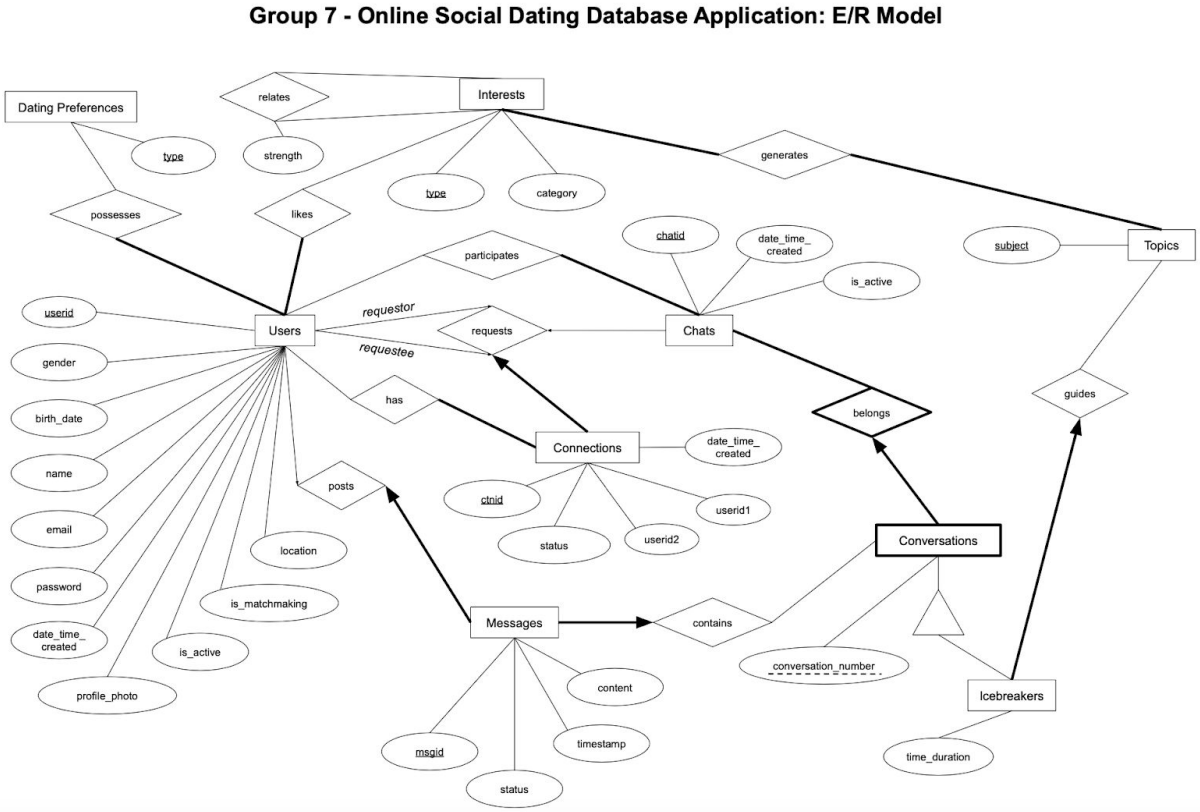
Project 2: Creating Your Database

COMP-421

Group 7

1)



2)

```
/* ========================================================== */
/* ========================================================== */
/* Create Tables */
/* ========================================================== */
/* ========================================================== */

CREATE TABLE Users(
    userid INT,
    gender CHAR(1) NOT NULL, /* M (male), F (female) or O (other) only */
```

```
    birth_date DATE,
    name VARCHAR(80) NOT NULL,
    email VARCHAR(320) NOT NULL UNIQUE, /* 64 local + 1 @ + 255 domain */
    password VARCHAR(128) NOT NULL,
    date_time_created DATE,
    profile_photo INT DEFAULT 0,
    is_active BOOLEAN,
    is_matchmaking BOOLEAN,
    location VARCHAR(100),
    PRIMARY KEY(userid),
    CONSTRAINT genderCheck CHECK(gender = 'M' OR gender = 'F' OR gender =
'O'),
    CONSTRAINT emailCheck CHECK(email LIKE '%@%'),
    CONSTRAINT photoChoice CHECK(profile_photo >= 0 AND profile_photo <
100)
);

CREATE TABLE Dating_Preferences(
    type VARCHAR(30),
    PRIMARY KEY(type)
);

CREATE TABLE Interests(
    type VARCHAR(30),
    category VARCHAR(30),
    PRIMARY KEY(type)
);

CREATE TABLE Topics(
    subject VARCHAR(30),
    PRIMARY KEY(subject)
);

CREATE TABLE Chats(
    chatid INT,
    date_time_created DATE,
    is_active BOOLEAN,
```

```
    requestid INT,
    PRIMARY KEY(chatid)
);

CREATE TABLE Conversations(
    chatid INT,
    conversation_number INT,
    PRIMARY KEY(chatid, conversation_number),
    FOREIGN KEY(chatid) REFERENCES Chats
);

CREATE TABLE Connections(
    ctnid INT,
    status VARCHAR(30),
    userid1 INT,
    userid2 INT,
    data_time_created DATE,
    PRIMARY KEY(ctnid),
    FOREIGN KEY(userid1) REFERENCES Users,
    FOREIGN KEY(userid2) REFERENCES Users
);

CREATE TABLE Messages(
    msgid INT,
    status VARCHAR(30),
    timestamp DATE,
    content VARCHAR(300),
    userid INT,
    chatid INT,
    conversation_number INT,
    PRIMARY KEY(msgid),
    FOREIGN KEY(userid) REFERENCES Users, /* For "posts" */
    FOREIGN KEY(conversation_number, chatid) REFERENCES Conversations /*
For "contains" */
);

CREATE TABLE Icebreakers(
```

```
    conversation_number INT,
    chatid INT,
    subject VARCHAR(30), /* For "guides" */
    time_duration INT,
    PRIMARY KEY(conversation_number, chatid, subject),
    FOREIGN KEY(conversation_number, chatid) REFERENCES Conversations,
    FOREIGN KEY(subject) REFERENCES Topics
);

/* ------------------------------------------------------- */
/* ------------------------------------------------------- */
/* ------------------------------------------------------- */
/* ------------------------------------------------------- */

CREATE TABLE relates(
    type1 VARCHAR(30),
    type2 VARCHAR(30),
    strength INT,
    PRIMARY KEY(type1, type2),
    FOREIGN KEY(type1) REFERENCES Interests,
    FOREIGN KEY(type2) REFERENCES Interests,
    CONSTRAINT min_max_str CHECK(strength >= 0 AND strength <= 10)
);

/*
- Participation constraints not met
*/
CREATE TABLE generates(
    type VARCHAR(30),
    subject VARCHAR(30),
    PRIMARY KEY(type, subject),
    FOREIGN KEY(type) REFERENCES Interests,
    FOREIGN KEY(subject) REFERENCES Topics
);

/*
- Participation constraint not met
```

```
*/
CREATE TABLE possesses(
    userid INT,
    type VARCHAR(30),
    PRIMARY KEY(userid, type),
    FOREIGN KEY(userid) REFERENCES Users,
    FOREIGN KEY(type) REFERENCES Dating_Preferences
);

/*
- Participation constraint not met
*/
CREATE TABLE likes(
    userid INT,
    type VARCHAR(30),
    PRIMARY KEY(userid, type),
    FOREIGN KEY(userid) REFERENCES Users,
    FOREIGN KEY(type) REFERENCES Interests
);

/*
- Participation constraint not met
*/
CREATE TABLE participates(
    userid INT,
    chatid INT,
    --type VARCHAR(30),
    PRIMARY KEY(userid, chatid),
    FOREIGN KEY(userid) REFERENCES Users,
    FOREIGN KEY(chatid) REFERENCES Chats
    --FOREIGN KEY(type) REFERENCES Interests
);

CREATE TABLE requests(
    userid_Request INT NOT NULL,
    userid_Requestee INT NOT NULL,
    chatid INT,
```

```
    ctnid INT NOT NULL,
    PRIMARY KEY(ctnid),
    FOREIGN KEY(userid_Request) REFERENCES Users,
    FOREIGN KEY(userid_Requestee) REFERENCES Users,
    FOREIGN KEY(chatid) REFERENCES Chats,
    FOREIGN KEY(ctnid) REFERENCES Connections
);

/* Key-Part constraint => No table */

/* ------------------------------
CREATE TABLE guides(
);
 ------------------------------*/

/* Weak-Ent constraint => No table */

/* ------------------------------
CREATE TABLE belongs(
);
 ------------------------------*/

CREATE TABLE has(
    userid INT,
    ctnid INT,
    PRIMARY KEY(userid, ctnid),
    FOREIGN KEY(userid) REFERENCES Users,
    FOREIGN KEY(ctnid) REFERENCES Connections
);

/* Key-Part constraint => No table */

/* ------------------------------
CREATE TABLE posts(

);
 ------------------------------*/
```

```
/* Key-Part constraint => No table */

/* ------------------------------
CREATE TABLE contains(
);
 --------------------------------*/
```

```
cs421=> \d chats users
        Table "cs421g07.chats"
     Column       |  Type   | Modifiers
------------------+---------+-----------
 chatid           | integer | not null
 date_time_created | date    |
 is_active        | boolean |
 requestid        | integer |
Indexes:
   "chats_pkey" PRIMARY KEY, btree (chatid)
Referenced by:
   TABLE "conversations" CONSTRAINT "conversations_chatid_fkey" FOREIGN KEY (chatid)
REFERENCES chats(chatid)
   TABLE "participates" CONSTRAINT "participates_chatid_fkey" FOREIGN KEY (chatid)
REFERENCES chats(chatid)
   TABLE "requests" CONSTRAINT "requests_chatid_fkey" FOREIGN KEY (chatid)
REFERENCES chats(chatid)

\d: extra argument "users" ignored
cs421=> \d connections
           Table "cs421g07.connections"
     Column       |         Type          | Modifiers
------------------+-----------------------+-----------
 ctnid            | integer               | not null
 status           | character varying(30) |
 userid1          | integer               |
 userid2          | integer               |
 data_time_created | date                  |
Indexes:
   "connections_pkey" PRIMARY KEY, btree (ctnid)
Foreign-key constraints:
```

```
    "connections_userid1_fkey" FOREIGN KEY (userid1) REFERENCES users(userid)
    "connections_userid2_fkey" FOREIGN KEY (userid2) REFERENCES users(userid)
Referenced by:
    TABLE "has" CONSTRAINT "has_ctnid_fkey" FOREIGN KEY (ctnid) REFERENCES
connections(ctnid)
    TABLE "requests" CONSTRAINT "requests_ctnid_fkey" FOREIGN KEY (ctnid)
REFERENCES connections(ctnid)

cs421=> \d conversations
       Table "cs421g07.conversations"
       Column        |  Type   | Modifiers
---------------------+---------+-----------
 chatid              | integer | not null
 conversation_number | integer | not null
Indexes:
    "conversations_pkey" PRIMARY KEY, btree (chatid, conversation_number)
Foreign-key constraints:
    "conversations_chatid_fkey" FOREIGN KEY (chatid) REFERENCES chats(chatid)
Referenced by:
    TABLE "icebreakers" CONSTRAINT "icebreakers_conversation_number_fkey" FOREIGN
KEY (conversation_number, chatid) REFERENCES conversations(chatid,
conversation_number)
    TABLE "messages" CONSTRAINT "messages_conversation_number_fkey" FOREIGN KEY
(conversation_number, chatid) REFERENCES conversations(chatid, conversation_number)

cs421=> \d dating_preferences
    Table "cs421g07.dating_preferences"
 Column |         Type          | Modifiers
--------+-----------------------+-----------
 type   | character varying(30) | not null
Indexes:
    "dating_preferences_pkey" PRIMARY KEY, btree (type)
Referenced by:
    TABLE "possesses" CONSTRAINT "possesses_type_fkey" FOREIGN KEY (type)
REFERENCES dating_preferences(type)

cs421=> \d generates
         Table "cs421g07.generates"
 Column  |         Type          | Modifiers
---------+-----------------------+-----------
 type    | character varying(30) | not null
 subject | character varying(30) | not null
Indexes:
```

```
    "generates_pkey" PRIMARY KEY, btree (type, subject)
Foreign-key constraints:
    "generates_subject_fkey" FOREIGN KEY (subject) REFERENCES topics(subject)
    "generates_type_fkey" FOREIGN KEY (type) REFERENCES interests(type)

cs421=> \d has
     Table "cs421g07.has"
 Column |  Type   | Modifiers
--------+---------+-----------
 userid | integer | not null
 ctnid  | integer | not null
Indexes:
    "has_pkey" PRIMARY KEY, btree (userid, ctnid)
Foreign-key constraints:
    "has_ctnid_fkey" FOREIGN KEY (ctnid) REFERENCES connections(ctnid)
    "has_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)

cs421=> \d icebreakers
              Table "cs421g07.icebreakers"
       Column        |         Type          | Modifiers
---------------------+-----------------------+-----------
 conversation_number | integer               | not null
 chatid              | integer               | not null
 subject             | character varying(30) | not null
 time_duration       | integer               |
Indexes:
    "icebreakers_pkey" PRIMARY KEY, btree (conversation_number, chatid, subject)
Foreign-key constraints:
    "icebreakers_conversation_number_fkey" FOREIGN KEY (conversation_number, chatid)
REFERENCES conversations(chatid, conversation_number)
    "icebreakers_subject_fkey" FOREIGN KEY (subject) REFERENCES topics(subject)

cs421=> \d interests
          Table "cs421g07.interests"
  Column  |         Type          | Modifiers
----------+-----------------------+-----------
 type     | character varying(30) | not null
 category | character varying(30) |
Indexes:
    "interests_pkey" PRIMARY KEY, btree (type)
Referenced by:
    TABLE "generates" CONSTRAINT "generates_type_fkey" FOREIGN KEY (type)
REFERENCES interests(type)
```

```
    TABLE "likes" CONSTRAINT "likes_type_fkey" FOREIGN KEY (type) REFERENCES
interests(type)
    TABLE "relates" CONSTRAINT "relates_type1_fkey" FOREIGN KEY (type1) REFERENCES
interests(type)
    TABLE "relates" CONSTRAINT "relates_type2_fkey" FOREIGN KEY (type2) REFERENCES
interests(type)

cs421=> \d likes
          Table "cs421g07.likes"
 Column |         Type          | Modifiers
--------+-----------------------+-----------
 userid | integer               | not null
 type   | character varying(30) | not null
Indexes:
    "likes_pkey" PRIMARY KEY, btree (userid, type)
Foreign-key constraints:
    "likes_type_fkey" FOREIGN KEY (type) REFERENCES interests(type)
    "likes_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)

cs421=> \d messages
               Table "cs421g07.messages"
      Column        |          Type          | Modifiers
--------------------+------------------------+-----------
 msgid              | integer                | not null
 status             | character varying(30)  |
 timestamp          | date                   |
 content            | character varying(300) |
 userid             | integer                |
 chatid             | integer                |
 conversation_number | integer               |
Indexes:
    "messages_pkey" PRIMARY KEY, btree (msgid)
Foreign-key constraints:
    "messages_conversation_number_fkey" FOREIGN KEY (conversation_number, chatid)
REFERENCES conversations(chatid, conversation_number)
    "messages_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)

cs421=> \d participates
Table "cs421g07.participates"
 Column |  Type   | Modifiers
--------+---------+-----------
 userid | integer | not null
 chatid | integer | not null
```

```
Indexes:
    "participates_pkey" PRIMARY KEY, btree (userid, chatid)
Foreign-key constraints:
    "participates_chatid_fkey" FOREIGN KEY (chatid) REFERENCES chats(chatid)
    "participates_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)

cs421=> \d possesses
         Table "cs421g07.possesses"
 Column |         Type          | Modifiers
--------+-----------------------+-----------
 userid | integer               | not null
 type   | character varying(30) | not null
Indexes:
    "possesses_pkey" PRIMARY KEY, btree (userid, type)
Foreign-key constraints:
    "possesses_type_fkey" FOREIGN KEY (type) REFERENCES dating_preferences(type)
    "possesses_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)

cs421=> \d relates
           Table "cs421g07.relates"
  Column  |         Type          | Modifiers
----------+-----------------------+-----------
 type1    | character varying(30) | not null
 type2    | character varying(30) | not null
 strength | integer               |
Indexes:
    "relates_pkey" PRIMARY KEY, btree (type1, type2)
Check constraints:
    "min_max_str" CHECK (strength >= 0 AND strength <= 10)
Foreign-key constraints:
    "relates_type1_fkey" FOREIGN KEY (type1) REFERENCES interests(type)
    "relates_type2_fkey" FOREIGN KEY (type2) REFERENCES interests(type)

cs421=> \d requests
        Table "cs421g07.requests"
      Column      |  Type   | Modifiers
------------------+---------+-----------
 userid_request   | integer | not null
 userid_requestee | integer | not null
 chatid           | integer |
 ctnid            | integer | not null
Indexes:
    "requests_pkey" PRIMARY KEY, btree (ctnid)
```

```
Foreign-key constraints:
    "requests_chatid_fkey" FOREIGN KEY (chatid) REFERENCES chats(chatid)
    "requests_ctnid_fkey" FOREIGN KEY (ctnid) REFERENCES connections(ctnid)
    "requests_userid_request_fkey" FOREIGN KEY (userid_request) REFERENCES
users(userid)
    "requests_userid_requestee_fkey" FOREIGN KEY (userid_requestee) REFERENCES
users(userid)

cs421=> \d topics
           Table "cs421g07.topics"
 Column  |         Type          | Modifiers
---------+-----------------------+-----------
 subject | character varying(30) | not null
Indexes:
    "topics_pkey" PRIMARY KEY, btree (subject)
Referenced by:
    TABLE "generates" CONSTRAINT "generates_subject_fkey" FOREIGN KEY (subject)
REFERENCES topics(subject)
    TABLE "icebreakers" CONSTRAINT "icebreakers_subject_fkey" FOREIGN KEY (subject)
REFERENCES topics(subject)

cs421=> \d users
                Table "cs421g07.users"
      Column       |          Type          | Modifiers
-------------------+------------------------+-----------
 userid            | integer                | not null
 gender            | character(1)           | not null
 birth_date        | date                   |
 name              | character varying(80)  | not null
 email             | character varying(320) | not null
 password          | character varying(128) | not null
 date_time_created | date                   |
 profile_photo     | integer                | default 0
 is_active         | boolean                |
 is_matchmaking    | boolean                |
 location          | character varying(100) |
Indexes:
    "users_pkey" PRIMARY KEY, btree (userid)
    "users_email_key" UNIQUE CONSTRAINT, btree (email)
Check constraints:
    "emailcheck" CHECK (email::text ~~ '%@%'::text)
    "gendercheck" CHECK (gender = 'M'::bpchar OR gender = 'F'::bpchar OR gender =
'O'::bpchar)
```

"photochoice" CHECK (profile_photo >= 0 AND profile_photo < 100)
Referenced by:
TABLE "connections" CONSTRAINT "connections_userid1_fkey" FOREIGN KEY (userid1) REFERENCES users(userid)
TABLE "connections" CONSTRAINT "connections_userid2_fkey" FOREIGN KEY (userid2) REFERENCES users(userid)
TABLE "has" CONSTRAINT "has_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)
TABLE "likes" CONSTRAINT "likes_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)
TABLE "messages" CONSTRAINT "messages_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)
TABLE "participates" CONSTRAINT "participates_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)
TABLE "possesses" CONSTRAINT "possesses_userid_fkey" FOREIGN KEY (userid) REFERENCES users(userid)
TABLE "requests" CONSTRAINT "requests_userid_request_fkey" FOREIGN KEY (userid_request) REFERENCES users(userid)
TABLE "requests" CONSTRAINT "requests_userid_requestee_fkey" FOREIGN KEY (userid_requestee) REFERENCES users(userid)

3)

```
/* Insert 1 */
/* ----------------------------------------------------*/
INSERT INTO Users VALUES(
    500, 'M', '1996-01-02', 'Bob', 'bob1y4u3@hotmail.com', '123abc',
        '2020-01-13', 1, TRUE, TRUE, 'Montreal');

INSERT INTO possesses VALUES(500, 'F');

/* Insert 2*/
/* ----------------------------------------------------*/
INSERT INTO Users VALUES(
    501, 'F', '1995-02-03', 'Ashley', 'ashley15748@hotmail.com',
'password',
        '2020-03-12', 1, FALSE, FALSE, 'Montreal');

INSERT INTO possesses VALUES(501, 'M');

/* Insert 3 */
```

```
/* ------------------------------------------------*/
INSERT INTO Users VALUES(
    502, 'M', '1989-11-27', 'Charles', 'charles153290@gmail.com',
'123abc',
        '2019-05-29', 15, TRUE, TRUE, 'New York');

INSERT INTO possesses VALUES(502, 'M/F');

/* Insert 4 */
/* ------------------------------------------------*/
INSERT INTO Users VALUES(
    503, 'M', '1995-07-14', 'George', 'george8492672@gmail.com',
'password',
        '2020-01-11', 2, TRUE, TRUE, 'Montreal');

INSERT INTO possesses VALUES(503, 'F');

/* Insert 5 */
/* ------------------------------------------------*/
INSERT INTO Users VALUES(
    504, 'F', '1994-09-23', 'James', 'james1239201@hotmail.com',
    '123abc', '2020-03-11', 1, TRUE, FALSE, 'New York');

INSERT INTO possesses VALUES(504, 'M');

/* Other inserts (Maybe can be automated in java) */
INSERT INTO Interests VALUES
    ('Soccer', 'Sports'),
    ('Hockey', 'Sports'),
    ('Rock', 'Music'),
    ('Hip-Hop', 'Music'),
    ('Adventures', 'Misc.');
```

```
cs421=> \i insertTables.sql
INSERT 0 1
INSERT 0 1
INSERT 0 1
```

```
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 5
```

4)

```
cs421=> select * from chats LIMIT 10;
 chatid | date_time_created | is_active | requestid
--------+-------------------+-----------+-----------
      0 | 2018-06-12        | t         |         1
      1 | 2018-10-01        | t         |         2
      2 | 2019-10-23        | t         |         3
      3 | 2018-07-17        | t         |         4
      4 | 2017-02-01        | t         |         5
      5 | 2017-11-09        | t         |         6
      6 | 2018-04-27        | t         |         7
      7 | 2018-03-02        | t         |         8
      8 | 2019-08-03        | t         |         9
      9 | 2019-01-13        | f         |        13
(10 rows)

cs421=> select * from connections LIMIT 10;
 ctnid |  status  | userid1 | userid2 | data_time_created
-------+----------+---------+---------+-------------------
     0 | pending  |       0 |      66 | 2017-07-23
     1 | accepted |       1 |     318 | 2018-06-12
     2 | rejected |       2 |     389 | 2018-10-01
     3 | accepted |       3 |     383 | 2019-10-23
     4 | accepted |       4 |     437 | 2018-07-17
     5 | pending  |       5 |     171 | 2017-02-01
     6 | rejected |       6 |     216 | 2017-11-09
     7 | pending  |       7 |     169 | 2018-04-27
     8 | pending  |       8 |      26 | 2018-03-02
     9 | rejected |       9 |     381 | 2019-08-03
(10 rows)

cs421=> select * from conversations LIMIT 10;
 chatid | conversation_number
```

```
--------+--------------------
     30 |                  0
    339 |                  1
    316 |                  2
    328 |                  3
    252 |                  4
    370 |                  5
    153 |                  6
    335 |                  7
     83 |                  8
    172 |                  9
(10 rows)

cs421=> select * from dating_preferences LIMIT 10;
 type
-------
 M
 F
 O
 M/F
 M/O
 F/O
 M/F/O
 NULL
(8 rows)

cs421=> select * from generates LIMIT 10;
   type   | subject
----------+---------
 football | sports
 pingpong | sports
 baseball | sports
 reading  | food
 dancing  | netflix
 drawing  | netflix
(6 rows)

cs421=> select * from has LIMIT 10;
 userid | ctnid
--------+-------
      0 |     0
     66 |     0
      1 |     1
```

```
   318 |     1
     2 |     2
   389 |     2
     3 |     3
   383 |     3
     4 |     4
   437 |     4
(10 rows)

cs421=> select * from icebreakers LIMIT 10;
 conversation_number | chatid | subject | time_duration
---------------------+--------+---------+---------------
(0 rows)

cs421=> select * from interests LIMIT 10;
   type   | category
----------+----------
 football | sport
 pingpong | sport
 baseball | sport
 reading  | others
 dancing  | others
 drawing  | others
 Soccer   | Sports
 Hockey   | Sports
 Rock     | Music
 Hip-Hop  | Music
(10 rows)

cs421=> select * from likes LIMIT 10;
 userid |   type
--------+----------
      0 | reading
      1 | football
      2 | reading
      3 | dancing
      4 | drawing
      5 | dancing
      6 | pingpong
      7 | drawing
      8 | pingpong
      9 | reading
(10 rows)
```

```
cs421=> select * from messages LIMIT 10;
 msgid |  status   | timestamp  |         content          | userid | chatid | conversation_number
-------+-----------+------------+--------------------------+--------+--------+---------------------
     0 | delivered | 2018-06-12 | Hey                      |      1 |      0 |
     1 | delivered | 2018-06-12 | Hey                      |      1 |      0 |
     2 | delivered | 2018-06-12 | Tim Hortons at 6         |    318 |      0 |
     3 | seen      | 2018-10-01 | One                      |      2 |      1 |
     4 | sent      | 2018-10-01 | Two                      |      2 |      1 |
     5 | sent      | 2018-10-01 | This is freaking boring |    389 |      1 |
     6 | seen      | 2018-10-01 | Hello                    |      2 |      1 |
     7 | pending   | 2018-10-01 | How is it going?         |    389 |      1 |
     8 | sent      | 2019-10-23 | University               |    383 |      2 |
     9 | pending   | 2019-10-23 | How are you?             |      3 |      2 |
(10 rows)

cs421=> select * from participates LIMIT 10;
 userid | chatid
--------+--------
      1 |      0
    318 |      0
      2 |      1
    389 |      1
      3 |      2
    383 |      2
      4 |      3
    437 |      3
      5 |      4
    171 |      4
(10 rows)

cs421=> select * from possesses LIMIT 10;
 userid | type
--------+------
      0 | F
      1 | F
      2 | F
      3 | F
      4 | F
      5 | F
      6 | F
      7 | F
      8 | F
```

```
      9 | F
(10 rows)

cs421=> select * from relates LIMIT 10;
  type1   |  type2   | strength
----------+----------+----------
 football | pingpong |        5
 football | baseball |        5
 pingpong | baseball |        2
 reading  | dancing  |        8
 reading  | drawing  |        6
 dancing  | drawing  |       10
(6 rows)

cs421=> select * from requests LIMIT 10;
 userid_request | userid_requestee | chatid | ctnid
----------------+------------------+--------+-------
              0 |               66 |        |     0
              1 |              318 |      0 |     1
              2 |              389 |      1 |     2
              3 |              383 |      2 |     3
              4 |              437 |      3 |     4
              5 |              171 |      4 |     5
              6 |              216 |      5 |     6
              7 |              169 |      6 |     7
              8 |               26 |      7 |     8
              9 |              381 |      8 |     9
(10 rows)

cs421=> select * from topics LIMIT 10;
 subject
---------
 sports
 food
 netflix
(3 rows)

userid | gender | birth_date |   name    |           email           |    password    | date_time_created |
profile_photo | is_active | is_matchmaking |  location
--------+--------+------------+-----------+---------------------------+----------------+-------------------+-------------
-+-----------+----------------+-------------
      0 | M      | 2001-10-09 | Spencer   | spencer14904@outlook.com  | mypassword     |
2018-02-17        |            65 | f         | f              | Houston
```

```
    1 | M      | 1977-01-06 | Rick      | rick25376@gmail.com       | qwerty         | 2017-12-26       |
72 | f        | f             | Detroid
    2 | M      | 1986-11-15 | Dylan     | dylan83955@yahoo.com      | qwerty         | 2019-08-06
|          80 | f        | f             | Houston
    3 | F      | 1972-05-06 | Abigail   | abigail91452@gmail.com    | abigail94055   | 2018-02-21
|          53 | f        | f             | Ottawa
    4 | M      | 1968-04-26 | Nathan    | nathan61996@hotmail.com   | qwerty         | 2017-08-23
|          83 | f        | f             | Columbus
    5 | M      | 1963-10-09 | Rick      | rick27688@gmail.ca        | qwertyuiop     | 2019-06-12       |
93 | f        | f             | Houston
    6 | M      | 2000-05-18 | Don       | don58825@yahoo.com        | password       | 2017-07-02
|          64 | f        | f             | San Antonio
    7 | M      | 2002-04-11 | Julien    | julien94967@hotmail.ca    | iloveyou       | 2019-11-12       |
87 | f        | f             | Austin
    8 | M      | 1962-05-16 | Christian | christian87096@hotmail.ca | christian82234 | 2017-09-25
|          93 | f        | f             | Boston
    9 | M      | 1982-01-02 | John      | john5288@hotmail.com      | admin          | 2017-11-23
|          11 | t        | t             | L.A.
(10 rows)
```

```
cs421=> select * from icebreakers LIMIT 10;
 conversation_number | chatid | subject | time_duration
---------------------+--------+---------+---------------
                   0 |     30 | netflix |             4
                   7 |    335 | food    |             3
                  10 |    362 | sports  |             3
                  12 |     13 | food    |             4
                  16 |     42 | netflix |             2
                  17 |    140 | sports  |             0
                  19 |    110 | food    |             1
                  24 |    277 | food    |             4
                  25 |    171 | food    |             3
                  28 |    312 | sports  |             5
(10 rows)
```

5)

5.1) This query finds all email addresses of users who currently have at least one active chat on the application.

```
SELECT email
FROM users
WHERE userid
IN (SELECT userid
FROM chats
where is_active = 'true')
;
```

```
          email
--------------------------
 spencer14904@outlook.com
 rick25376@gmail.com
 dylan83955@yahoo.com
 abigail91452@gmail.com
 nathan61996@hotmail.com
 rick27688@gmail.ca
 don58825@yahoo.com
 julien94967@hotmail.ca
 christian87096@hotmail.ca
 john5288@hotmail.com
 william62157@hotmail.ca
```

cindy32308@hotmail.com
benjamin91528@outlook.ca
logan5607@hotmail.com
lucy63883@mail.com
don56675@gmail.com
luke13958@hotmail.com
steve23134@hotmail.com
jeff57703@hotmail.ca
mateo66764@gmail.com
liam86444@live.com
eric40031@hotmail.com
george48620@aol.com
connor98270@outlook.com
peter26305@hotmail.ca
dan91894@gmail.ca
ian5154@yahoo.com
vlad90850@hotmail.com
patrick48756@gmail.com
spencer67963@outlook.ca
charles37467@gmail.com
sebastian5051@hotmail.ca
zack33850@gmail.com
benjamin42438@gmail.com
chris54467@gmail.ca
timmy71893@outlook.com
majd816@gmail.com
logan72569@yahoo.com
dan36618@gmail.com
jerry91616@live.com
chris23971@outlook.ca
patrick93572@hotmail.com
spencer49867@gmail.com
charles35767@hotmail.com
william65030@gmail.com
chris94339@gmail.com
mateo1026@hotmail.com
mark15550@yahoo.com
jack59460@outlook.com
patrick69503@outlook.com
(50 rows)

5.2) This query counts the number of conversations that are no longer active and that, therefore, can be deleted by the application to optimize storage use.

```
SELECT COUNT(chatid)
FROM conversations
WHERE chatid IN (SELECT chatid
FROM chats
WHERE is_active = 'false')
;

 count
-------
   106
(1 row)
```

5.3) This query counts the number of positive (e.g. like, love, joy) words shared in messages between users. This allows the application to perform sentiment analysis on how well our application is matchmaking users and gain insight on how well matched users are enjoying their conversations.

```
SELECT COUNT(*)
FROM
(
SELECT msgid, content
FROM messages
WHERE content LIKE '%like%'
UNION
SELECT msgid, content
FROM messages
WHERE content LIKE '%love%'
UNION
SELECT msgid, content
FROM messages
WHERE content LIKE '%meet%'
UNION
SELECT msgid, content
FROM messages
WHERE content LIKE '%joy%'
UNION
SELECT msgid, content
FROM messages
WHERE content LIKE '%lol%'
```

```
) AS goodWords
;

 count
-------
   501
(1 row)
```

5.4) This query counts the number of actively matchmaking users who have a given interest, which can help the application more efficiently find icebreaker topics for future matches by starting with interests that are most shared between users.

```
SELECT type, COUNT(userid) AS Users
FROM
(
SELECT *
FROM likes
WHERE likes.userid IN
(SELECT userid
FROM users
WHERE is_matchmaking = 'true'
))
AS userLikes
GROUP BY type
ORDER BY users
;

   type           | users
------------------+--------
 football         |     2
 drawing          |     2
 pingpong         |     2
 reading          |     5
 dancing          |     5
 baseball         |     6
(6 rows)
```

5.5) This query finds all users that have interests with the highest strength (with other interests) via the relates table. These users are most likely to have successful conversations (should the topic of conversation be generated by these interests) with their matched users.

```
SELECT *
FROM users
WHERE userid IN
(SELECT userid
FROM likes, (SELECT *
FROM relates
WHERE strength >= all (SELECT strength FROM relates)
) AS types
WHERE likes.type = types.type1
OR likes.type = types.type2
)
LIMIT 50;

userid |   name   |   location
--------+----------+---------------
     3 | Abigail  | Ottawa
     4 | Nathan   | Columbus
     5 | Rick     | Houston
     7 | Julien   | Austin
    11 | Cindy    | Seattle
    12 | Benjamin | San Jose
    17 | Steve    | Indianapolis
    21 | Eric     | Phoenix
    22 | George   | Washington
    28 | Patrick  | San Francisco
    32 | Zack     | Houston
    34 | Chris    | Denver
    37 | Logan    | Edmonton
    46 | Mateo    | Dallas
    47 | Mark     | Phoenix
    48 | Jack     | El Paso
    51 | Ezekiel  | Houston
    52 | Noah     | Montreal
    53 | Liam     | Jacksonville
    62 | Mike     | Jacksonville
    63 | Jack     | Ottawa
    72 | Tim      | Jacksonville
    74 | Ashley   | Seattle
    77 | Marco    | Chicago
    79 | Mike     | El Paso
    89 | Xavier   | Dallas
    91 | Vlad     | Houston
    93 | Connor   | Denver
```

95 | Peter | Chicago
96 | Jerry | Philadelphia
99 | Logan | San Antonio
103 | Jack | Columbus
111 | Hannah | Chicago
115 | Tony | New York
121 | Melissa | San Francisco
122 | Susie | El Paso
125 | Tim | Chicago
129 | Ian | El Paso
132 | Vlad | San Antonio
134 | Patrick | Calgary
135 | Robert | Ottawa
138 | Peter | Toronto
142 | Trish | Vancouver
145 | Edward | Detroid
147 | Dan | Columbus
151 | Xavier | Edmonton
160 | Camila | Ottawa
162 | Natalie | San Diego
165 | Benjamin | Montreal
166 | Brandon | Washington
(50 rows)

6)

6.1) This data modification command updates messages with pending status to sent.

```
SELECT *
FROM messages;
UPDATE messages
SET status = 'sent'
WHERE status = 'pending'
;

UPDATE 356
```

6.2) This data modification command inserts the results from the query that finds the interests with the highest strength between them into the topics table.

```
INSERT INTO topics
(
SELECT type1
FROM relates
WHERE strength >= all (SELECT strength FROM relates)
UNION
SELECT type2
FROM relates
WHERE strength >= all (SELECT strength FROM relates)
)
;
```

```
INSERT 0 2
```

6.3) This data modification command removes all the relationship strengths between interests below some certain threshold (in this case 5), analogous to the relationships being too weakly related to warrant a connection.

```
DELETE FROM relates
WHERE strength <= 5
;
```

```
DELETE 3
```

6.4) This data modification command updates all the connections with status “pending” to the value “expired” if they’re past a certain age (in this case: if they’re from 2018 or older).

```
UPDATE connections
SET status = 'expired' WHERE status = 'pending' AND data_time_created < '2019-01-01'
;
```

```
UPDATE 116
```

7)

7.1) This view represents all of the users (userid) with some interest(s) that don’t belong to a specific category, and therefore are categorized as “other”.

```
CREATE VIEW OtherInterests (userid, type)
AS
SELECT *
FROM likes
```

```
WHERE type IN
(
SELECT type
FROM interests
WHERE category = 'others'
)
;
```

CREATE VIEW

```
SELECT *
FROM OtherInterests
WHERE userid < 12
;
```

```
 userid |  type
--------+---------
      9 | reading
      2 | reading
      0 | reading
     11 | dancing
      5 | dancing
      3 | dancing
      7 | drawing
      4 | drawing
```

```
UPDATE OtherInterests
SET type = 'football'
WHERE userid < 20
;
```

UPDATE 12

This view represents all the chats that have been marked as inactive.

```
CREATE VIEW InactiveChats
AS
SELECT *
FROM chats
WHERE is_active = 'false'
```

;

CREATE VIEW

```
SELECT *
FROM InactiveChats
WHERE date_time_created > '2019-11-01'
;
```

```
 chatid | date_time_created | is_active | requestid
--------+-------------------+-----------+-----------
     37 | 2019-12-16        | f         |       47
    119 | 2019-11-22        | f         |      160
    233 | 2019-11-12        | f         |      305
    363 | 2019-12-14        | f         |      482
(4 rows)
```

```
UPDATE InactiveChats
SET is_active = 't'
WHERE date_time_created > '2019-12-01'
;
```

UPDATE 2

Both views in this case were updatable because both contain a one-to-one relationship between the columns in the view and the columns in the underlying table. More specifically, the columns of the view have the same name and type as the underlying table.

8)

```
INSERT INTO Users VALUES
    ('505', 'W', '1996-01-02', 'Melissa', 'melissa123@hotmail.com',
'123abc', '2020-01-13', 1, TRUE, TRUE, 'Montreal'); /* Not valid gender */

INSERT INTO Users VALUES
    ('505', 'F', '1996-01-02', 'Melissa', 'melissa123ahotmail.com',
'123abc', '2020-01-13', 1, TRUE, TRUE, 'Montreal'); /* Not valid email */
```

```
INSERT INTO Users VALUES
    ('505', 'F', '1996-01-02', 'Melissa', 'melissa123@hotmail.com',
'123abc', '2020-01-13', 100, TRUE, TRUE, 'Montreal'); /* Not valid photo
id */

INSERT INTO Users VALUES
    ('505', 'F', '1996-01-02', 'Melissa', 'melissa123@hotmail.com',
'123abc', '2020-01-13', -1, TRUE, TRUE, 'Montreal'); /* Not valid photo id
*/

INSERT INTO relates VALUES
    ('Soccer', 'Soccer', 11); /* Not valid strength value */

INSERT INTO relates VALUES
    ('Soccer', 'Soccer', -1); /* Not valid strength value */
```

```
psql:checkConstraints.sql:7: ERROR:  new row for relation "users" violates check constraint
"gendercheck"
DETAIL:  Failing row contains (505, W, 1996-01-02, Melissa, melissa123@hotmail.com,
123abc, 2020-01-13, 1, t, t, Montreal).
psql:checkConstraints.sql:10: ERROR:  new row for relation "users" violates check constraint
"emailcheck"
DETAIL:  Failing row contains (505, F, 1996-01-02, Melissa, melissa123ahotmail.com, 123abc,
2020-01-13, 1, t, t, Montreal).
psql:checkConstraints.sql:13: ERROR:  new row for relation "users" violates check constraint
"photochoice"
DETAIL:  Failing row contains (505, F, 1996-01-02, Melissa, melissa123@hotmail.com, 123abc,
2020-01-13, 100, t, t, Montreal).
psql:checkConstraints.sql:16: ERROR:  new row for relation "users" violates check constraint
"photochoice"
DETAIL:  Failing row contains (505, F, 1996-01-02, Melissa, melissa123@hotmail.com, 123abc,
2020-01-13, -1, t, t, Montreal).
psql:checkConstraints.sql:19: ERROR:  new row for relation "relates" violates check constraint
"min_max_str"
DETAIL:  Failing row contains (Soccer, Soccer, 11).
psql:checkConstraints.sql:22: ERROR:  new row for relation "relates" violates check constraint
"min_max_str"
DETAIL:  Failing row contains (Soccer, Soccer, -1).
```

9)

The data for our project was automatically generated using java programming. We use random number generation along with a bank of information such as names, topics and interests in order to produce more realistic data. The output displayed in question 4 is the result of the automatic data generation system.