

**These tables referenced in some midterm questions. (See reverse for additional tables.)**

**For Part I, (Questions 1-5) and Part V (Questions 19-21), here are the scores from the Japanese Baseball League in the Scores Table.**

**Scores**

Team	Day	Opponent	Runs
Dragons	Sunday	Swallows	4
Tigers	Sunday	Bay Stars	9
Carp	Sunday	Giants	2
Swallows	Sunday	Dragons	7
Bay Stars	Sunday	Tigers	2
Giants	Sunday	Carp	4
Dragons	Monday	Carp	6
Tigers	Monday	Bay Stars	5
Carp	Monday	Dragons	3
Swallows	Monday	Giants	0
Bay Stars	Monday	Tigers	7
Giants	Monday	Swallows	5

For Part III (Questions 10-14), here are tables Customers, Activities and Slopes that describe customer participation in winter activities on slopes.

Customers

<u>cid</u>	Cname	level	type	age
36	Cho	Beginner	snowboard	18
34	Luke	Inter	snowboard	25
87	Ice	Advanced	ski	20
39	Paul	Beginner	ski	33

Activities

<u>cid</u>	<u>slope-id</u>	day
36	s3	01/05/09
36	s1	01/06/09
36	s1	01/07/09
87	s2	01/07/09
87	s1	01/07/09
34	s2	01/05/09

Slopes

<u>slope-id</u>	name	color
s1	Mountain Run	blue
s2	Olympic Lady	black
s3	Magic Carpet	green
s4	KT-22	black

**CMPS 182, Midterm Exam, Winter 2015, Shel Finkelstein**

**Student Name:** \_\_\_\_\_

**Student ID:** \_\_\_\_\_

**UCSC Email:** \_\_\_\_\_

**Midterm Points**

<b>Part</b>	<b>Max Points</b>	<b>Points</b>
<b>I</b>	30	
<b>II</b>	16	
<b>III</b>	30	
<b>IV</b>	16	
<b>V</b>	18	
<b>Total</b>	110	

Part I: (30 Points, 6 each)

Questions 1-5 are about recent results of the Japanese Baseball League, with data shown in the Scores table. What is the result of each of the following SQL queries? (Show attribute names at the top in all SQL results.)

Question 1:

```
SELECT Team, Day, Runs
FROM Scores
WHERE Opponent = 'Tigers';
```

Answer 1:

Team	Day	Runs
Bay Stars	Sunday	2
Bay Stars	Monday	7

Question 2:

```
SELECT Team, Opponent
FROM Scores
WHERE Day='Monday'
      AND Runs = 5;
```

Answer 2:

Team	Opponent
Tigers	Bay Stars
Giants	Swallows

Question 3:

```
SELECT Distinct Team
FROM Scores
WHERE Runs > 4;
```

Answer 3:

Team
Tigers
Swallows
Dragons
Bay Stars
Giants

Question 4:

```
SELECT Day, Opponent,
FROM Scores
WHERE Team = 'Dragons';
ORDER BY Opponent;
```

Answer 4:

Day	Opponent
Monday	Carp
Sunday	Swallows

Question 5:

```
SELECT Day, SUM(Runs)
FROM Scores
WHERE Team IN ('Dragons', 'Swallows', 'Giants')
GROUP BY Day;
```

Answer 5:

Day	SUM(Runs)
Sunday	15
Monday	11

Part II: (16 points, 4 each)

Answer the following questions with **YES** or **NO**. (TRUE or FALSE is also okay)

Question 6: Is the following a legal SQL query for a table EMPLOYEES(name, age, salary, department)?

```
SELECT name, MAX(age), department
FROM EMPLOYEES
WHERE salary > 5000
GROUP BY department;
```

Answer 6:

**NO**

Question 7: If you define a View in a database, then the result of that view is then calculated and stored in the database.

Answer 7:

**NO**

Question 8: Are the following two SQL queries always equivalent to each other, assuming that Employees is a table that has a salary attribute that can sometimes be NULL?

```
SELECT COUNT(*)
FROM EMPLOYEES;
```

```
SELECT COUNT(salary)
FROM EMPLOYEES;
```

Answer 8:

**NO**

Question 9: Is the following equality always true when R is a relation and c1 and c1 are conditions?

$\sigma_{c1} (\sigma_{c2} (R) ) = \sigma_{c1 \text{ AND } c2}(R)?$

Answer 9:

**YES**

Part III: (30 points, 6 each):

Questions 10-12 are about the instances of the tables Customers, Slopes and Activities.

Question 10: What is the result of the following SQL query:

```
SELECT *  
FROM Customers  
WHERE level='Beginner';
```

Answer 10:

cid	Cname	level	type	age
36	Cho	Beginner	snowboard	18
39	Paul	Beginner	ski	33

Question 11: What is the result of the following SQL query:

```
SELECT Cname, age  
FROM Customers  
WHERE EXISTS ( SELECT *  
                FROM Activities  
                WHERE Customers.cid=Activities.cid  
                AND Activities.day = '01/07/09' );
```

Answer 11:

Cname	age
Cho	18
Ice	20



Question 12: What is the result of the following SQL query:

```
SELECT Customers.Cname, Customers.level, Slopes.name
FROM Customers, Activities, Slopes
WHERE Customers.cid = Activities.cid
      AND Slopes.slope-id = Activities.slope-id
      AND Customers.age < 21;
```

Answer 12:

Customers.Cname	Customers.level	Slopes.name
Cho	Beginner	Magic Carpet
Cho	Beginner	Mountain Run
Cho	Beginner	Mountain Run
Ice	Advanced	Olympic Lady
Ice	Advanced	Mountain Run

For Question 13 and Question 14, write SQL queries that are correct for any instances of the tables. (NULL is not permitted in any column.)

Question 13: Write a SQL query that outputs the age of the oldest person who had an activity on each day that there was an activity. Only the day and the oldest age for that day should be in the result.

Answer 13:

```
SELECT MAX(Customers.age), Activities.Day
FROM Customers, Activities
WHERE Customers.cid = Activities.cid
GROUP BY Activities.Day;
```

There is more than one way of answering this question; this is the simplest.

Question 14: Write a SQL query that outputs the count of the number of different slopes for which the customer named 'Cho' did an activity.

Answer 14:

```
SELECT COUNT(DISTINCT Slopes.slope-id)
FROM Slopes
WHERE EXISTS
( SELECT *
  FROM Customers, Activities
  WHERE Customers.cid = Activities.cid
    AND Activities.slope_id = Slopes.slope-id
    AND Customers.cname = 'Cho' );
```

There is more than one way of answering this question; this is one of the simplest.

Part IV: (16 points, 4 each):

Answer questions 15-18.

Question 15: If  $R(A,B)$  is a relation where A's domain is  $(a_1, a_2, a_3)$  and B's domain is  $(b_1, b_2, b_3, b_4)$ , what the maximum number of different tuples that can be in R, assuming that A and B can't be NULL?

Answer 15: 12

Question 16: Let  $R(A,B,C)$  be a relation where A, B and C can't be NULL, and where A is a key for that relation. Assume that A's domain has 10 different values, B's domain has 3 different values, and C's domain has 4 different values. What is the maximum number of different tuples that can be in R?

Answer 16: 10

Question 17: What does mean for a transaction to have the Atomicity property? Also, explain why Atomicity is important for transactions.

Answer 17:

Atomicity means that all or none of the operations in the transaction are performed, all if the transaction commits, not if it rolls back.

Atomicity is important because a partially complete transaction can produce a surprising and inconsistent database state, such as a transfer from a checking to a savings account that subtracts money from checking but doesn't add it to savings.

Question 18: What is one advantage, and what is one disadvantage, of having an index on a database table? (Provide only one of each.)

Answer 18:

Advantage: Any one of:

- Fast access to database table when query searches on column(s) on which index is defined.
- Particular useful for maintaining uniqueness of primary key or other uniqueness constraint without having to search entire table when data is inserted (or unique columns are updated).

Disadvantage: Any one of:

- Additional update costs since indexes must be updated when table is modified.
- Extra space to store indexes; also extra cache use.
- Added complexity of optimization because use of index should be considered.

Part V: (18 points, 6 each). The following questions relate to the Scores table used in Part I. Assume that Scores table attributes don't allow NULL.

Question 19: For the Scores table instance used in Part I, what is the result of the following SQL query?

```
SELECT S1.Day, S1.Team
FROM Scores S1
WHERE S1.Runs <= ALL
  ( SELECT Runs
    FROM Scores S2
    WHERE S1.Day = S2.Day );
```

Answer 19:

S1.Day	S1.Team
Sunday	Carp
Sunday	Bay Stars
Monday	Swallows

Question 20: What is the result of the following SQL query on the Scores table instance used in Part I?

```
SELECT S1.Team
FROM Scores S1
WHERE S1.Runs >
  ( SELECT Runs
    FROM Scores S2
    WHERE S1.Team = S2.Opponent );
```

Answer 20:

Runtime Error (because subquery returns two values).

Question 21: Write a SQL statement that doubles every Monday score in the Scores table.

Answer 21:

```
UPDATE Scores  
SET Runs = Runs * 2  
WHERE Day = 'Monday';
```