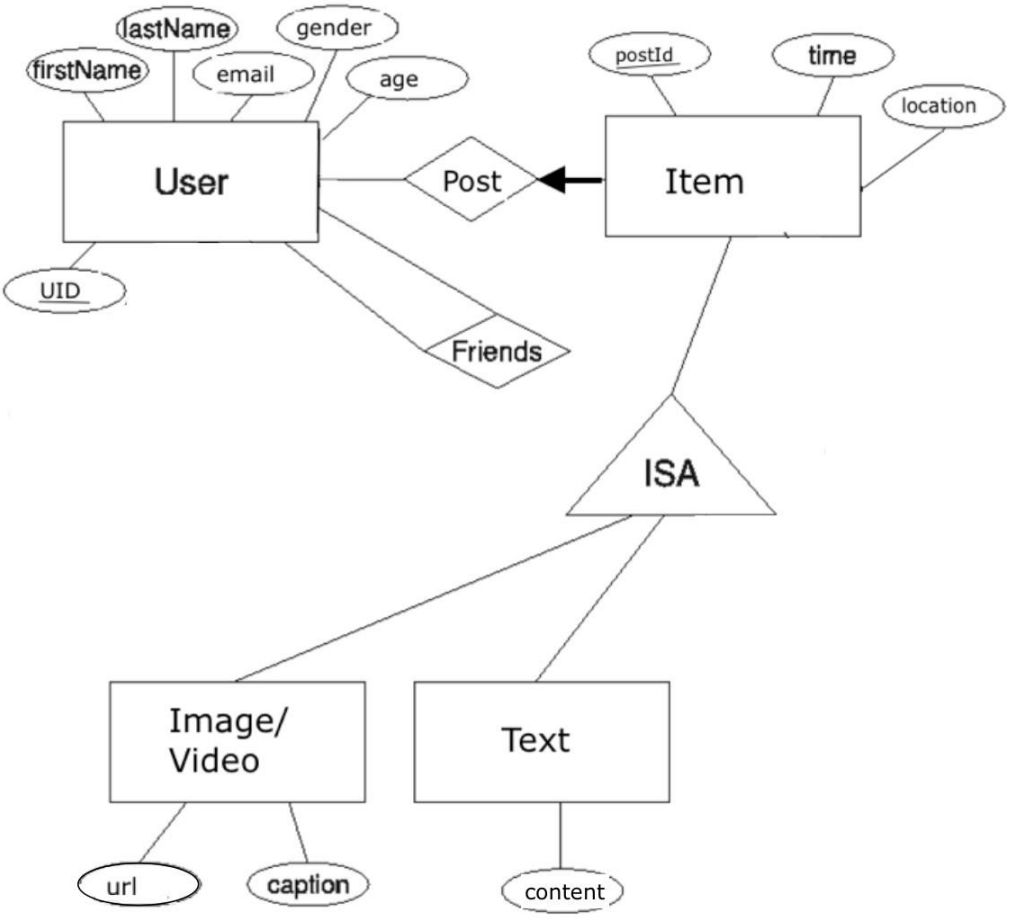
Kayvaun Khoshkhou (920357344)

Homework2
CSC 675/CSC 775

Homework 2 will focus on problems from chapters 3, 4 and 5 of the book.

Q1 (24 points) Consider the following ER diagram for an apartment rental application.

Write the SQL-DDL statements, to create tables for the following ER diagram and capture as many of the constraints as possible. Specify your decisions with the foreign key constraint.



Kayvaun Khoshkhou (920357344)

```
CREATE TABLE User(
UID int primary key,
firstName varchar(50),
lastName varchar(50),
email varchar(50),
gender varchar(10),
age int
 );

 CREATE TABLE ItemPost (
postId int primary key,
time time,
location varchar(50),
UID int,
Foreign key (UID) references User (UID)
 );

CREATE TABLE friends (
UID int,
Fid int,
Primary key (UID, fID)
Foreign key (UID) references User (UID)
Foreign key (fID) references User (UID)
);

CREATE TABLE Texts(
postId int primary key,
content varchar(500),
foreign key (postId) references ItemPost(postId)
 );

CREATE TABLE ImageVideo(
postId int primary key,
url varchar(50),
caption varchar(100),
foreign key(postId) references ItemPost(postId)
 );
```

Kayvaun Khoshkhou (920357344)

Q2. (10 points, 2 points each) Assume we have the following instance of table sailors:

sid	*sname*	*rating*	*age*
18	jones	3	30.0
41	jonah	6	56.0
22	ahab	7	44.0
63	moby	*null*	15.0

a. What is the result of this query:

SELECT AVG (S.rating)
FROM Sailors S

3+6+7 / 4 = 4

b. What is the result of this query:

SELECT SUM (S.rating)
FROM Sailors S

3+6+7 = 16

c. What is the result of this query:

SELECT COUNT (S.rating)
FROM Sailors S

d. Show the left outer join of S with itself, with the join condition being sid=sid.

e. Show the right outer join of S with itself, with the join condition being sid=sid.

Kayvaun Khoshkhou (920357344)

Q3. (6 points)

a. Which one of the following queries finds sailors who have reserved at least a boat but not a red boat?

Query1:

```
SELECT  R.sid
FROM    Boats B, Reserves R
WHERE  B.bid=R.bid   AND   B.color<>'red'
```

Query2:

```
SELECT R.sid
FROM   Reserves R
EXCEPT
SELECT R.sid
FROM   Boats B, Reserves R
WHERE  B.bid=R.bid   AND   B.color='red'
```

b. What are the results of query1 and query2 in part A, considering the following instances of Boats and Reserves tables?

SID	BID	DAY
1	101	10-OCT-17
1	104	10-OCT-19
3	101	10-JUL-19
3	102	10-OCT-18
3	103	07-NOV-17

BID	BNAME	COLOR
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Kayvaun Khoshkhou (920357344)

Q4. (24 points, 6 points each) Consider boat reservation database. Answer these questions using SQL.

a. Find the names of sailors with a higher rating than all sailors who are younger than 20 .

SELECT sname FROM s s1 WHERE NOT EXISTS (SELECT * FROM s S2 WHERE S2.age < 20 AND s.rating <= S2.rating)

b. Find name of sailors who have not reserved a boat whose name includes the string "Marine".

SELECT sname FROM s s1 WHERE sid NOT IN (SELECT sid FROM r, s WHERE r.sid = s.sid AND sname LIKE '%Marine%')

c. Find the names of sailors who have reserved at least two boats.

SELECT sname FROM s, r r1, r r2 WHERE s.sid = r1.sid AND s.sid = r2.sid AND r1.bid<>r2.bid

d. For each boat reserved by at least 2 sailors older than 20, find the boat id and the average age of such sailors.

SELECT bid, AVG(age) FROM b, r, s WHERE s.age > 20 AND b.bid = r.bid AND s.sid = r.sid GROUP BY bid HAVING 2 <= COUNT(DISTINCT s.sid)

Kayvaun Khoshkhou (920357344)

Q5. (24 points, 6 points each) Consider the following schema:
Suppliers(sid: integer , sname: string , address: string)
Parts(pid: integer , pname: string , color: string)
Catalog(sid: integer , pid: integer , cost: real)

The Catalog relation lists the prices charged for parts by Suppliers. Write the following queries in SQL:

a. Find the name s of suppliers who supply every red part.

b. Find the sid s of suppliers who supply only red parts.

c. Find the name s of parts supplied by “Acme Suppliers” and no one else.

d. Find name of suppliers who charge for some parts less than the average price of that part.

Q6. (12 points, 6 points each) Consider the following relational schema:
STUDENT(SID, SNAME, DEPT)
ENROLL(CID, SID, GRADE, SEMESTER)
COURSE(CID, DEPT)

Write the following SQL queries in Relational Algebra:

A.

```
SELECT      S.SNAME
FROM STUDENT    S,      ENROLL      E
WHERE       S.SID=E.SID  AND   E.CID='CSE565'
```

B.

```
SELECT      *
FROM STUDENT    S
WHERE       NOT   EXISTS       (SELECT      *
                  FROM ENROLL      E
      WHERE       E.SID  =      S.SID  AND   E.GRADE      =       4.0)
```