CMPS 180, Midterm Exam, Winter 2016, Shel Finkelstein

Student Name: ______________________________

Student ID: ______________________________

UCSC Email: ______________________________

Midterm Points

Part	Max Points	Points
I	24	
II	30	
III	18	
IV	30	
Total	102	

If you need additional space, you may answer problems on the blank pages of the exam, but be sure to number problems when you do that.

Please show your **UCSC ID** when you turn in your exam.

Part I: (24 points, 4 each)

The following **TRUE** or **FALSE** questions refer to the Customers(<u>cid</u>, cname, type, level, age) relation, where cid is the primary key. cid and age are INTEGER and cname, type and level are CHAR(20).

Question 1: TRUE or **FALSE:** If Customers has 7 tuples in it, then there will be 49 tuples in the result of the query:

```
SELECT *
FROM Customers, Customers;
```

Answer 1: __________

Question 2: TRUE or **FALSE** : The two queries below are equivalent, meaning that they always have the same answer, no matter what's in the Customers relation.

```
SELECT c.cname
FROM Customers c
WHERE c.age > ALL  (SELECT c2.age
                    FROM Customers c2
                    WHERE c2.type = 'snowboard');

SELECT c.cname
FROM Customers c
WHERE c.age > (SELECT MAX(c2.age)
               FROM Customers c2
               WHERE c2.type = 'snowboard');
```

Answer 2: __________

Question 3: TRUE or **FALSE:** The following is a legal SQL query.

```
SELECT level,  MAX(age), type
FROM Customer
WHERE type <> 'snowboard'
GROUP BY level;
```

Answer 3: __________

Question 4: If an instance of Customers has 1000 tuples in it, and the value of cname is different for every tuple in that instance, then the CREATE statement for Customers must have specified that cname is UNIQUE.

Answer 4: __________

Question 5:

If we execute a transaction using Isolation Level READ COMMITTED, and if we execute the following statement twice during that transaction:

```
SELECT level
FROM Customers
WHERE cid=123;
```

then the value of level in the query's result must be the same both times that the statement is executed in the transaction.

Answer 5: __________

Question 6:

If Customers has only one tuple, and the value of age is NULL for that tuple, then the result of the query:

```
SELECT SUM(age), COUNT(age)
FROM Customers;
```

will be a single tuple that has values (NULL, 0).

Answer 6: __________

Part II: (30 points, 6 each):

Question 7: CREATE statements can specify attributes as PRIMARY KEY, and may also specify attributes as UNIQUE. Explain two differences between PRIMARY KEY and UNIQUE.

Answer 7a:

Answer 7b:

Question 8: If Students is a relation in your database, what would each of the following two statements do? Are their effects different?

```
DELETE FROM Students;

DROP TABLE Students;
```

Answer 8:

Question 9: Explain the meanings of UNION and UNION ALL in SQL. After explaining the meanings, answer the following question: If a tuple appears 12 times in the answer to Q1 and 3 times in the answer to Q2, how many times would it appear in Q1 UNION Q2, and how many times would it appear in Q1 UNION ALL Q2.

Answer 9:

Question 10: Views can be used as if they were tables in most but not all ways. Provide clear examples:

a) where a view can be used as if it was a table and
b) where a view cannot be used as if it's a table.

Answer 10a:

Answer 10b:

Question 11: Provide the Truth Table for the condition "p AND q" in SQL's three-valued logic (TRUE, FALSE, UNKNOWN).

Answer 11:

Part III: (18 points, 6 each):

The multiple choice questions below are about the following relations. The primary key is underlined in each relation.

```
Beers(name, manufacturer)       // For each beer, gives the beer's manufacturer
Sells(bar, beer, price)         // For each bar and beer, gives beer's price at bar
Likes(drinker, beer)            // Indicates that drinker likes a beer
```

What does each SQL statement do? For each question, give the best answer.

Question 12:

```
SELECT b.name
FROM Beers b, Likes lk1, Likes lk2
WHERE b.name=lk1.beer AND b.name=lk2.beer AND lk1.drinker <> lk2.drinker;
```

a) Finds names of all the different beers that at least one drinker likes.
b) Finds names of all the different beers that at least two drinkers like.
c) Finds names of all beers that at least two drinkers like, with duplicates possible.
d) Finds all drinkers that like at least two different beers, with duplicates possible.
e) None of the above

Answer 12: __________

Question 13:

```
SELECT b.name
FROM Beers b
WHERE EXISTS
      ( SELECT *
        FROM Likes lk1, Likes lk2
         WHERE lk1.beer <> lk2.beer );
```

a) Finds all the different beers that at least one drinker likes.
b) Finds all the different beers that at least two drinkers like.
c) Finds all the beers that at least two drinkers like, with duplicates possible.
d) Finds all drinkers that like at least two different beers, with duplicates possible.
e) None of the above

Answer 13: __________

Question 14:

```
SELECT s1.bar
FROM Sells s1
WHERE s1.price < ANY ( SELECT s2.price
                       FROM Sells s2
                       WHERE s2.bar = ‘Cheers’ );
```

a) Finds bars that sell a beer that has a lower price than all the beers sold at the bar ‘Cheers’, with no duplicates.
b) Finds bars that sell a beer that has a lower price than all the beers sold at the bar ‘Cheers’, with duplicates possible.
c) Finds bars that sell a beer that has a lower price than at least one beer sold at the bar ‘Cheers’, with no duplicates.
d) Finds bars that sell a beer that has a lower price than at least one beer sold at the bar ‘Cheers’, with duplicates possible.
e) None of the above

Answer 14: __________

Part IV: (30 points, 10 each):

The familiar relations Persons, Houses and Tenants are shown below, with Primary Keys underlined. Assume that all attribute values are NOT NULL.

Persons (SSN, Name, HouseID, ApartmentNumber, Salary)

Houses (HouseID, HouseAddress, ApartmentCount, Color)

Tenants (HouseID, ApartmentNumber, LeaseTenantSSN, LeaseStartDate, LeaseExpirationDate, Rent, LastRentPaidDate, RentOverdue)

Write SQL queries for each of the following questions. If you want to create and then use views to answer these questions, that's okay, but it's not required.

Question 15: Find the SSN and name for each person whose salary is more than 50000.

Answer 15:

Question 16: Find the name of every person who lives in ApartmentNumber 8 of the House whose address is ‘650 Main Street’. The names in your result should appear in alphabetical order, as in a dictionary. No name should appear more than once in your result.

Answer 16:

Question 17: ApartmentCount tells us how many apartments there are in a house, but some apartments may not have tenants leasing them. The Tenants relation tells us which apartments have lease tenants.

For each house, find the number of apartments in that house that have lease tenants, the LeaseTenantCount. Your result should show the HouseID and LeaseTenantCount for each house. <u>But only include a house in your result if that house has more than 5 lease tenants</u>.

In your result, call the second attribute LeaseTenantCount.

Answer 17: