Database and Web Services Project Assignment 4

Ilyas Benyamna, Ujjwal Khadka, Saad Aboujid, Idris Chendid

Instructions

To get the following results, we first had to create the tables, to see how tables were created see `TABLE_CREATION.sql`.
Afterwards we had to populate the tables, we did that with `INSERT` queries, you can see those queries in `INSERTIONS.sql`.
For testing we populated a local database, we also populated the CLAMV sql database called `group4`.
To see the SELECT queries we came up with see file `SELECTs.sql`. The query results were obtained using PopSQL.

Query Execution Logs

This query finds what assigned tasks are in which projects and to which employee as what role:

```
SELECT project_name, task_name, username, role_name FROM User
INNER JOIN Employee
      ON User.uid = Employee.uid
INNER JOIN assigned
      ON assigned.eid = Employee.eid
INNER JOIN Role
      ON assigned.rid = Role.rid
INNER JOIN Task
      ON Task.task_id = assigned.task_id
INNER JOIN Project
      ON Project.pid = Task.project_id;
```

project_name	task_name	username	role_name
project 1	task 4	hAkka	IT
project 1	task 3	JDoe	Salesman
project 1	task 2	hhmidouch	Media Buyer
project 1	task 1	JKumar	IT
project 2	task 4	hhmidouch	Media Buyer
project 2	task 3	JKumar	IT
project 2	task 2	ASharma	IT
project 2	task 1	ADhamma	IT

Finding the username of employees who have a task along with the project they work in, also includes those who

are not assigned a task:

```
SELECT username, project_name, task_name, role_name FROM User
LEFT JOIN Employee
      ON User.uid = Employee.uid
LEFT JOIN assigned
      ON assigned.eid = Employee.eid
LEFT JOIN Role
      ON assigned.rid = Role.rid
LEFT JOIN Task
      ON Task.task_id = assigned.task_id
LEFT JOIN Project
      ON Project.pid = Task.project_id;
```

username	project_name	task_name	role_name
ADhamma	project 2	task 1	IT
ASharma	project 2	task 2	IT
hAkka	project 1	task 4	IT
hhmidouch	project 1	task 2	Media Buyer
hhmidouch	project 2	task 4	Media Buyer
JDoe	project 1	task 3	Salesman
JKumar	project 1	task 1	IT
JKumar	project 2	task 3	IT
pdriver	NULL	NULL	NULL
pparker	NULL	NULL	NULL
soloman	NULL	NULL	NULL

Finding the names of tasks that have not been assigned yet:

```
SELECT task_name FROM Task
LEFT JOIN assigned
     ON Task.task_id = assigned.task_id
WHERE assigned.task_id IS NULL;
```

task_name
task 5 (unassigned)

Counting how many tasks are in each project:

```
SELECT project_name, COUNT(*) AS number_of_tasks
FROM Project
INNER JOIN Task
```

```
      ON Project.pid = Task.project_id
GROUP BY Project.pid;
```

project_name	number_of_tasks
project 1	4
project 2	5

Counting how many tasks are assigned to each employee

```
SELECT username, COUNT(assigned.task_id) AS number_of_tasks
FROM User
LEFT JOIN Employee
      ON User.uid = Employee.uid
LEFT JOIN assigned
      ON assigned.eid = Employee.eid
GROUP BY User.uid;
```

username	number_of_tasks
JKumar	2
hhmidouch	2
JDoe	1
hAkka	1
ADhamma	1
ASharma	1
pparker	0
pdriver	0
soloman	0

Each team member's username and his role in the team

```
SELECT username, team_name,role_name FROM User
INNER JOIN Employee
      ON User.uid = Employee.uid
INNER JOIN Team_Member
      ON Team_Member.eid = Employee.eid
INNER JOIN Role
      ON Role.rid = Team_Member.rid
INNER JOIN Team
      ON Team.tid = Team_Member.tid;
```

.
.

username	team_name	role_name
JKumar	Team 1	IT
hhmidouch	Team 1	Media Buyer
JDoe	Team 1	Salesman
hAkka	Team 2	IT
ADhamma	Team 2	IT
ASharma	Team 2	IT

Table with User's username and his team and his role, also shows employees not in a team and not in a role:

```
SELECT username, role_name, team_name FROM User
LEFT JOIN Employee
      ON User.uid = Employee.uid
LEFT JOIN Team_Member
      ON Team_Member.eid = Employee.eid
LEFT JOIN Role
      ON Role.rid = Team_Member.rid
LEFT JOIN Team
      ON Team.tid = Team_Member.tid;
```

username	role_name	team_name
ADhamma	IT	Team 2
ASharma	IT	Team 2
hAkka	IT	Team 2
hhmidouch	Media Buyer	Team 1
JDoe	Salesman	Team 1
JKumar	IT	Team 1
pdriver	NULL	NULL
pparker	NULL	NULL
soloman	NULL	NULL

How many members in each role? There is also another query that can be used to select how many members are in a specific team.

```
    SELECT role_name, COUNT(*) AS number_of_members  FROM Team_Member
INNER JOIN Role
      ON Role.rid = Team_Member.rid
GROUP BY Role.rid;
```

```
SELECT role_name, COUNT(*) AS number_of_members  FROM Team_Member
INNER JOIN Role
      ON Role.rid = Team_Member.rid
GROUP BY Role.rid
HAVING Role.rid=1;
```

role_name	number_of_members
IT	4
Media Buyer	1
Salesman	1

role_name	number_of_members
IT	4

How many members are in each team?

```
SELECT team_name, COUNT(*) AS number_of_members
FROM Team_Member
INNER JOIN Team
      ON Team_Member.tid = Team.tid
GROUP BY team_name;
```

team_name	number_of_members
Team 1	3
Team 2	3

Each project and who manages it:

```
SELECT project_name, username FROM Project
INNER JOIN  Manager
      ON Manager.pid = Project.pid
INNER JOIN User
      ON User.uid = Manager.uid;
```

project_name	username
project 1	pparker
project 2	pdriver

Which third parties participate in each project?

```
SELECT company_name, project_name, client, partner
FROM Third_Party
INNER JOIN helps_or_requests
      ON Third_Party.tpid = helps_or_requests.tpid
INNER JOIN Project
      ON Project.pid = helps_or_requests.pid;
```

company_name	project_name	client	partner
facebook	project 1	0	1
microsoft	project 2	1	0
google	project 2	1	1

How many third parties in each project?

```
SELECT project_name,
        COUNT(client IS TRUE) AS number_of_clients,
        COUNT(partner IS TRUE) AS number_of_partners
FROM Third_Party
INNER JOIN helps_or_requests
      ON Third_Party.tpid = helps_or_requests.tpid
INNER JOIN Project
      ON Project.pid = helps_or_requests.pid
GROUP BY Project.pid;
```

project_name	number_of_clients	number_of_partners
project 1	1	1
project 2	2	2

The duration of each project:

```
SELECT project_name, DATEDIFF(end_date, start_date) AS project_duration_in_days
FROM Project;
```

project_name	project_duration_in_days
project 1	31
project 2	134

The duration of each task:

```
SELECT project_name,task_name, DATEDIFF(Task.end_date, Task.start_date) AS task_duration_i
FROM Task
INNER JOIN Project
      ON Project.pid = Task.project_id;
```

project_name	task_name	task_duration_in_days
project 1	task 1	2
project 1	task 2	23
project 1	task 3	17
project 1	task 4	30
project 2	task 1	63
project 2	task 2	35
project 2	task 3	95
project 2	task 4	436
project 2	task 5 (unassigned)	436

The sum of the tasks' duration in each project:

```
SELECT project_name,SUM(DATEDIFF(Task.end_date, Task.start_date)) AS duration_of_tasks_in_p
INNER JOIN Project
      ON Project.pid = Task.project_id
GROUP BY Project.pid;
```

project_name	duration_of_tasks_in_project
project 1	72
project 2	1065