**DDL SCRIPT**

CREATE TABLE Post (

Post\_id INT PRIMARY KEY ,

Caption VARCHAR(255),

Video BYTEA,

Song BYTEA,

Image BYTEA,

Location VARCHAR(50),

Post\_date DATE,

User\_id INT NOT NULL,

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE Users(

User\_id INT PRIMARY KEY,

Gender CHAR NOT NULL,

Fname VARCHAR(30) NOT NULL,

Lname VARCHAR(30) NOT NULL,

DOB DATE NOT NULL,

Is\_verified BOOLEAN NOT NULL,

Bio VARCHAR(100),

Password VARCHAR(15) NOT NULL,

Profile\_pic BYTEA

);

CREATE TABLE Likes (

Like\_time TIME DEFAULT CURRENT\_TIME,

User\_id INT NOT NULL,

Post\_id INT NOT NULL,

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Post\_id) REFERENCES Post(Post\_id) ON DELETE CASCADE ON UPDATE CASCADE,

PRIMARY KEY(User\_id, Post\_id)

);

CREATE TABLE Post\_comment (

Post\_cmt\_id INT PRIMARY KEY,

Comment\_time TIME DEFAULT CURRENT\_TIME,

Text\_content VARCHAR(100),

User\_id INT NOT NULL,

Post\_id INT NOT NULL,

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Post\_id) REFERENCES Post(Post\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE Ad (

Organizer\_name VARCHAR(30) PRIMARY KEY,

URL VARCHAR(255),

text\_content VARCHAR(50)

);

CREATE TABLE Collection (

Collection\_name VARCHAR(30) PRIMARY KEY,

Creation\_date DATE,

Description VARCHAR(120)

);

CREATE TABLE placed\_as (

Post\_id INT NOT NULL,

Organizer\_name VARCHAR(30) NOT NULL,

PRIMARY KEY (Post\_id, Organizer\_name),

FOREIGN KEY (Post\_id) REFERENCES Post(Post\_id),

FOREIGN KEY (Organizer\_name) REFERENCES Ad(Organizer\_name)

);

CREATE TABLE saved\_as (

Post\_id INT NOT NULL,

Collection\_name VARCHAR(30) NOT NULL,

PRIMARY KEY (Post\_id, Collection\_name),

FOREIGN KEY (Post\_id) REFERENCES Post(Post\_id),

FOREIGN KEY (Collection\_name) REFERENCES Collection(Collection\_name)

);

CREATE TABLE User\_email (

User\_id INT NOT NULL,

Email\_id VARCHAR(50),

PRIMARY KEY (User\_id, Email\_id),

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE User\_contact (

User\_id INT NOT NULL,

Contact\_no CHAR(13),

PRIMARY KEY (User\_id, Contact\_no),

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE Groups(

Group\_id INT PRIMARY KEY ,

Group\_name VARCHAR(30) NOT NULL,

Description VARCHAR(150),

Date\_created DATE

);

CREATE TABLE Message (

Content VARCHAR(100) NOT NULL,

Message\_timestamp TIMESTAMP,

Sendby\_user INT NOT NULL,

Receivedby\_grp INT,

Receivedby\_user INT NOT NULL,

PRIMARY KEY (Message\_timestamp, Sendby\_user,Receivedby\_user),

FOREIGN KEY (Sendby\_user) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Receivedby\_grp) REFERENCES Groups(Group\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Receivedby\_user) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE Comment\_Likes (

comment\_like\_time TIME DEFAULT CURRENT\_TIME,

User\_id INT NOT NULL,

Post\_cmt\_id INT NOT NULL,

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Post\_cmt\_id) REFERENCES Post\_comment(Post\_cmt\_id) ON DELETE CASCADE ON UPDATE CASCADE

PRIMARY KEY (User\_id,Post\_cmt\_id)

);

CREATE TABLE in1 (

User\_id INT NOT NULL,

Group\_id INT NOT NULL,

PRIMARY KEY (User\_id, Group\_id),

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Group\_id) REFERENCES Groups(Group\_id) ON DELETE CASCADE ON update cascade

)

CREATE TABLE Follow (

Follower\_id INT NOT NULL,

Followee\_id INT NOT NULL,

PRIMARY KEY (Follower\_id, Followee\_id),

FOREIGN KEY (Follower\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Followee\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE

);

CREATE TABLE Tag (

User\_id int NOT NULL,

Post\_id int NOT NULL,

PRIMARY KEY (User\_id,Post\_id),

FOREIGN KEY (User\_id) REFERENCES Users(User\_id) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY (Post\_id) REFERENCES Post(Post\_id) ON DELETE CASCADE ON UPDATE CASCADE

);