Big Apple: An Office Networking Platform

Team Members:

Jay Jayesh Shah: jjs816
Anshul Garg: ag6516

Big Apple is an application that allows all the employee to connect and update each other with their existing projects, form teams to contribute to projects, update their skills and interest in their profiles, add certifications, share pictures and location of the conferences, events and socialize with other employee across the firm.

Following are the functionalities of the system:

Create/Register a Big Apple profile:
All employees of the organization are allowed to create a profile on Big Apple. The employee can make the profile public or private upon his discretion. Once the employee deletes his profile, all the data corresponding to the employee profile will be deleted.

Create or Update Posts on the Wall:
Employees can create and update posts on their wall. The posts can be anything specific to the employee. Examples of which can be all the courses that he is takes, his views on modern technologies, hotlinks to his github repository, twitter handle etc. He can also create post specific to the project and provide his team explicitly to view the content.

Network and build relationships:
Big Apple allows an employee to network and collaborate with other employees by sending a connect request. An employee can also use Big Apple to form teams and collaborate with the team members in project activities. Connection can be team, friends or distinct friends given the scenario.

Upload Multimedia Contents:
Employees can post pictures, videos etc of presentation, conferences, training resources with Big Apple providing the provision to either share it across team or across the organization.

Adding Location details and Comments :
Employees can post their location details on Big Apple that allows others to know where you are. This feature can be used when the employee attends a event and would like to showcase his presence across the organizations. Big Apple allows an employee to comment on post and also allows him to make the comment restricted to Team or Public .

Likes:
Employees can like posts, comment or multimedia contents posted by an employee.

Database Requirements:

Big Apple will be designed as a relational database application which will be using MySQL as its database server.

MySQL version:

Variable_name	Value
version_compile_machine	x86_64
tls_version	TLSv1,TLSv1.1
▶ version_comment	MySQL Community Server (GPL)
version_compile_os	macos10.12
innodb_version	5.7.19
version	5.7.19
protocol_version	10

Entity Relationship Diagram:
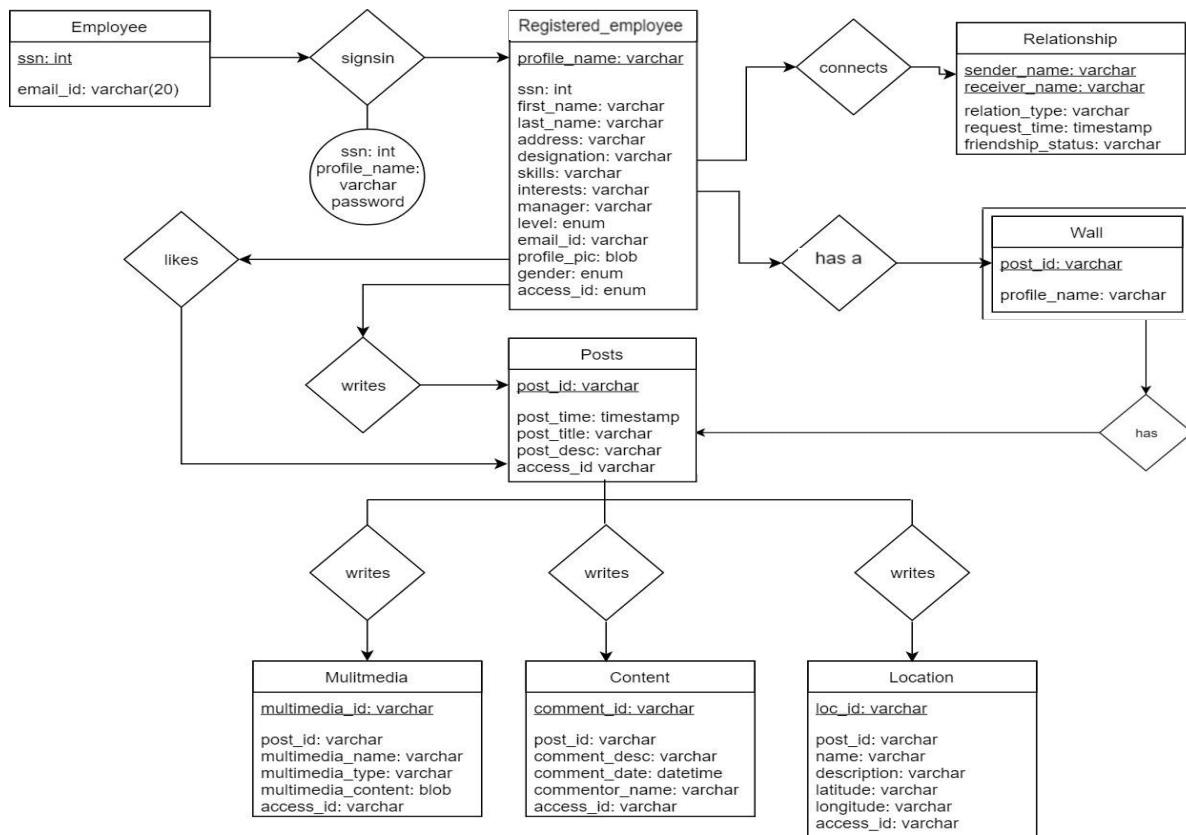
Assumptions:

- Every employee is assigned a level between A to D where A refers to Junior associates, B refers to senior associates, C refers to managers, D refers to CEO, President, Trustee. By default a new employee is assigned the level A following which the HR assigns the corresponding level based on their skills and experience.
- Access id are allowed at the profile and post level.
- Access id's can be S: Self, T: Team, F: Friends, FOF : Friend and Friends of Friends, P: Public.
- Relation_type in the relationship table indicates the type of friendship, F: Friends, FOF: Friend of Friends, T: Team.

Relational Schema:

- employee(**ssn,**email_id)
- signin(ssn,password)
- registered_employee(**profile_name**,ssn,first_name, last_name, address, designation, skils, interests, manager, level,access_id,email_id, profile_pic,gender)
- post(**post_id**, post_date, post_title, post_desc,access_id)
- wall(post_id, profile_name)
- relationship(sender_name, receiver_name, relation_type, request_time, request_status)
- multimedia_content(**multimedia_id**,post_id,multimedia_name,multimedia_type, multimedia_content,access_id)
- comment(**comment_id**,post_id,desc,comment_date,commentor_name,access_id)
- location(**loc_id,**post_id,name,description,latitude,longitude,access_id)
- likes(**like_id**,post_id,comment_id,multimedia_id,loc_id,viewer_name,like_date)

Foreign Keys:

- ssn in signin table is a foreign key referencing the ssn in the employee table.
- ssn in the registered_employee table is a foreign key referencing the ssn in employee table.
- post_id,profile_name in wall table references the post_id in post and profile_name in the registered_employee table respectively.
- In multimedia_connent table post_id is the foreign key referencing table post.
- In comment table post_id is the foreign key referencing table post and commentor_name is the foreign key referencing table registered_employee.
- In like table likes viewer_name is the foreign key referencing table registered_employee and post_id is the foreign key referencing table post and multimedia_id is the foreign key referencing table multimedia_content and loc_id is the foreign key referencing table location.
- In relationship table sender_name references table registered_employee.

```
create database BigApple;
use BigApple;

drop table if exists employee;

#Create employee table
drop table if exists employee;
create table employee(
ssn int primary key, email_id varchar(20) not null
);

#Create sign in table
drop table if exists signin;
create table signin
(
ssn int unique,
password varchar(50) not null,
profile_name varchar(30) primary key,
foreign key (ssn) References employee(ssn)
on delete cascade

);

#Create registered_employee  table
drop table if exists registered_employee;
SET FOREIGN_KEY_CHECKS=1;
create table registered_employee
(
profile_name varchar(30) primary key,
first_name varchar(20) not null ,
last_name varchar(20) not null,
address varchar (255),
designation varchar(20),
skills varchar(255),
interests varchar(255),
manager varchar(20),
level enum('A','B','C','D') default 'A',
email_id varchar(50),
profile_pic blob,
gender enum('Male','Female'),
access_id enum('P','T','F','S','FOF') default 'S',
foreign key (profile_name) References signin(profile_name) );
```

#Create post

```sql
drop table if exists post;
create table post
(
post_id varchar(30),
post_time timestamp not null,
post_title varchar(150),
post_desc varchar(255),
access_id varchar(5) default 'S',
primary key (post_id)
);
```

#Create wall(post_id,profile_name)

```sql
drop table if exists wall;
create table wall
(
post_id varchar(20),
profile_name varchar(30),
foreign key (profile_name) references registered_employee(profile_name)
on delete cascade
on update cascade,
foreign key (post_id) references post(post_id)
on delete cascade
on update cascade
);
```

#Create relationship table

```sql
drop table if exists relationship;
create table relationship
(
sender_name varchar(30),
receiver_name varchar(30),
relation_type varchar(20),
request_time timestamp,
friendship_status varchar(20) NOT NULL,

foreign key (sender_name) references registered_employee(profile_name)
on delete cascade
on update cascade,
foreign key (receiver_name) references registered_employee(profile_name)
```

```
on delete cascade
on update cascade,

primary key(sender_name,receiver_name)
 );


#Create multimedia table

drop table if exists multimedia_content;
create table multimedia_content
(
multimedia_id varchar(10) primary key,
post_id varchar(20),
multimedia_name varchar(20),
Multimedia_type varchar(20),
multimedia_content blob,
access_id varchar(5) default 'S',
foreign key (post_id) references post(post_id)
on delete cascade
on update cascade
);


#Create table Comment
drop table if exists comment;
create table comment
(
comment_id varchar(20) primary key,
post_id varchar(20),
comment_desc varchar(255),
comment_date datetime,
commentor_name varchar(30),
access_id varchar(5) default 'S',
foreign key (commentor_name) references registered_employee(profile_name)
on delete cascade
on update cascade,
foreign key (post_id) references post(post_id)
on delete cascade
on update cascade
);
```

```sql
#Create location table
drop table if exists location;
create table location
(
loc_id varchar(20) primary key,
post_id varchar(20),
name varchar(100),
description varchar(255),
latitude varchar(30),
longitude varchar(30),
access_id varchar(5) default 'S',
foreign key (post_id) references post(post_id)
on delete cascade
on update cascade
);

#Create like table

drop table if exists likes;
create table likes
(
like_id varchar(20) primary key,
post_id varchar(30),
multimedia_id varchar(10),
comment_id varchar(20),
loc_id varchar(20),
viewer_name varchar(20),
like_time datetime,
foreign key (post_id) references post(post_id) on delete cascade on update cascade,
foreign key (multimedia_id) references multimedia_content(multimedia_id) on delete cascade on update
cascade,
foreign key (comment_id) references comment(comment_id) on delete cascade on update cascade,
foreign key (loc_id) references location(loc_id) on delete cascade on update cascade,
foreign key (viewer_name) references registered_employee(profile_name) on delete cascade on update
cascade
);
```

Procedures :

```
DELIMITER $$

#Procedure 1- Insert into employee table
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_employee`(
IN e_ssn int,
IN e_email varchar(20)
)
BEGIN
INSERT INTO employee
(
ssn,
email_id
) Values
(
e_ssn,
e_email
);
END $$


#Procedure 2: insert into signin table
DROP PROCEDURE IF EXISTS `insert_signin_details` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_signin_details`(
IN e_ssn int,
IN e_password varchar(50),
IN e_profile varchar(30)
)
BEGIN
INSERT INTO signin
(
ssn,
password,
profile_name
) Values
(
e_ssn,
e_password,
e_profile
);
END $$
```

```sql
#Procedure 3: for inserting registered user
DROP PROCEDURE IF EXISTS `insert_registered_employee` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_registered_employee`(
IN profile_name varchar(30),
IN first_name varchar(20),IN last_name varchar(20),IN address varchar(255),IN designation varchar(20),
IN skills varchar(255),IN interests varchar(255),IN manager varchar(20),
IN level enum('A','B','C','D'),IN email_id varchar(50),
IN profile_pic blob,IN gender enum('Male','Female'),
IN access_id enum('P','T','F','S','FOF'),
IN create_time datetime
)
BEGIN
INSERT INTO registered_employee
(
profile_name,first_name, last_name,address,designation,skills,
interests,manager,level, email_id, profile_pic,gender,access_id,profile_created
) Values
(
profile_name,first_name,last_name,address,designation,skills,interests,manager,level,
email_id,profile_pic,gender,access_id, create_time
);
END $$


#Procedure 4: for inserting post
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_post`(
IN post_id varchar(30), IN post_time timestamp,
IN post_title varchar(150),IN post_desc varchar(255),
IN access_id varchar(3)
)
BEGIN insert into post
(
post_id,post_time,post_title,post_desc,access_id
) values
(post_id,post_time,post_title,post_desc,access_id
); END $$


#Procedure 5: saving records to wall
CREATE DEFINER=`root`@`localhost` PROCEDURE `add_to_wall`(
IN post_id varchar(20),
IN profile_name varchar(30)
```

```
)
BEGIN insert into wall
(
post_id ,
profile_name
)
values(post_id, profile_name
);
END $$


#Procedure 6 request to connect with colleagues
DROP PROCEDURE IF EXISTS `insert_relationship` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_relationship`(
p_sender_name varchar(30),
p_receiver_name varchar(20),
p_relation varchar(20),
p_time_request_sent timestamp,
friendship_status varchar(20)
)
BEGIN
insert into relationship (
sender_name , receiver_name , relation_type,
request_time, friendship_status
)
Values
(p_sender_name , p_receiver_name , p_relation , p_time_request_sent, friendship_status
);
END $$


#Procedure 7 insert multimedia contents
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_multimedia`(
IN p_multimedia_id varchar(10),
IN p_post_id varchar(20),
IN p_multimedia_name varchar(20),
IN p_multimedia_type varchar(20),
IN p_multimedia_content blob,
IN p_access_id varchar(5)
)
BEGIN insert into multimedia_content
(
multimedia_id, post_id, multimedia_name,
multimedia_type,
```

```
multimedia_content, access_id
) values
(
p_multimedia_id, p_post_id, p_multimedia_name, p_multimedia_type,
p_multimedia_content, p_access_id
); END $$


#Procedure 8 insert comments
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_comments`(
IN comment_id varchar(20),IN post_id varchar(20),
IN comment_desc varchar(255), IN comment_date datetime,
IN commentor_name varchar(30),
IN access_id varchar(5)
 )
BEGIN insert into comment
(
comment_id,
post_id, comment_desc, comment_date, commentor_name,access_id
)
values(
comment_id, post_id,
comment_desc, comment_date, commentor_name, access_id
); END $$


#Procedure 9: inserts location
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_locations`(
IN loc_id varchar(20),IN post_id varchar(20),
IN lname varchar(100),In description varchar(255),
IN latitude varchar(30), IN longitude varchar(30),
IN access_id varchar(5)
)
BEGIN insert into location
(
loc_id, post_id,name,description, latitude,longitude,access_id
)
values (
loc_id,post_id,lname,
description,latitude,longitude,access_id
); END $$

#Procedure 10:inserts likes
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_likes`(
IN p_like_id varchar(20),IN post_id varchar(30),IN p_multimedia_id varchar(10),
```

```sql
IN p_comment_id varchar(20), IN p_loc_id varchar(20),
IN p_viewer varchar(20), IN p_like_time datetime
)
BEGIN insert into likes (
like_id, post_id,
multimedia_id,comment_id,loc_id,
viewer_name,like_time
) values
(
p_like_id,post_id,p_multimedia_id,
p_comment_id,p_loc_id,
p_viewer,p_like_time
); END $$
```

```sql
#Procedure 11: Read from registered employee table
CREATE DEFINER=`root`@`localhost` PROCEDURE `registered_employee_acc`()
BEGIN select * from registered_employee;
END $$
```

```sql
#Procedure 12 :Read posts
CREATE DEFINER=`root`@`localhost` PROCEDURE `read_posts`()
BEGIN select * from post;
END $$
```

```sql
#Procedure 13 : Read wall
CREATE DEFINER=`root`@`localhost` PROCEDURE `read_wall`()
BEGIN select * from wall;
END $$
```

```sql
#Procedure 14: Read relationships
CREATE DEFINER=`root`@`localhost` PROCEDURE `view_relationship`()
BEGIN select * from relationship;
END $$
```

```sql
#Procedure 15: View Multimedia Contents
CREATE DEFINER=`root`@`localhost` PROCEDURE `multimedia_contents`()
BEGIN select * from multimedia_content;
END $$
```

```
#Procedure 16
CREATE DEFINER=`root`@`localhost` PROCEDURE `comment`()
BEGIN select * from comment;
END $$

#Procedure 17: View Location
CREATE DEFINER=`root`@`localhost` PROCEDURE `location`()
BEGIN select * from location;
END $$

# Procedure 18: View likes
CREATE DEFINER=`root`@`localhost` PROCEDURE `likes`()
BEGIN select * from likes;
END $$

#Procedure 19: View employee table
CREATE DEFINER=`root`@`localhost` PROCEDURE `employee_acc`()
BEGIN select * from employee;
END $$

#Procedure 20: View sign_in  table
CREATE DEFINER=`root`@`localhost` PROCEDURE `employee_signin`()
BEGIN select * from signin;
END $$

#Procedure 21: for inserting registered user
DROP PROCEDURE IF EXISTS `insert_profile_id` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `insert_profile_id`(
IN profile_name varchar(30),
IN email_id varchar(50),
IN fname varchar(20),
IN lname varchar(20),
)
BEGIN
INSERT INTO registered_employee
(
profile_name,
email_id,
first_name,
last_name
) Values (
profile_name, email_id,fname,lname
);
```

```
END $$
#Procedure 22: for iretrieving posts
DROP PROCEDURE IF EXISTS `post_retrieval` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `post_retrieval`(
IN profile_name varchar(30)
)
BEGIN

select *
from wall
where profile_name = profile_name or profile_name
 in
(select distinct receiver_name
from relationship
where sender_name = profile_name and friendship_status = 'Accepted' or friendship_status = 'sent' and
relation_type = 'T' or 'F'
);

END $$




DELIMITER ;
```

Update Procedure:

```
DELIMITER $$

DROP PROCEDURE IF EXISTS `update_post_table` $$
# Update Post Table
CREATE DEFINER=`root`@`localhost` PROCEDURE `update_post_table`
(
IN post_id varchar(30), IN post_time timestamp,
IN access_id varchar(3)
)
BEGIN
update post
set post_time= post_time,access_id = access_id
where post_id = post_id;
END $$

DELIMITER ;
```

FUNCTIONS:

use BigApple;

DELIMITER $$

```
#Function: To check for a new or an existing employee
DROP FUNCTION IF EXISTS `validate_user` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `validate_user`(
e_ssn int
)
RETURNS varchar(50)

BEGIN

        DECLARE check_employee varchar(50);

        select ssn into check_employee from signin where ssn=e_ssn;

        if check_employee!='NULL' Then
                return ("Employee subscribed to BigApple");
    else
                return ("New user registration. You have to sign up first!!");

        end if;
END $$

DELIMITER;
```

select validate_user(742347663) as Status;

```
select validate_user(742347663) as Status
```

☐ Show all | Number of rows: 25 ▼   Filter rows: Searcl

+ Options
Status
Employee subscribed to BigApple

select validate_user(542347663) as Status;

```
select validate_user(542347663) as Status
```

☐ Show all | Number of rows: 25 ▼ Filter

+ Options
Status
New user registration. You have to sign up first!!

Test Data.

Creating test data as part of the test framework for Big Apple.

Insert into employee

call insert_employee(742347663,'jjs816@nyu.edu');
call insert_employee(742347661,'gss383@nyu.edu');
call insert_employee(378092902,'ag6516@nyu.edu');
call insert_employee(407840577,'akk198@nyu.edu');
call insert_employee(940237837,'ryk123@nyu.edu');
call insert_employee(379486342,'sa4920@nyu.edu');
call insert_employee(1111111111,'test@nyu.edu');

				ssn	email_id
☐	Edit	Copy	Delete	378092902	ag6516@nyu.edu
☐	Edit	Copy	Delete	379486342	sa4920@nyu.edu
☐	Edit	Copy	Delete	407840577	akk198@nyu.edu
☐	Edit	Copy	Delete	742347661	gss383@nyu.edu
☐	Edit	Copy	Delete	742347663	jjs816@nyu.edu
☐	Edit	Copy	Delete	940237837	ryk123@nyu.edu
☐	Edit	Copy	Delete	1111111111	test@nyu.edu

Insert into signin

call insert_signin_details(742347663,'password','jjs816');
call insert_signin_details(742347661,'password','gss383');
call insert_signin_details(378092902,'password','ag6516');
call insert_signin_details(407840577,'password','akk198');
call insert_signin_details(940237837,'password','ryk123');
call insert_signin_details(379486342,'password','sa4920');

				ssn	password	profile_name
☐	🖉 Edit	🎝 Copy	⊝ Delete	378092902	password	ag6516
☐	🖉 Edit	🎝 Copy	⊝ Delete	407840577	password	akk198
☐	🖉 Edit	🎝 Copy	⊝ Delete	742347661	password	gss383
☐	🖉 Edit	🎝 Copy	⊝ Delete	742347663	password	jjs816
☐	🖉 Edit	🎝 Copy	⊝ Delete	940237837	password	ryk123
☐	🖉 Edit	🎝 Copy	⊝ Delete	379486342	password	sa4920

Insert into registered_employee

call insert_registered_employee('jjs816','Jay','Shah','New York','Senior Engineer','Java Php','Dancing','Karl','A','jjs816@nyu.edu',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','Male','S');
call insert_registered_employee('gss383','Gauri','Sarode','New York','Senior Engineer','Python Php','Movies','Karl','A','gss383@nyu.edu',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','Female','P');
call insert_registered_employee('ag6516','Anshul','Garg','Bayridge','Data Engineer','Java React','Mingling','Gauri','B','ag6516@nyu.edu',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','Male','S');
call insert_registered_employee('akk198','Arun','Kondani','Manhattan','Website developer','Python React','Cribbing','Gauri','B','akk198@nyu.edu',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','Male','P');
call insert_registered_employee('ryk123','Rahul','Keswani','Long Island','Data Scientist','React Php','Coding','Anshul','C','ryk123@nyu.edu',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','Male','S');
call insert_registered_employee('sa4920','Shubham','Aggarwal','Boston','Full stack developer','Python Java','Drinking','Anshul','C','sa4920@nyu.edu',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','Male','P');

				post_id	post_time	post_title	post_desc	access_id
☐	🖉 Edit	🎝 Copy	⊝ Delete	post_1	2010-07-22 22:30:12	First post	Day one it is!	
☐	🖉 Edit	🎝 Copy	⊝ Delete	post_2	2010-08-01 13:30:32	Second post	Second day it is!	
☐	🖉 Edit	🎝 Copy	⊝ Delete	post_3	2010-08-10 20:30:17	Third post	Some day it is!	
☐	🖉 Edit	🎝 Copy	⊝ Delete	post_4	2010-09-20 04:30:29	Fourth post	So much work such little time!	
☐	🖉 Edit	🎝 Copy	⊝ Delete	post_5	2010-09-22 12:30:30	Fifth post	In the intense meeting discussion!	
☐	🖉 Edit	🎝 Copy	⊝ Delete	post_6	2010-10-02 12:20:43	Sixth post	Had the best curly fries ever!	

↑__ ☐ Check all   With selected: 🖉 Edit   🎝 Copy   ⊝ Delete   🖳 Export

Insert into post

call insert_post('post_1','2010-07-22 22:30:12','First post','Day one it is!',' ');
call insert_post('post_2','2010-08-01 13:30:32','Second post','Second day it is!',' ');
call insert_post('post_3','2010-08-10 20:30:17','Third post','Some day it is!',' ');
call insert_post('post_4','2010-09-20 04:30:29','Fourth post','So much work such little time!',' ');
call insert_post('post_5','2010-09-22 12:30:30','Fifth post','In the intense meeting discussion!',' ');

call insert_post('post_6','2010-10-02 12:20:43','Sixth post','Had the best curly fries ever!',' ');

	multimedia_id	post_id	multimedia_name	Multimedia_type	multimedia_content	access_id
☐ ✎ Edit ⅀ Copy ⊖ Delete	mul_1	post_1	Miami Image	jpg	[BLOB - 15 B]	S
☐ ✎ Edit ⅀ Copy ⊖ Delete	mul_2	post_2	Florida image	jpg	[BLOB - 15 B]	P
☐ ✎ Edit ⅀ Copy ⊖ Delete	mul_3	post_1	Sunset beach	jpg	[BLOB - 15 B]	F
☐ ✎ Edit ⅀ Copy ⊖ Delete	mul_4	post_3	Curly fries	jpg	[BLOB - 15 B]	S
☐ ✎ Edit ⅀ Copy ⊖ Delete	mul_5	post_1	Miami Image	jpg	[BLOB - 15 B]	P

Insert into wall

call add_to_wall('post_1','jjs816');
call add_to_wall('post_2','gss383');
call add_to_wall('post_3','jjs816');
call add_to_wall('post_4','gss383');
call add_to_wall('post_5','gss383');
call add_to_wall('post_6','ryk123');

post_id	profile_name
post_1	jjs816
post_2	gss383
post_3	jjs816
post_4	gss383
post_5	gss383
post_6	ryk123

Insert into relationship

call insert_relationship('jjs816','gss383','T','2010-07-22 22:30:12','Accepted');
call insert_relationship('jjs816','ryk123','T','2012-07-22 22:30:12','Sent');
call insert_relationship('jjs816','akk198','F','2013-07-22 22:30:12','Accepted');
call insert_relationship('jjs816','ag6516','F','2010-07-22 22:30:12','Accepted');
call insert_relationship('gss383','ag6516','T','2010-07-22 22:30:12','Accepted');
call insert_relationship('ryk123','gss383','T','2010-07-22 22:30:12','Sent');

	sender_name	receiver_name	relation_type	request_time	friendship_status
☐ ✎ Edit ⅀ Copy ⊖ Delete	gss383	ag6516	T	2010-07-22 22:30:12	Accepted
☐ ✎ Edit ⅀ Copy ⊖ Delete	jjs816	ag6516	F	2010-07-22 22:30:12	Accepted
☐ ✎ Edit ⅀ Copy ⊖ Delete	jjs816	akk198	F	2013-07-22 22:30:12	Accepted
☐ ✎ Edit ⅀ Copy ⊖ Delete	jjs816	gss383	T	2010-07-22 22:30:12	Accepted
☐ ✎ Edit ⅀ Copy ⊖ Delete	jjs816	ryk123	T	2012-07-22 22:30:12	Sent
☐ ✎ Edit ⅀ Copy ⊖ Delete	ryk123	gss383	T	2010-07-22 22:30:12	Sent

Insert into multimedia_content

call insert_multimedia('mul_1','post_1','Miami
Image','jpg',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','S');
call insert_multimedia('mul_2','post_2','Florida
image','jpg',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','P');
call insert_multimedia('mul_3','post_1','Sunset
beach','jpg',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','F');
call insert_multimedia('mul_4','post_3','Curly
fries','jpg',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','S');
call insert_multimedia('mul_5','post_1','Miami
Image','jpg',x'C9CBBBCCCEB9C8CABCCCCEB9C9CBBB','P');

	multimedia_id	post_id	multimedia_name	Multimedia_type	multimedia_content	access_id
Edit Copy Delete	mul_1	post_1	Miami Image	jpg	[BLOB - 15 B]	S
Edit Copy Delete	mul_2	post_2	Florida image	jpg	[BLOB - 15 B]	P
Edit Copy Delete	mul_3	post_1	Sunset beach	jpg	[BLOB - 15 B]	F
Edit Copy Delete	mul_4	post_3	Curly fries	jpg	[BLOB - 15 B]	S
Edit Copy Delete	mul_5	post_1	Miami Image	jpg	[BLOB - 15 B]	P

Insert into comment

call insert_comments('cmnt_1','post_3','So cool!','2010-07-22 22:30:12','gss383','P');
call insert_comments('cmnt_2','post_4','Let me join in','2010-09-21 04:30:29','ryk123','F');
call insert_comments('cmnt_3','post_5','Where is this!','2010-09-22 15:30:30','ryk123','F');
call insert_comments('cmnt_4','post_5','How does this work?','2010-09-23 12:30:30','sa4920','P');
call insert_comments('cmnt_5','post_5','Nice post man','2010-09-24 22:30:30','jjs816','P');
call insert_comments('cmnt_6','post_2','Isnt it there already?','2010-09-25 12:30:30','jjs816','F');
call insert_comments('cmnt_7','post_1','How was your conference?','2010-09-23 12:30:30','sa4920','P');
call insert_comments('cmnt_8','post_1','Its good to see you here. We should meet','2010-09-24 22:30:30','jjs816','P');
call insert_comments('cmnt_9','post_1','When is the conference?','2010-09-25 12:30:30','gss383','P');

+ Options

	comment_id	post_id	comment_desc	comment_date	commentor_name	access_id
Edit Copy Delete	cmnt_1	post_3	So cool!	2010-07-22 22:30:12	gss383	P
Edit Copy Delete	cmnt_2	post_4	Let me join in	2010-09-21 04:30:29	ryk123	F
Edit Copy Delete	cmnt_3	post_5	Where is this!	2010-09-22 15:30:30	ryk123	F
Edit Copy Delete	cmnt_4	post_5	How does this work?	2010-09-23 12:30:30	sa4920	P
Edit Copy Delete	cmnt_5	post_5	Nice post man	2010-09-24 22:30:30	jjs816	P
Edit Copy Delete	cmnt_6	post_2	Isnt it there already?	2010-09-25 12:30:30	jjs816	F
Edit Copy Delete	cmnt_7	post_1	How was your conference?	2010-09-23 12:30:30	sa4920	P
Edit Copy Delete	cmnt_8	post_1	Its good to see you here. We should meet	2010-09-24 22:30:30	jjs816	P
Edit Copy Delete	cmnt_9	post_1	When is the conference?	2010-09-25 12:30:30	gss383	P

Check all   With selected: Edit   Copy   Delete   Export

Insert into Locations

call insert_locations('loc_1','post_1','Miami Convention Center','Conference on AI for next generations','40.7587° N', '73.9787° W','P');
call insert_locations('loc_2','post_2','Wall Street','Meeting for a tender passing','42.7217° N', '34.9787° W','F');
call insert_locations('loc_3','post_3','Bobst Washington people','Conference for product deployment','78.7587° S', '13.9787° E','P');
call insert_locations('loc_4','post_4','Android robotic palace','Making aurduino work','30.7587° E', '23.9787° S','F');

	loc_id	post_id	name	description	latitude	longitude	access_id
Edit Copy Delete	loc_1	post_1	Miami Convention Center	Conference on AI for next generations	40.7587° N	73.9787° W	P
Edit Copy Delete	loc_2	post_2	Wall Street	Meeting for a tender passing	42.7217° N	34.9787° W	F
Edit Copy Delete	loc_3	post_3	Bobst Washington people	Conference for product deployment	78.7587° S	13.9787° E	P
Edit Copy Delete	loc_4	post_4	Android robotic palace	Making aurduino work	30.7587° E	23.9787° S	F

↑ ☐ Check all   With selected: ✏ Edit 📋 Copy ⊖ Delete 📤 Export

Insert into likes:

call insert_likes('like_1','post_1','mul_1','cmnt_1',NULL,'gss383','2010-07-22 22:30:12');
call insert_likes('like_2','post_2','mul_1','cmnt_1',NULL,'akk198','2010-08-12 16:45:45');
call insert_likes('like_3','post_1','mul_1','cmnt_1',NULL,'akk198','2010-09-14 18:20:23');
call insert_likes('like_4','post_2','mul_1','cmnt_1',NULL,'ag6516','2010-09-22 19:35:19');
call insert_likes('like_5','post_2','mul_2','cmnt_1',NULL,'ag6516','2010-09-22 19:35:19');
call insert_likes('like_6','post_3',NULL,NULL,NULL,'ag6516','2010-09-22 19:35:19');
call insert_likes('like_7','post_2','mul_2','cmnt_2',NULL,'ag6516','2010-09-22 19:35:19');
call insert_likes('like_8','post_2','mul_1',NULL,NULL,'ag6516','2010-09-22 19:35:19');
call insert_likes('like_9','post_4','mul_1',NULL,NULL,'ag6516','2010-09-22 19:35:19');
call insert_likes('like_10','post_2','mul_3','cmnt_1',NULL,'ag6516','2010-09-22 19:35:19');

+ Options

	like_id	post_id	multimedia_id	comment_id	loc_id	viewer_name	like_time
Edit Copy Delete	like_1	post_1	mul_1	cmnt_1	NULL	gss383	2010-07-22 22:30:12
Edit Copy Delete	like_10	post_2	mul_3	cmnt_1	NULL	ag6516	2010-09-22 19:35:19
Edit Copy Delete	like_2	post_2	mul_1	cmnt_1	NULL	akk198	2010-08-12 16:45:45
Edit Copy Delete	like_3	post_1	mul_1	cmnt_1	NULL	akk198	2010-09-14 18:20:23
Edit Copy Delete	like_4	post_2	mul_1	cmnt_1	NULL	ag6516	2010-09-22 19:35:19
Edit Copy Delete	like_5	post_2	mul_2	cmnt_1	NULL	ag6516	2010-09-22 19:35:19
Edit Copy Delete	like_6	post_3	NULL	NULL	NULL	ag6516	2010-09-22 19:35:19
Edit Copy Delete	like_7	post_2	mul_2	cmnt_2	NULL	ag6516	2010-09-22 19:35:19
Edit Copy Delete	like_8	post_2	mul_1	NULL	NULL	ag6516	2010-09-22 19:35:19
Edit Copy Delete	like_9	post_4	mul_1	NULL	NULL	ag6516	2010-09-22 19:35:19

☐ Check all   With selected: ✏ Edit 📋 Copy ⊖ Delete 📤 Export

Views:

#Display Posts by all the users
CREATE OR REPLACE VIEW All_Posts_view AS
SELECT post_id, first_name as Name
FROM wall as w natural join registered_employee as re
where w.profile_name=re.profile_name;

#All Posts
select * from All_Posts_view;

#Specific posts
select post_id AS posts
from All_Posts_view
where Name='Gauri';

#Display Multimedia Contents of a specific location
CREATE OR REPLACE VIEW Multimedia_list_view AS
select m.multimedia_name as Name,m.Multimedia_type as Type,m.multimedia_content as Content
from multimedia_content as m, location as l
where l.post_id=m.post_id and l.name like '%Miami%' and (m.access_id='P' or m.access_id='F');

select * from Multimedia_list_view;

#Display friends list of any User.
Create OR Replace VIEW friendlist_view AS
select first_name as Friend,friendship_status as Status, request_time as Time
from relationship as r,registered_employee as e
where r.receiver_name=e.profile_name and r.sender_name = 'jjs816';

Queries:

These queries were created to check for different features of the Big Apple application:

#Check for connections for 2 users

select friendship_status as Status, request_time as Time
from relationship as r
where r.sender_name = 'jjs816' AND r.receiver_name='gss383';

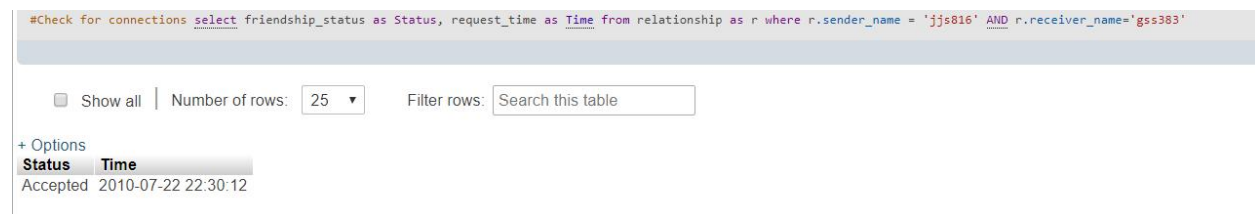#Display friends list of any User.
Create OR Replace VIEW friendlist_view AS
select first_name as Friend,friendship_status as Status, request_time as Time
from relationship as r,registered_employee as e
where r.receiver_name=e.profile_name and r.sender_name = 'jjs816';

#Count Total likes by post for specific user
select l.post_id as Post,count(l.post_id) as Like_Count
from likes as l natural join wall as w
where w.profile_name='jjs816'

group by l.post_id;

+ Options

Post	Like_Count
post_1	2
post_3	1

#Filter post by date
select post_title,post_desc
from post
where post_time between date_sub(now(), interval 10 year)
and date(now()) ;

```
#Filter post by date select post_title,post_desc from post where post_time between date_sub(now(), interval 10 year ) and date(now())
```

Show all | Number of rows: 25 ▼   Filter rows: Search this table   Sort by key: None ▼

· Options

post_title	post_desc
First post	Day one it is!
Second post	Second day it is!
Third post	Some day it is!
Fourth post	So much work such little time!
Fifth post	In the intense meeting discussion!
Sixth post	Had the best curly fries ever!

```
select * from signin where profile_name = 'jjs816' and password='password'
```

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

+ Options

←T→		ssn	password	profile_name
☐	🖉 Edit ᠍ Copy ⊝ Delete	742347663	password	jjs816

This Query selects all the post to be displayed on the wall of the user and his friends:

✔ Showing rows 0 - 3 (4 total, Query took 0.0004 seconds.)

```
select * from wall where profile_name = 'ryk123' or (profile_name in (select distinct receiver_name from relationship where sender_name = 'jjs816' and friendship_status = 'Accepted' or friendship_status = 'sent' and relation_type = 'T' or 'F'))
```

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

☐ Show all | Number of rows: 25 ▼   Filter rows: Search this table   Sort by key: None ▼

+ Options

post_id	profile_name
post_2	gss383
post_4	gss383
post_5	gss383
post_6	ryk123