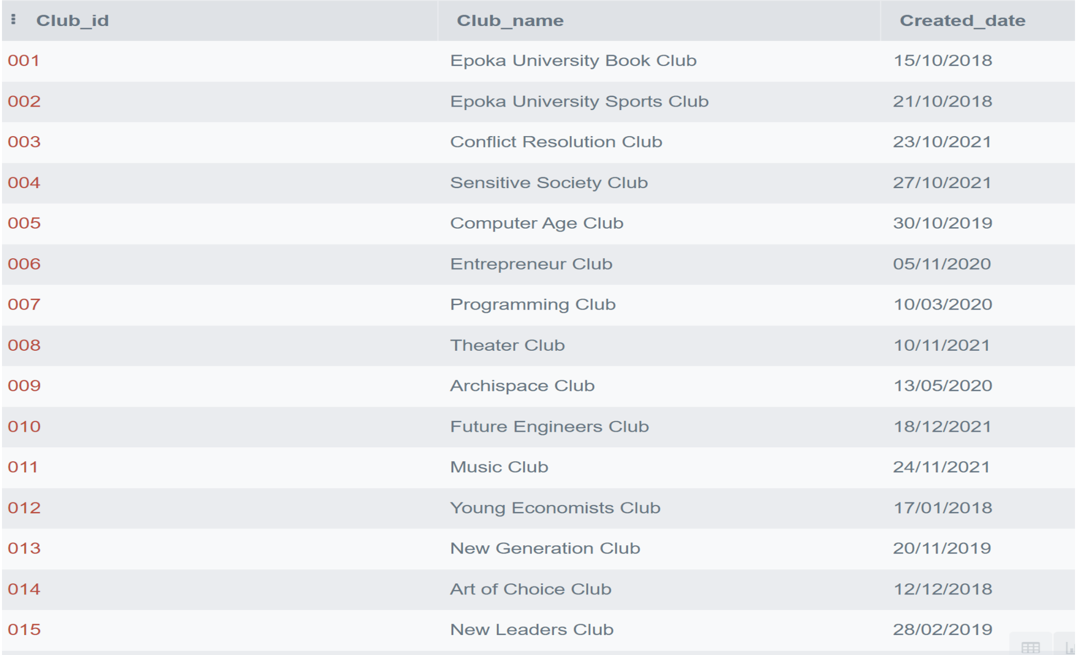
```
Select clubstaff_id, Pers_name, Pers_surname,role_id,club_id
From ClubStaff
Where clubstaff_id>'2255'
Order By club_id;
```



clubstaff_id	Pers_name	Pers_surname	Role_id	Club_id
2256	Valbona	Madhi	5589	006
2257	Andres	Agata	5590	007
2258	Bianka	Qelia	5591	007
2259	Dolores	Darbi	5592	007
2260	Felicia	Estina	5593	007
2261	Hilda	Shtino	5594	007
2262	Pamela	Pani	5595	008
2263	Albana	Kolmarku	5596	008
2264	Aida	Makia	5597	008
2265	Xhina	Ortikulli	5598	008
2266	Zelda	Fshati	5599	008
2267	Enriko	Mali	5600	009
2268	Ardi	Stuhi	5601	009
2269	Flora	Burgulli	5602	009
2270	Mustafa	Mallunxa	5603	009
2271	Tina	Vomi	5604	009
2272	Dora	Njemza	5605	010
2273	Sokol	Bermulli	5606	010
2274	Tea	Benga	5607	010
2275	Olivia	Kokoshi	5608	010
2276	Majlinda	Namuzi	5609	010
2277	Sandri	Alhysa	5610	011
2278	Arber	Kecia	5611	011
2279	Xhoana	Urimi	5612	011
2280	Vullnet	Brimaj	5613	011
2281	Eglantina	Lushi	5614	011
2282	Klaudia	Selma	5615	012
2283	Jovan	Marko	5616	012
2284	Betim	Haziza	5618	012
2285	Valentin	Stojku	5619	012
2301	Erik	Boris	5635	015
2286	Mirea	Kushi	5620	012
2287	Erla	Krumi	5621	013
2288	Irma	Lleshi	5622	013
2289	Sadi	Jano	5623	013
2290	Ersida	Haxhiaj	5624	013
2291	Xhengis	Vrapi	5625	013
2292	Islam	Guri	5626	014
2293	Alvin	Rrapi	5627	014
2294	Bora	Kelmendi	5628	014
2295	Aristea	Malaj	5629	014
2296	Krist	Luni	5630	014
2297	Alvin	Amanda	5631	015
2298	Barbara	Cristina	5632	015
2299	Cosme	Douglas	5633	015
2300	Dalila	Elida	5634	015

c) Retrieve club id, name, the date that they are created and sorted by club id.

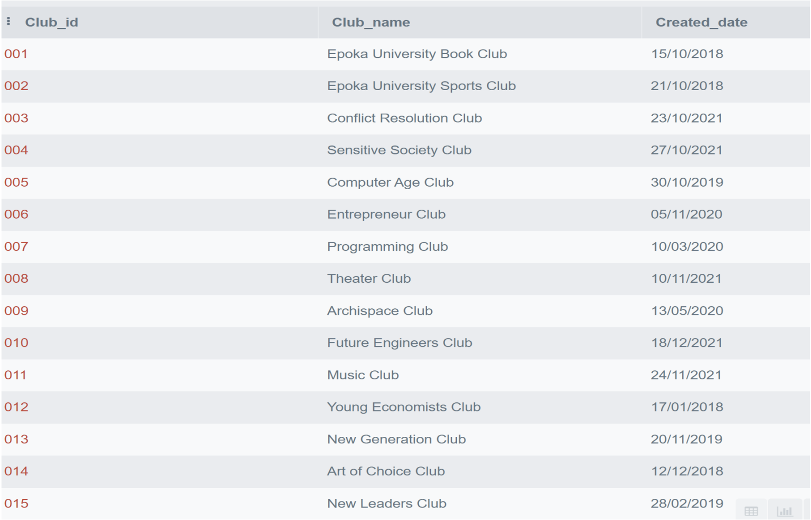
```
SELECT club_id,club_name,created_date
FROM Clubs
ORDER By club_id
```



Club_id	Club_name	Created_date
001	Epoka University Book Club	15/10/2018
002	Epoka University Sports Club	21/10/2018
003	Conflict Resolution Club	23/10/2021
004	Sensitive Society Club	27/10/2021
005	Computer Age Club	30/10/2019
006	Entrepreneur Club	05/11/2020
007	Programming Club	10/03/2020
008	Theater Club	10/11/2021
009	Archispace Club	13/05/2020
010	Future Engineers Club	18/12/2021
011	Music Club	24/11/2021
012	Young Economists Club	17/01/2018
013	New Generation Club	20/11/2019
014	Art of Choice Club	12/12/2018
015	New Leaders Club	28/02/2019

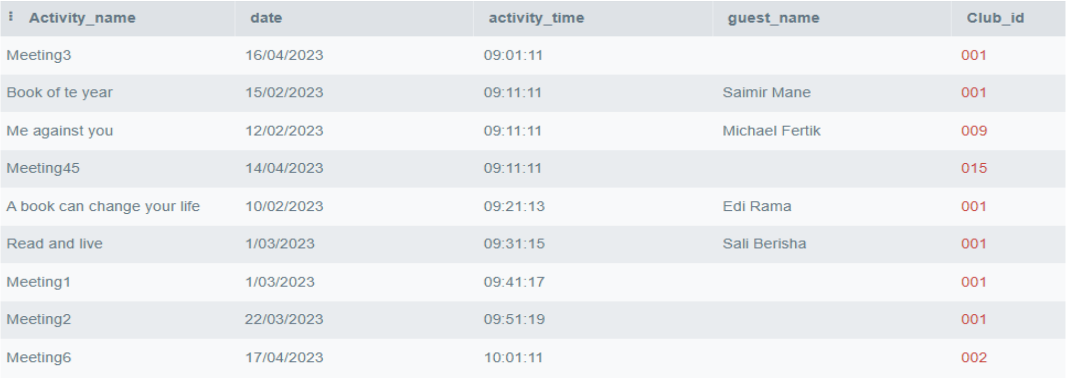
d) Retrieve person name and person surname that are sorted by club id.

```
SELECT  Pers_name, Pers_surname
FROM ClubStaff
Order by club_id
```

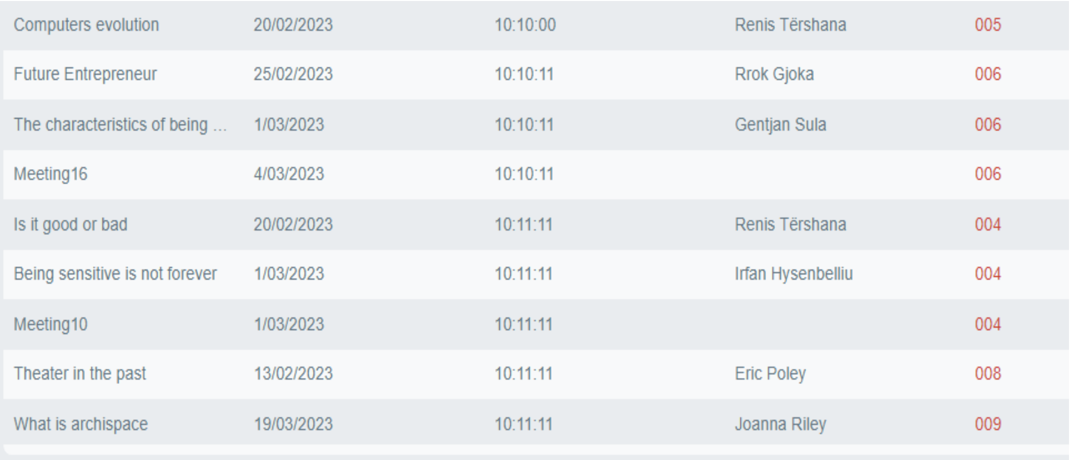


Club_id	Club_name	Created_date
001	Epoka University Book Club	15/10/2018
002	Epoka University Sports Club	21/10/2018
003	Conflict Resolution Club	23/10/2021
004	Sensitive Society Club	27/10/2021
005	Computer Age Club	30/10/2019
006	Entrepreneur Club	05/11/2020
007	Programming Club	10/03/2020
008	Theater Club	10/11/2021
009	Archispace Club	13/05/2020
010	Future Engineers Club	18/12/2021
011	Music Club	24/11/2021
012	Young Economists Club	17/01/2018
013	New Generation Club	20/11/2019
014	Art of Choice Club	12/12/2018
015	New Leaders Club	28/02/2019

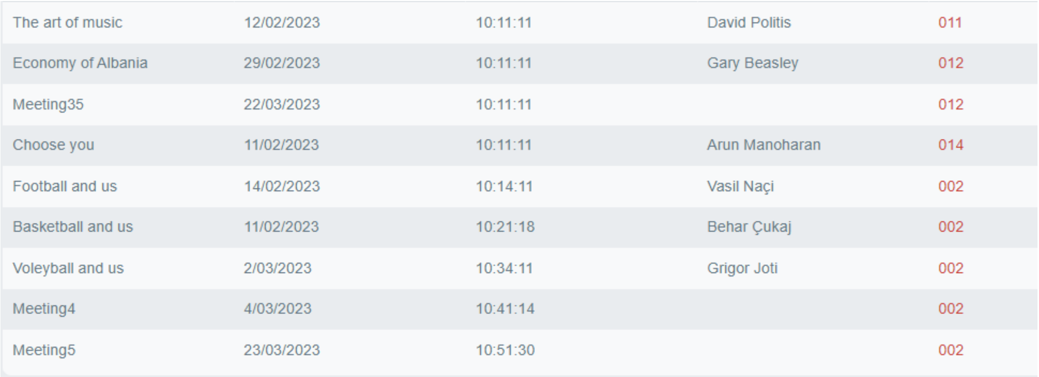
e) Retrieve activity_name, date, hour named as activity_name,guest_name and club_id being ordered in ascending order.
SELECT activity_name, date, hour AS activity_time,guest_name,club_id
FROM Activities
ORDER by hour ASC;



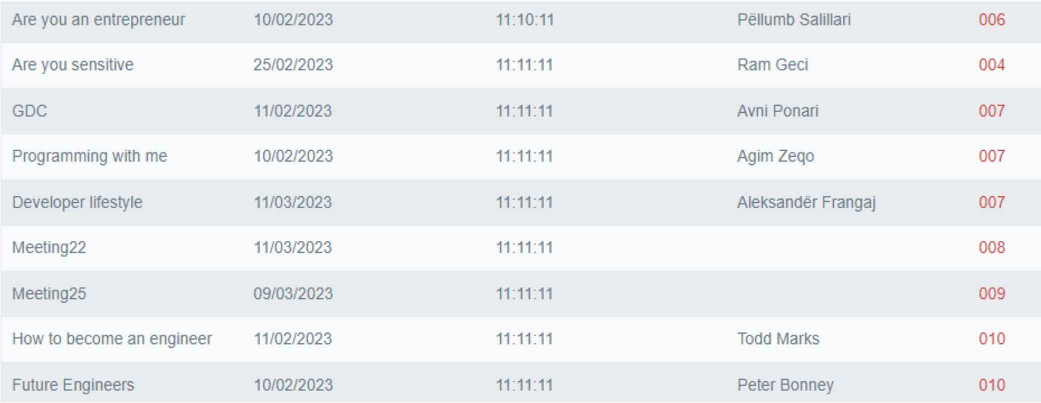
Activity_name	date	activity_time	guest_name	Club_id
Meeting3	16/04/2023	09:01:11		001
Book of te year	15/02/2023	09:11:11	Saimir Mane	001
Me against you	12/02/2023	09:11:11	Michael Fertik	009
Meeting45	14/04/2023	09:11:11		015
A book can change your life	10/02/2023	09:21:13	Edi Rama	001
Read and live	1/03/2023	09:31:15	Sali Berisha	001
Meeting1	1/03/2023	09:41:17		001
Meeting2	22/03/2023	09:51:19		001
Meeting6	17/04/2023	10:01:11		002



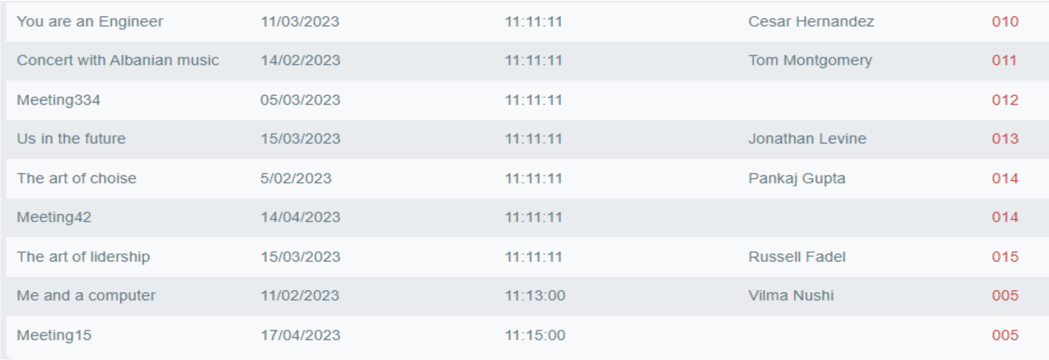
Computers evolution	20/02/2023	10:10:00	Renis Tërshana	005
Future Entrepreneur	25/02/2023	10:10:11	Rrok Gjoka	006
The characteristics of being ...	1/03/2023	10:10:11	Gentjan Sula	006
Meeting16	4/03/2023	10:10:11		006
Is it good or bad	20/02/2023	10:11:11	Renis Tërshana	004
Being sensitive is not forever	1/03/2023	10:11:11	Irfan Hysenbelliu	004
Meeting10	1/03/2023	10:11:11		004
Theater in the past	13/02/2023	10:11:11	Eric Poley	008
What is archispace	19/03/2023	10:11:11	Joanna Riley	009



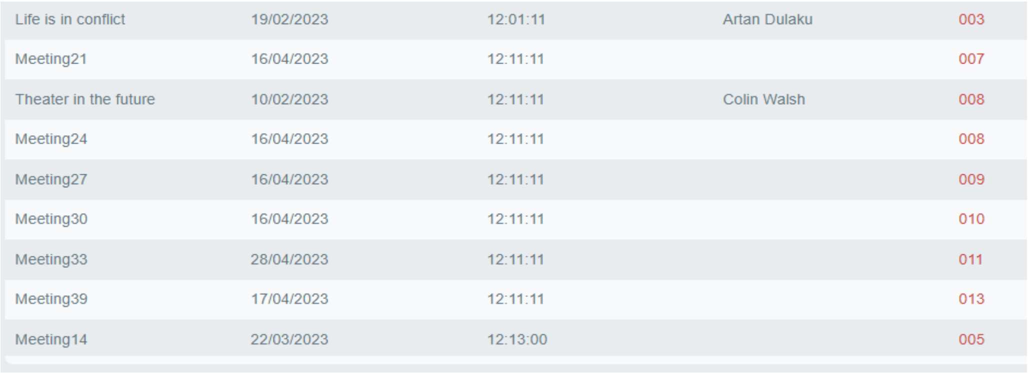
The art of music	12/02/2023	10:11:11	David Politis	011
Economy of Albania	29/02/2023	10:11:11	Gary Beasley	012
Meeting35	22/03/2023	10:11:11		012
Choose you	11/02/2023	10:11:11	Arun Manoharan	014
Football and us	14/02/2023	10:14:11	Vasil Naçi	002
Basketball and us	11/02/2023	10:21:18	Behar Çukaj	002
Voleyball and us	2/03/2023	10:34:11	Grigor Joti	002
Meeting4	4/03/2023	10:41:14		002
Meeting5	23/03/2023	10:51:30		002



Are you an entrepreneur	10/02/2023	11:10:11	Pëllumb Salillari	006
Are you sensitive	25/02/2023	11:11:11	Ram Geci	004
GDC	11/02/2023	11:11:11	Avni Ponari	007
Programming with me	10/02/2023	11:11:11	Agim Zeqo	007
Developer lifestyle	11/03/2023	11:11:11	Aleksandër Frangaj	007
Meeting22	11/03/2023	11:11:11		008
Meeting25	09/03/2023	11:11:11		009
How to become an engineer	11/02/2023	11:11:11	Todd Marks	010
Future Engineers	10/02/2023	11:11:11	Peter Bonney	010



You are an Engineer	11/03/2023	11:11:11	Cesar Hernandez	010
Concert with Albanian music	14/02/2023	11:11:11	Tom Montgomery	011
Meeting334	05/03/2023	11:11:11		012
Us in the future	15/03/2023	11:11:11	Jonathan Levine	013
The art of choise	5/02/2023	11:11:11	Pankaj Gupta	014
Meeting42	14/04/2023	11:11:11		014
The art of lidership	15/03/2023	11:11:11	Russell Fadel	015
Me and a computer	11/02/2023	11:13:00	Vilma Nushi	005
Meeting15	17/04/2023	11:15:00		005



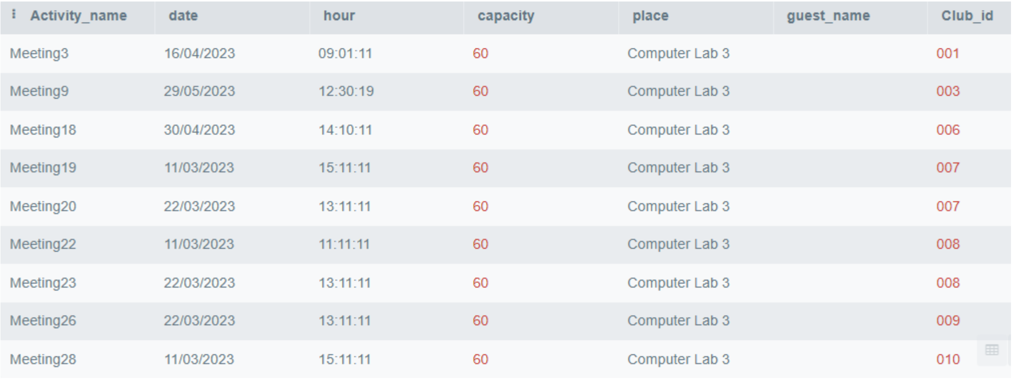
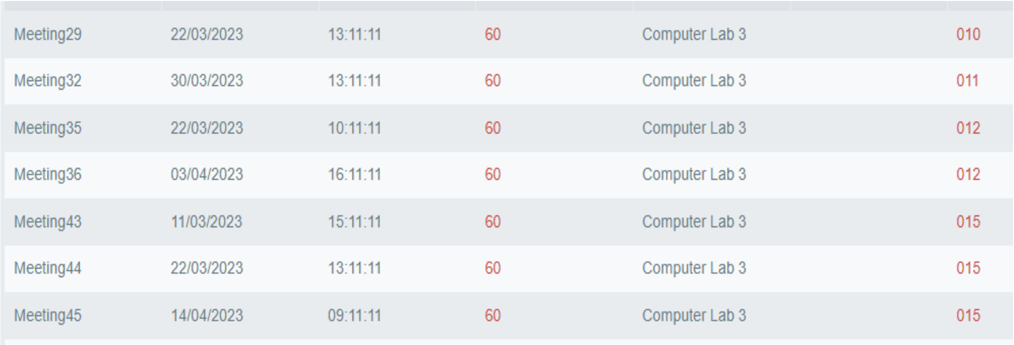
Life is in conflict	19/02/2023	12:01:11	Artan Dulaku	003
Meeting21	16/04/2023	12:11:11		007
Theater in the future	10/02/2023	12:11:11	Colin Walsh	008
Meeting24	16/04/2023	12:11:11		008
Meeting27	16/04/2023	12:11:11		009
Meeting30	16/04/2023	12:11:11		010
Meeting33	28/04/2023	12:11:11		011
Meeting39	17/04/2023	12:11:11		013
Meeting14	22/03/2023	12:13:00		005

……..

5. ***Like***

a) Retrieve the record for each activity whose place is in the "Computer Lab 3".

```
SELECT  *
FROM Activities
WHERE place LIKE '%Computer Lab 3%';
```

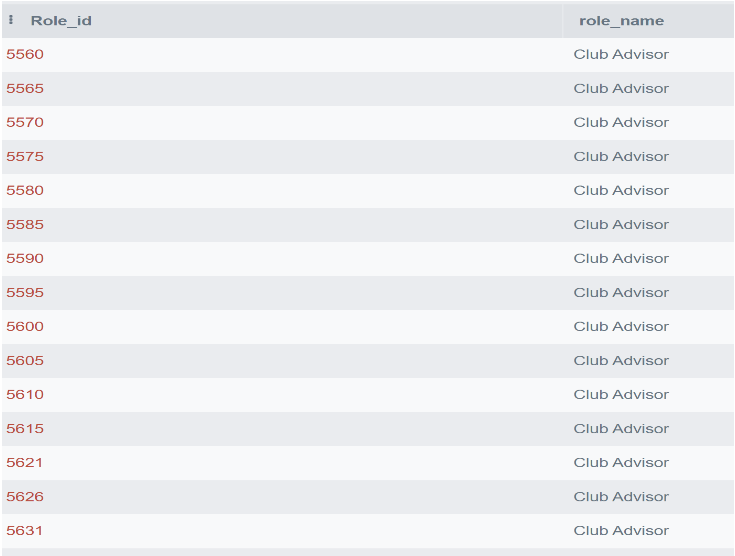
Activity_name	date	hour	capacity	place	guest_name	Club_id
Meeting3	16/04/2023	09:01:11	60	Computer Lab 3		001
Meeting9	29/05/2023	12:30:19	60	Computer Lab 3		003
Meeting18	30/04/2023	14:10:11	60	Computer Lab 3		006
Meeting19	11/03/2023	15:11:11	60	Computer Lab 3		007
Meeting20	22/03/2023	13:11:11	60	Computer Lab 3		007
Meeting22	11/03/2023	11:11:11	60	Computer Lab 3		008
Meeting23	22/03/2023	13:11:11	60	Computer Lab 3		008
Meeting26	22/03/2023	13:11:11	60	Computer Lab 3		009
Meeting28	11/03/2023	15:11:11	60	Computer Lab 3		010
Meeting29	22/03/2023	13:11:11	60	Computer Lab 3		010
Meeting32	30/03/2023	13:11:11	60	Computer Lab 3		011
Meeting35	22/03/2023	10:11:11	60	Computer Lab 3		012
Meeting36	03/04/2023	16:11:11	60	Computer Lab 3		012
Meeting43	11/03/2023	15:11:11	60	Computer Lab 3		015
Meeting44	22/03/2023	13:11:11	60	Computer Lab 3		015
Meeting45	14/04/2023	09:11:11	60	Computer Lab 3		015

b) Retrieve the record for each Activity whose Activity_name contains the phrase 'Future'.

```
Select *
from Activities
where activity_name LIKE '%Future%';
```

c) Retrieve role id and role name whose role name contains the phrase Club Advisor

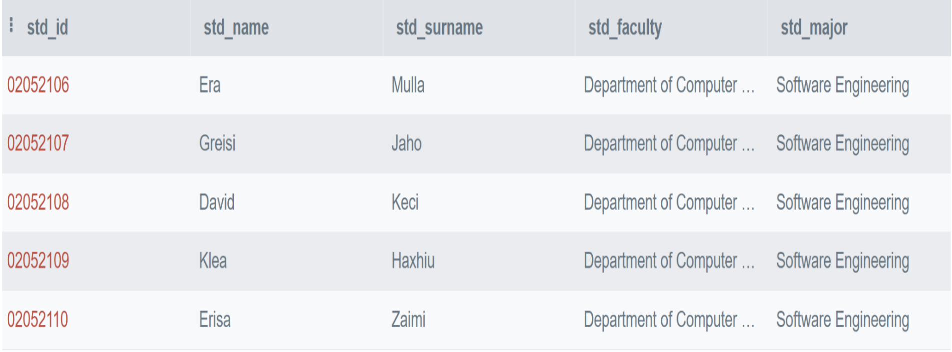
```
SELECT role_id,role_name
FROM StaffRole
WHERE role_name LIKE "%Club Advisor%";
```



Role_id	role_name
5560	Club Advisor
5565	Club Advisor
5570	Club Advisor
5575	Club Advisor
5580	Club Advisor
5585	Club Advisor
5590	Club Advisor
5595	Club Advisor
5600	Club Advisor
5605	Club Advisor
5610	Club Advisor
5615	Club Advisor
5621	Club Advisor
5626	Club Advisor
5631	Club Advisor

d) Retrieve student id, student name and surname, student faculty and major from table Members where student major contains the phrase Software Engineering.

```
SELECT std_id,std_name,std_surname,std_faculty,std_major
FROM Members
WHERE std_major LIKE "%Software Engineering%"
```



std_id	std_name	std_surname	std_faculty	std_major
02052106	Era	Mulla	Department of Computer ...	Software Engineering
02052107	Greisi	Jaho	Department of Computer ...	Software Engineering
02052108	David	Keci	Department of Computer ...	Software Engineering
02052109	Klea	Haxhiu	Department of Computer ...	Software Engineering
02052110	Erisa	Zaimi	Department of Computer ...	Software Engineering

e) Retrieve student id, student name, student faculty, student major and student email from table Members where name is equal to Fiona.

```
Select std_id , std_name, std_surname, std_faculty,  std_major, std_email
   FROM Members
   where std_name like 'Fiona'
```

6. ***Aggregate Functions***

a) Show how many activities will be held.

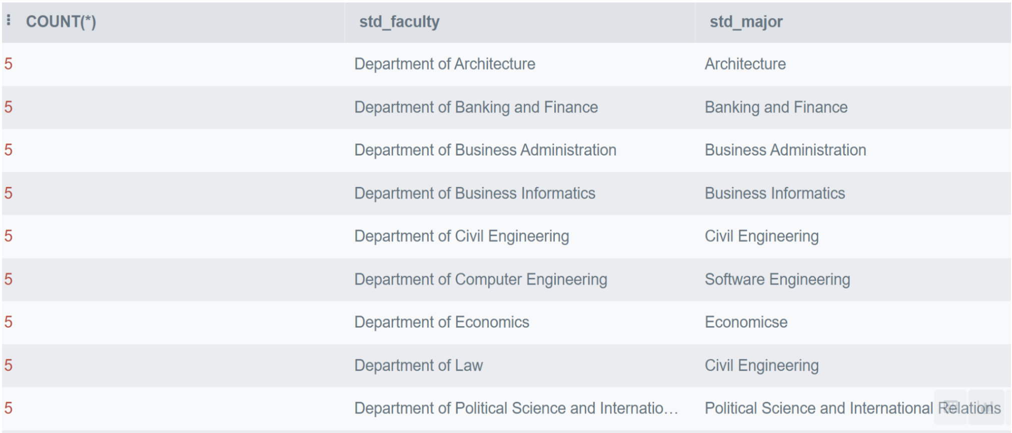
```
 SELECT COUNT(*)
FROM Activities;
```

b) Show how many students are part of clubs, whose id is smaller than 007.

```
Select COUNT(*)
from PartOf
where club_id <'007';
```

c) Retrieve the number of students for each dep and major, and the corresponding names

```
SELECT COUNT(*),std_faculty,std_major
FROM Members
GROUP BY std_faculty;
```



COUNT(*)	std_faculty	std_major
5	Department of Architecture	Architecture
5	Department of Banking and Finance	Banking and Finance
5	Department of Business Administration	Business Administration
5	Department of Business Informatics	Business Informatics
5	Department of Civil Engineering	Civil Engineering
5	Department of Computer Engineering	Software Engineering
5	Department of Economics	Economicse
5	Department of Law	Civil Engineering
5	Department of Political Science and Internatio…	Political Science and International Relations

d) Retrieve the numbers of students, student id and student name from table members where std_major is Software Engineering.

```
SELECT COUNT(*),std_id,std_name
FROM Members
WHERE std_major='Software Engineering';
```

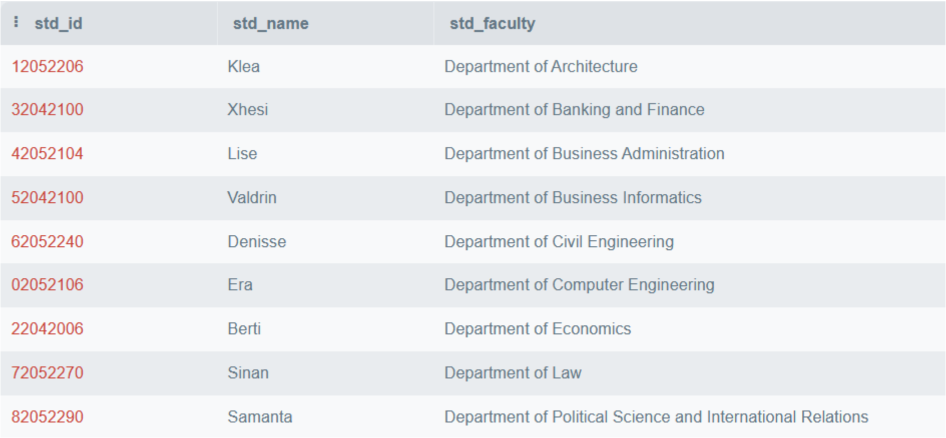
7. ***GroupBy***

a) For each club, retrieve the club_id, place, and the number of times each club will hold its activity in a place.

```
SELECT club_id, place, COUNT(*)
FROM Activities
GROUP BY club_id, place;
```

b) For each member, retrieve the std_id, std_name and std_faculty, whose std_id is greater than 006 supplied by the std_faculty

```
Select std_id,std_name,std_faculty
FROM Members
Where std_id>'006'
GROUP BY std_faculty;
```



std_id	std_name	std_faculty
12052206	Klea	Department of Architecture
32042100	Xhesi	Department of Banking and Finance
42052104	Lise	Department of Business Administration
52042100	Valdrin	Department of Business Informatics
62052240	Denisse	Department of Civil Engineering
02052106	Era	Department of Computer Engineering
22042006	Berti	Department of Economics
72052270	Sinan	Department of Law
82052290	Samanta	Department of Political Science and International Relations

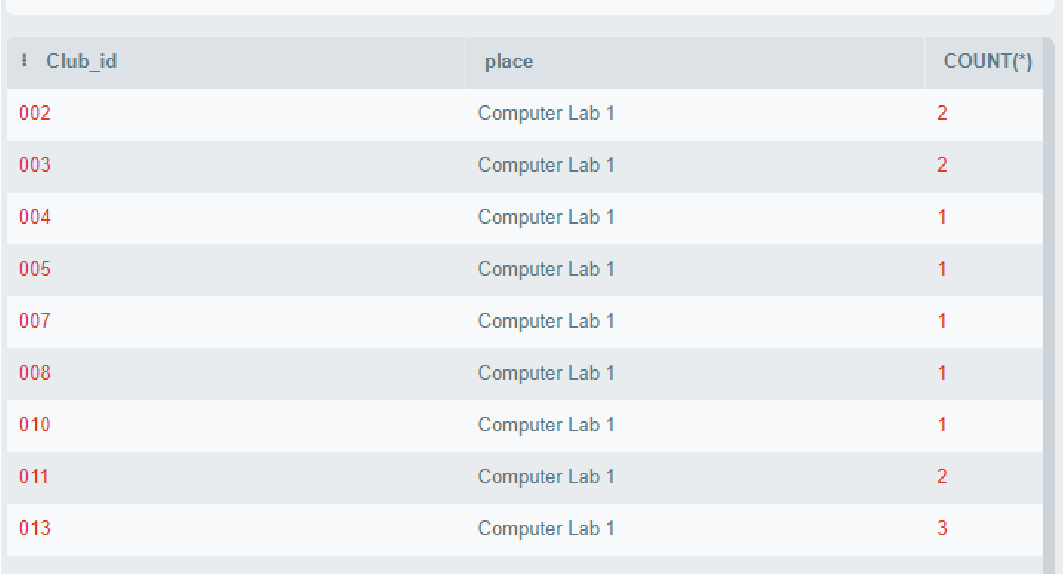
c) Retrieve certificate_id,certificate_res,std_id and signed_by of the 15 first certificates.

```
SELECT certificate_id,certificate_res,std_id,signed_by
FROM Certificate
WHERE certificate_id<= 9915
GROUP BY certificate_id;
```

8. ***Having***

a) For each club, retrieve the club_id, place, and the number of times each of the clubs will hold its activity in the 'Computer Lab 1'.

```
SELECT club_id, place, COUNT(*)
FROM Activities
GROUP BY club_id, place
HAVING place LIKE '%Computer Lab 1%';
```



Club_id	place	COUNT(*)
002	Computer Lab 1	2
003	Computer Lab 1	2
004	Computer Lab 1	1
005	Computer Lab 1	1
007	Computer Lab 1	1
008	Computer Lab 1	1
010	Computer Lab 1	1
011	Computer Lab 1	2
013	Computer Lab 1	3

b) For each activity_name that has been shown more than 4 times, retrieve student_id, number of all contents of part_of, grouped by activity name

```
Select std_id, COUNT(*),activity_name
FROM part_of
GROUP BY activity_name
HAVING COUNT(*)>4;
```

c) Retrieve acticity_name,club_id, capacity of the place where activity is held and capacity of the place is more or equal to 60.

```
SELECT activity_name,club_id,capacity
FROM Activities
Group by place='Computer Lab 2'
HAVING capacity>=60;
```

e) For each person whose name starts with letter A and is in the table ClubStaff and StaffRole retrieve name and surname

```
 select Pers_name, Pers_surname
from ClubStaff ,StaffRole
GROUP by Pers_name
HAVING Pers_name LIKE 'A%';
```

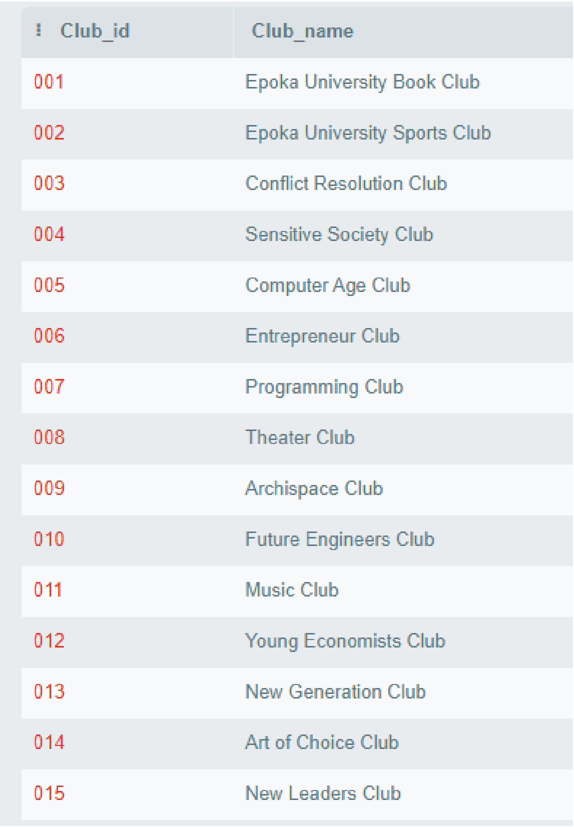
9. ***Nested Queries***

a) For each activity whose capacity is below the average capacity of all activities retrieve the activity_name, guest_name, place, and capacity.

```
SELECT activity_name, guest_name, place, capacity
FROM Activities
WHERE capacity < (SELECT AVG (capacity) FROM Activities);
```

b) For each club whose club_id is below the average of club_id of all clubs, retrieve club_id and club_name

```
Select club_id,club_name
from clubs
WHERE club_id<(select AVG(club_id) FROM Clubs);
```



Club_id	Club_name
001	Epoka University Book Club
002	Epoka University Sports Club
003	Conflict Resolution Club
004	Sensitive Society Club
005	Computer Age Club
006	Entrepreneur Club
007	Programming Club
008	Theater Club
009	Archispace Club
010	Future Engineers Club
011	Music Club
012	Young Economists Club
013	New Generation Club
014	Art of Choice Club
015	New Leaders Club

c) --For each club capacity whose capacity is equal to the avg of all clubs capacity, retrieve club name and the id.

```
Select club_name ,club_id
FROM Clubs
WHERE club_capacity =(SELECT  AVG(club_capacity) FROM Clubs);
```

10. ***Join***

a) For each activity, retrieve the club_id, place and the name of each club that will hold its activity in the 'Computer Lab 1'.

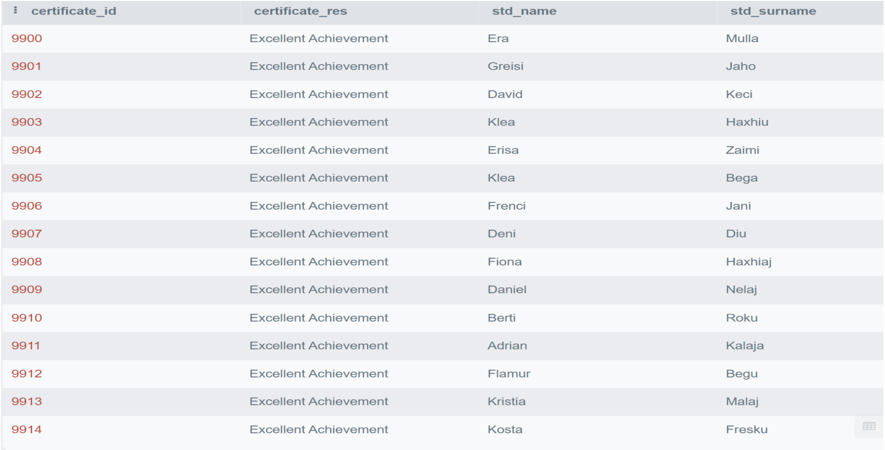
```
SELECT Activities.Club_id, Clubs.Club_name, Activities.place, COUNT(*) as counting
FROM Activities
INNER JOIN Clubs ON Activities.club_id = Clubs.Club_id
GROUP BY Activities.club_id, place
HAVING place LIKE '%Computer Lab 1%';
```

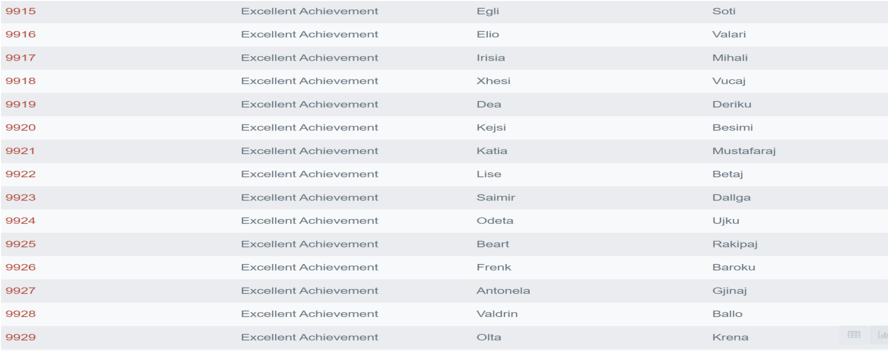
b) For each club, retrieve club_name, created_date,club_capacity, and pers name.

```
Select club_name,created_date,club_capacity, Pers_name
from Clubs, ClubStaff
WHERE Clubs.Club_id=ClubStaff.Club_id;
```

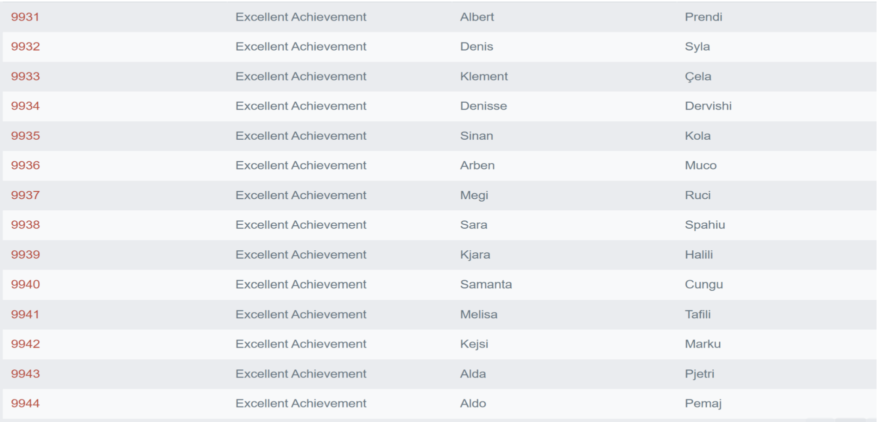
c) For each member retrieve member name and surname, certificate id, and the results

```
SELECT  certificate_id,certificate_res,std_name,std_surname
FROM Certificate, Members
WHERE Certificate.std_id=Members.std_id;
```





certificate_id	certificate_res	std_name	std_surname
9900	Excellent Achievement	Era	Mulla
9901	Excellent Achievement	Greisi	Jaho
9902	Excellent Achievement	David	Keci
9903	Excellent Achievement	Klea	Haxhiu
9904	Excellent Achievement	Erisa	Zaimi
9905	Excellent Achievement	Klea	Bega
9906	Excellent Achievement	Frenci	Jani
9907	Excellent Achievement	Deni	Diu
9908	Excellent Achievement	Fiona	Haxhiaj
9909	Excellent Achievement	Daniel	Nelaj
9910	Excellent Achievement	Berti	Roku
9911	Excellent Achievement	Adrian	Kalaja
9912	Excellent Achievement	Flamur	Begu
9913	Excellent Achievement	Kristia	Malaj
9914	Excellent Achievement	Kosta	Fresku
9915	Excellent Achievement	Egli	Soti
9916	Excellent Achievement	Elio	Valari
9917	Excellent Achievement	Irisia	Mihali
9918	Excellent Achievement	Xhesi	Vucaj
9919	Excellent Achievement	Dea	Deriku
9920	Excellent Achievement	Kejsi	Besimi
9921	Excellent Achievement	Katia	Mustafaraj
9922	Excellent Achievement	Lise	Betaj
9923	Excellent Achievement	Saimir	Dallga
9924	Excellent Achievement	Odeta	Ujku
9925	Excellent Achievement	Beart	Rakipaj
9926	Excellent Achievement	Frenk	Baroku
9927	Excellent Achievement	Antonela	Gjinaj
9928	Excellent Achievement	Valdrin	Ballo
9929	Excellent Achievement	Olta	Krena



9931	Excellent Achievement	Albert	Prendi
9932	Excellent Achievement	Denis	Syla
9933	Excellent Achievement	Klement	Çela
9934	Excellent Achievement	Denisse	Dervishi
9935	Excellent Achievement	Sinan	Kola
9936	Excellent Achievement	Arben	Muco
9937	Excellent Achievement	Megi	Ruci
9938	Excellent Achievement	Sara	Spahiu
9939	Excellent Achievement	Kjara	Halili
9940	Excellent Achievement	Samanta	Cungu
9941	Excellent Achievement	Melisa	Tafili
9942	Excellent Achievement	Kejsi	Marku
9943	Excellent Achievement	Alda	Pjetri
9944	Excellent Achievement	Aldo	Pemaj

d) Retrieve club_name and clubstaff_id from tables clubs and clubstaff where club_id is the same in both tables.

```
SELECT club_name,clubstaff_id
FROM Clubs, ClubStaff
WHERE Clubs.club_id=ClubStaff.club_id;
```

e) Retrieve std_name,std_surname, and std_email from tables Certificate and Members where std id is the same.

```
SELECT std_name,std_surname,std_email
 FROM Certificate, Members
WHERE Certificate.std_id=Members.std_id;
```

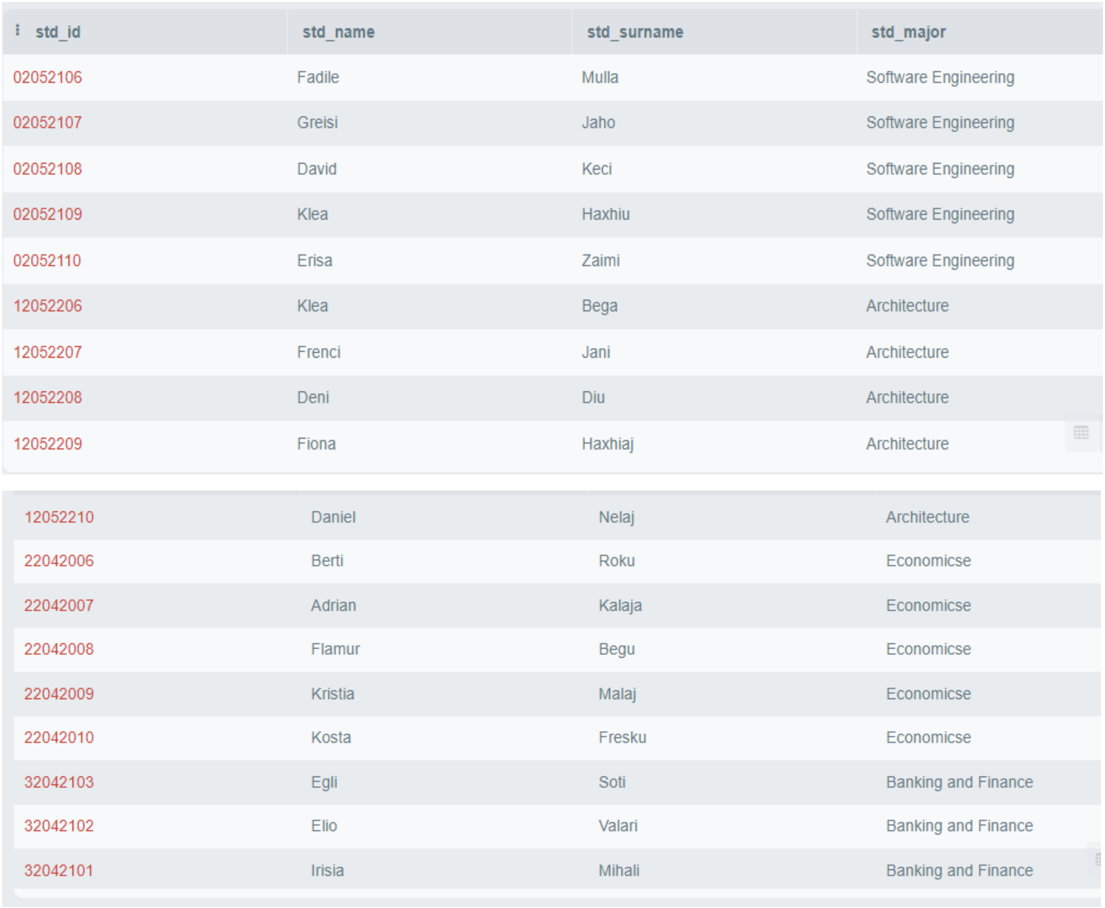
11. ***Alias***

a) For each activity, retrieve the club_id, place, and the name of each club that will hold its activity in the 'Computer Lab 1'.

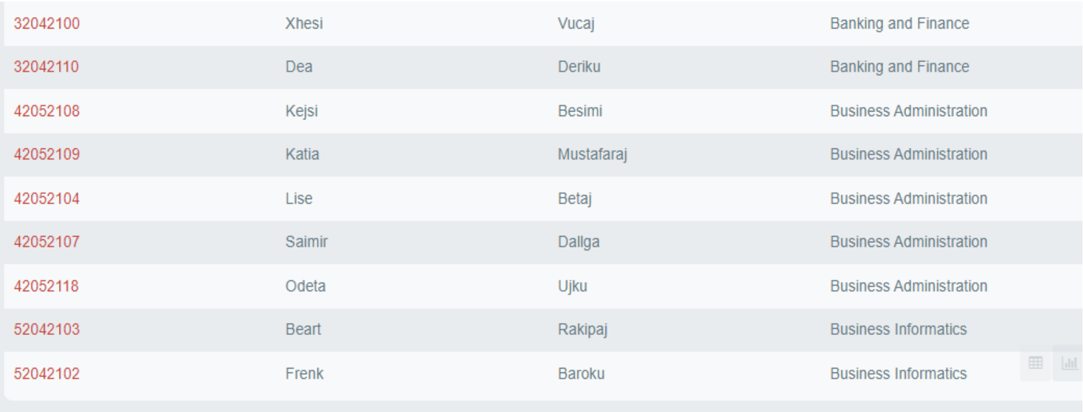
```
SELECT a.Club_id, c.Club_name, a.place, COUNT(*) as counting
FROM Activities a
INNER JOIN Clubs c ON a.club_id = c.Club_id
GROUP BY a.club_id, place
HAVING place LIKE '%Computer Lab 1%';
```

b) For each club, retrieve club_name, created_date, and activity name.
Select c.club_name,c.created_date,a.activity_name
from Clubs c, Activities a
WHERE c.Club_id=a.Club_id;

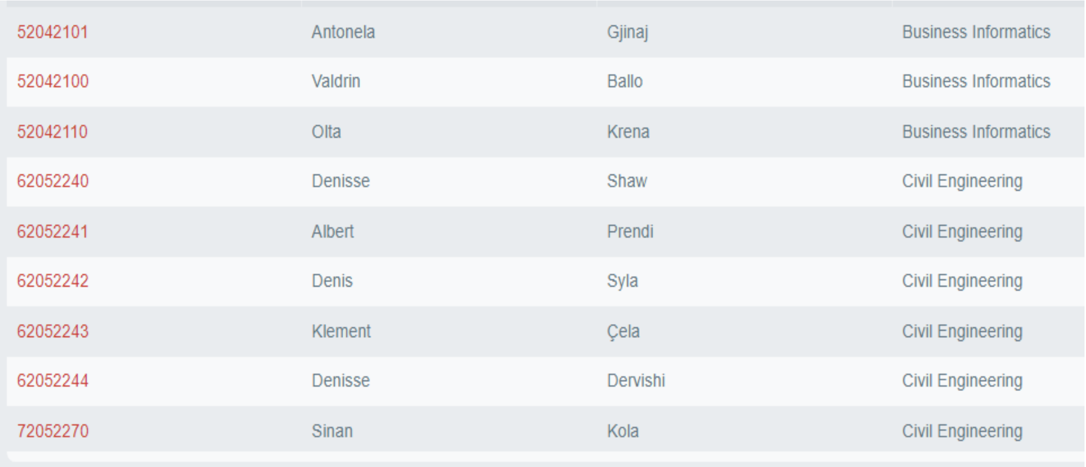
c) Retrieve student id, std name, std surname, and major from tables members and certificate.
SELECT m.std_id,m.std_name,m.std_surname,m.std_major
FROM Members m, Certificate c
WHERE m.std_id=c.std_id;



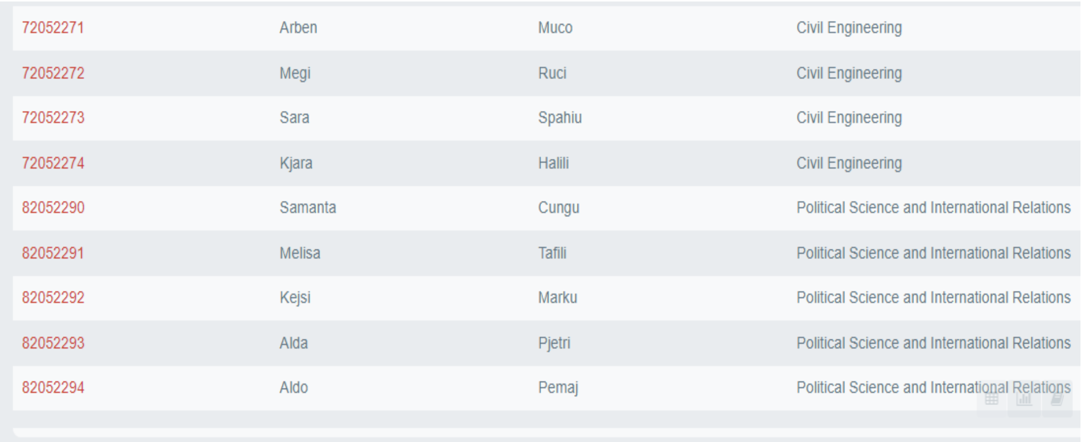
std_id	std_name	std_surname	std_major
02052106	Fadile	Mulla	Software Engineering
02052107	Greisi	Jaho	Software Engineering
02052108	David	Keci	Software Engineering
02052109	Klea	Haxhiu	Software Engineering
02052110	Erisa	Zaimi	Software Engineering
12052206	Klea	Bega	Architecture
12052207	Frenci	Jani	Architecture
12052208	Deni	Diu	Architecture
12052209	Fiona	Haxhiaj	Architecture
12052210	Daniel	Nelaj	Architecture
22042006	Berti	Roku	Economicse
22042007	Adrian	Kalaja	Economicse
22042008	Flamur	Begu	Economicse
22042009	Kristia	Malaj	Economicse
22042010	Kosta	Fresku	Economicse
32042103	Egli	Soti	Banking and Finance
32042102	Elio	Valari	Banking and Finance
32042101	Irisia	Mihali	Banking and Finance



32042100	Xhesi	Vucaj	Banking and Finance
32042110	Dea	Deriku	Banking and Finance
42052108	Kejsi	Besimi	Business Administration
42052109	Katia	Mustafaraj	Business Administration
42052104	Lise	Betaj	Business Administration
42052107	Saimir	Dallga	Business Administration
42052118	Odeta	Ujku	Business Administration
52042103	Beart	Rakipaj	Business Informatics
52042102	Frenk	Baroku	Business Informatics



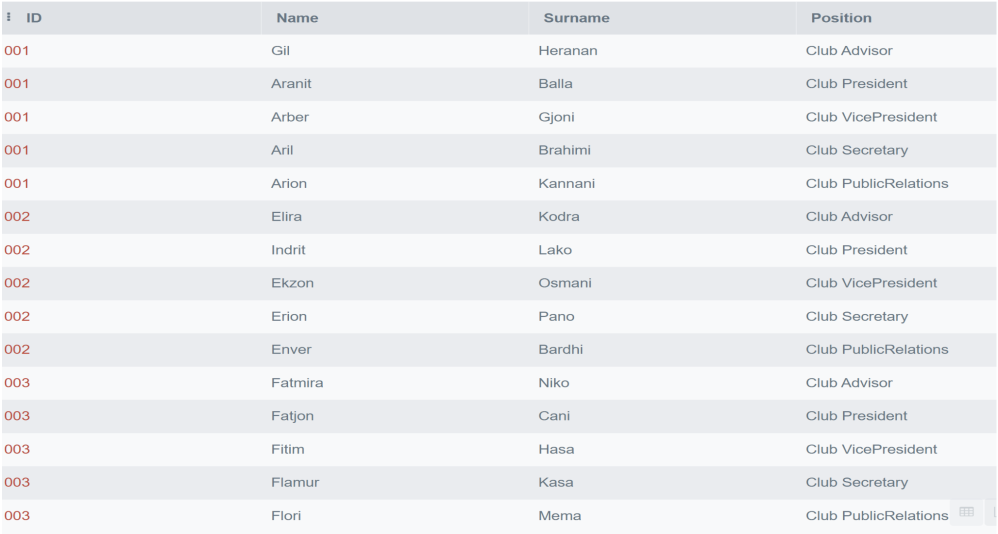
52042101	Antonela	Gjinaj	Business Informatics
52042100	Valdrin	Ballo	Business Informatics
52042110	Olta	Krena	Business Informatics
62052240	Denisse	Shaw	Civil Engineering
62052241	Albert	Prendi	Civil Engineering
62052242	Denis	Syla	Civil Engineering
62052243	Klement	Çela	Civil Engineering
62052244	Denisse	Dervishi	Civil Engineering
72052270	Sinan	Kola	Civil Engineering



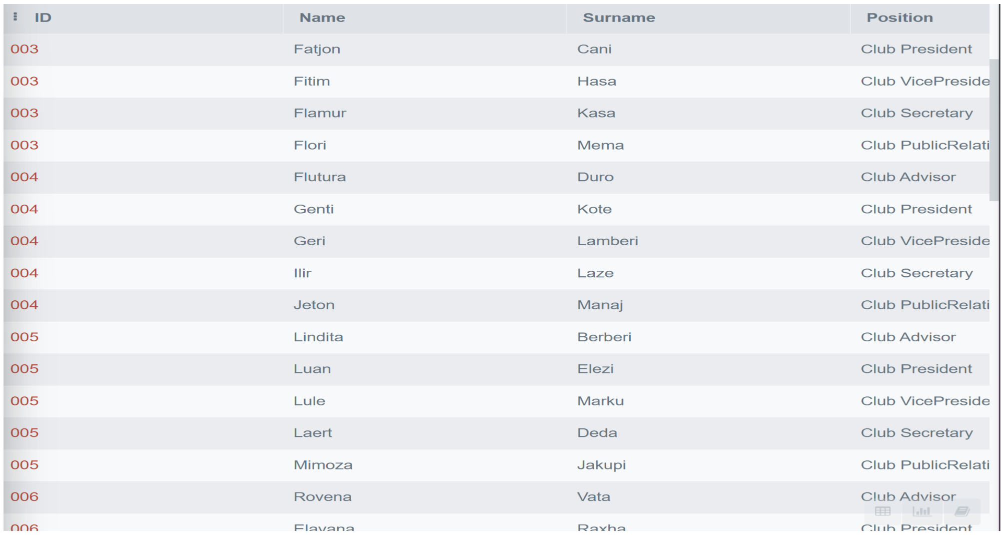
72052271	Arben	Muco	Civil Engineering
72052272	Megi	Ruci	Civil Engineering
72052273	Sara	Spahiu	Civil Engineering
72052274	Kjara	Halili	Civil Engineering
82052290	Samanta	Cungu	Political Science and International Relations
82052291	Melisa	Tafili	Political Science and International Relations
82052292	Kejsi	Marku	Political Science and International Relations
82052293	Alda	Pjetri	Political Science and International Relations
82052294	Aldo	Pemaj	Political Science and International Relations

d) Retrieve club id, name, surname of the staff members and their roles.

```
SELECT  c.club_id AS ID,c.Pers_name AS Name,c.Pers_surname AS Surname,s.role_name As
Position
FROM ClubStaff c, StaffRole s
WHERE c.role_id=s.role_id
```

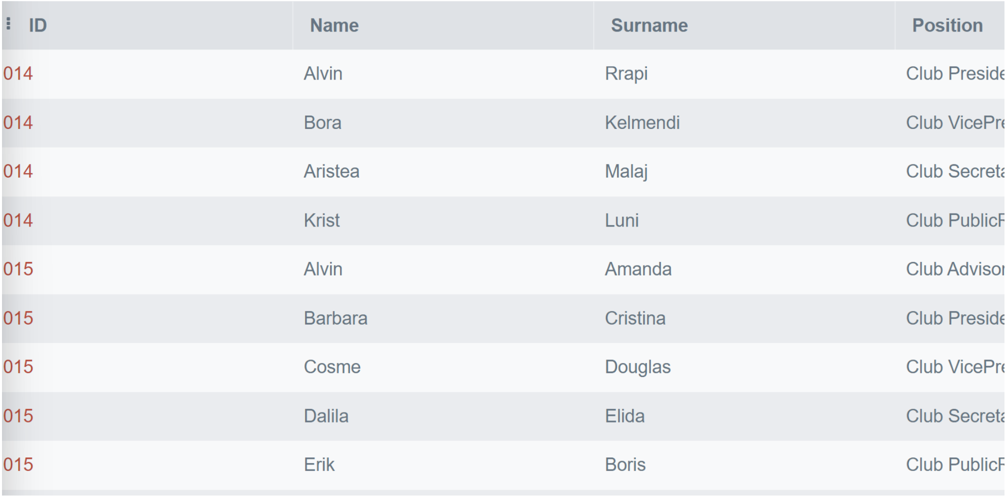


ID	Name	Surname	Position
001	Gil	Heranan	Club Advisor
001	Aranit	Balla	Club President
001	Arber	Gjoni	Club VicePresident
001	Aril	Brahimi	Club Secretary
001	Arion	Kannani	Club PublicRelations
002	Elira	Kodra	Club Advisor
002	Indrit	Lako	Club President
002	Ekzon	Osmani	Club VicePresident
002	Erion	Pano	Club Secretary
002	Enver	Bardhi	Club PublicRelations
003	Fatmira	Niko	Club Advisor
003	Fatjon	Cani	Club President
003	Fitim	Hasa	Club VicePresident
003	Flamur	Kasa	Club Secretary
003	Flori	Mema	Club PublicRelations



ID	Name	Surname	Position
003	Fatjon	Cani	Club President
003	Fitim	Hasa	Club VicePreside
003	Flamur	Kasa	Club Secretary
003	Flori	Mema	Club PublicRelati
004	Flutura	Duro	Club Advisor
004	Genti	Kote	Club President
004	Geri	Lamberi	Club VicePreside
004	Ilir	Laze	Club Secretary
004	Jeton	Manaj	Club PublicRelati
005	Lindita	Berberi	Club Advisor
005	Luan	Elezi	Club President
005	Lule	Marku	Club VicePreside
005	Laert	Deda	Club Secretary
005	Mimoza	Jakupi	Club PublicRelati
006	Rovena	Vata	Club Advisor
006	Flavana	Raxha	Club President

. . .



ID	Name	Surname	Position
014	Alvin	Rrapi	Club Preside
014	Bora	Kelmendi	Club VicePre
014	Aristea	Malaj	Club Secreta
014	Krist	Luni	Club PublicF
015	Alvin	Amanda	Club Advisor
015	Barbara	Cristina	Club Preside
015	Cosme	Douglas	Club VicePre
015	Dalila	Elida	Club Secreta
015	Erik	Boris	Club PublicF

12. ***Alter Table***

a)
```
ALTER TABLE ClubStaff ADD year CHAR (5);
ALTER TABLE ClubStaff DROP year;
```

b) Add std_name in table Certificate
```
ALTER TABLE Certificate ADD std_name char(15);
```

c)Add a column about the certificate name to the relation table Certificate and after drop it
```
Alter Table Certificate ADD
certificate_Name Char(15);
```

d)
```
Alter Table Certificate DROP
certificate_Name;
```

e)
```
ALTER TABLE Members ADD studentAge CHAR(11);
```

13. ***Update***

a)
```
INSERT INTO StaffRole VALUES ('5636','Advisor');
UPDATE StaffRole
SET role_name = 'Club Advisor'
WHERE role_id = 5636;
```

b) Update guest name into "Dan Brown", where the place is E110 and club_id is 001.
```
UPDATE Activities
SET guest_name='Dan Brown'
Where place='E110' AND club_id='001';
```

c) Update capacity to 50 where the club name is Programming Club
```
UPDATE Clubs
SET club_capacity= 50
Where club_name Like "%Programming Club%";
```

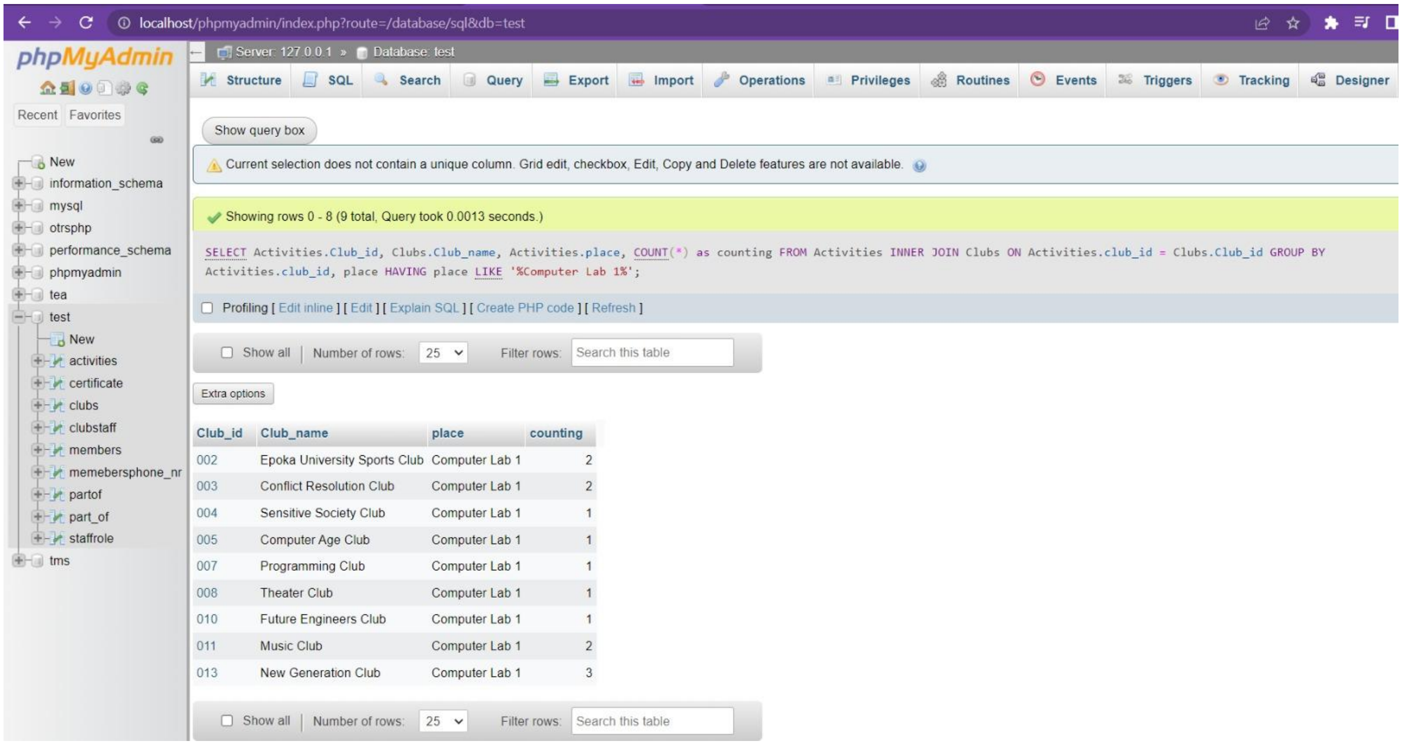
d) Update member phone number where student id is 52042103
```
UPDATE MemebersPhone_nr
SET phone_nr='0687765123'
WHERE std_id='52042103';
```

e) Update name where id is equal to 02052106
UPDATE Members
SET std_name = 'Fadile'
WHERE std_id = '02052106';

14. ***Delete***

a) Delete statement to delete from table activities all values where clubid is 001.
DELETE FROM Activities
WHERE Club_id='001';

7. Bonus Question



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 8 (9 total, Query took 0.0013 seconds.)

```
SELECT Activities.Club_id, Clubs.Club_name, Activities.place, COUNT(*) as counting FROM Activities INNER JOIN Clubs ON Activities.club_id = Clubs.Club_id GROUP BY
Activities.club_id, place HAVING place LIKE '%Computer Lab 1%';
```

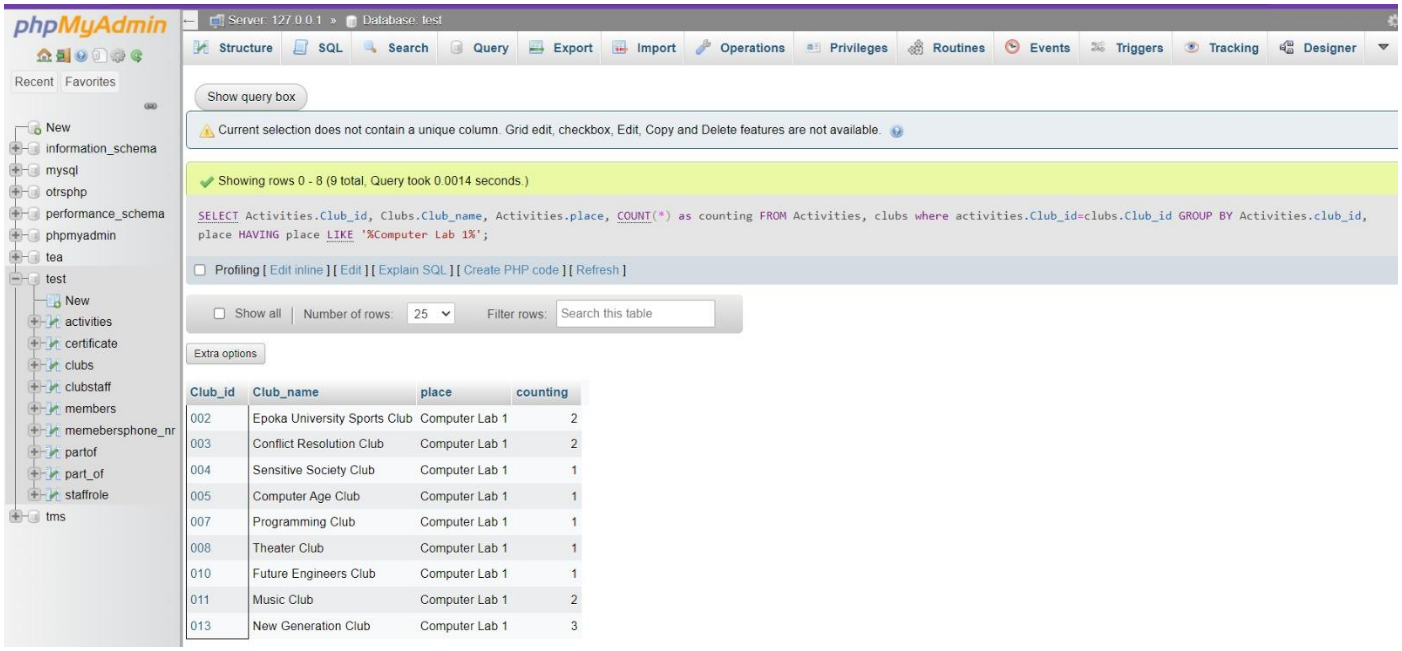
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

Club_id	Club_name	place	counting
002	Epoka University Sports Club	Computer Lab 1	2
003	Conflict Resolution Club	Computer Lab 1	2
004	Sensitive Society Club	Computer Lab 1	1
005	Computer Age Club	Computer Lab 1	1
007	Programming Club	Computer Lab 1	1
008	Theater Club	Computer Lab 1	1
010	Future Engineers Club	Computer Lab 1	1
011	Music Club	Computer Lab 1	2
013	New Generation Club	Computer Lab 1	3

□ Show all | Number of rows: 25 Filter rows: Search this table

This is a query where is INNER JOIN is implemented. The speed for this query o be executed is 0.0013 sec.



⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 8 (9 total, Query took 0.0014 seconds.)

```
SELECT Activities.Club_id, Clubs.Club_name, Activities.place, COUNT(*) as counting FROM Activities, clubs where activities.Club_id=clubs.Club_id GROUP BY Activities.club_id,
place HAVING place LIKE '%Computer Lab 1%';
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
□ Show all | Number of rows: 25
Filter rows: Search this table
Extra options

Club_id	Club_name	place	counting
002	Epoka University Sports Club	Computer Lab 1	2
003	Conflict Resolution Club	Computer Lab 1	2
004	Sensitive Society Club	Computer Lab 1	1
005	Computer Age Club	Computer Lab 1	1
007	Programming Club	Computer Lab 1	1
008	Theater Club	Computer Lab 1	1
010	Future Engineers Club	Computer Lab 1	1
011	Music Club	Computer Lab 1	2
013	New Generation Club	Computer Lab 1	3

This is a query where "WHERE" is implemented and the speed for this query to be executed is 0.0014 sec.

NOTE:

We want to thank each member of our group for their valuable contribution and effort that they have put in this project. We want to thank our client for believing in us with it, and we are thankful to help improving our university's systems.