1. Question ID 568: The latest scores from the Japanese Baseball League are in the table with schema

   Scores(Team, Opponent, RunsFor, RunsAgainst)

   The data in this table is as follows:

Team	Opponent	RunsFor	RunsAgainst
Dragons	Tigers	5	3
Carp	Swallows	4	6
Bay Stars	Giants	2	1
Marines	Hawks	5	3
Ham Fighters	Buffaloes	1	6
Lions	Golden Eagles	8	12
Tigers	Dragons	3	5
Swallows	Carp	6	4
Giants	Bay Stars	1	2
Hawks	Marines	3	5
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

   What is the result of executing on this data the query:

```
SELECT Team
FROM Scores
WHERE RunsFor > RunsAgainst AND
    RunsFor <= RunsAgainst + 2
```

   Question Explanation: The query asks for the teams that won their game (RunsFor > RunsAgaint), but not by more than two runs (RunsFor <= RunsAgainst + 2). These teams are the Dragons, Bay Stars, Marines, and Swallows.

2. Question ID 569: The latest scores from the Japanese Baseball League are in the table with schema

   Scores(Team, Opponent, RunsFor, RunsAgainst)

   The data in this table is as follows:

Team	Opponent	RunsFor	RunsAgainst
Dragons	Tigers	5	3
Carp	Swallows	4	6
Bay Stars	Giants	2	1
Marines	Hawks	5	3
Ham Fighters	Buffaloes	1	6
Lions	Golden Eagles	8	12
Tigers	Dragons	3	5
Swallows	Carp	6	4
Giants	Bay Stars	1	2
Hawks	Marines	3	5
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

What is the result of executing on this data the query:

```
SELECT Team AS Winner, RunsFor + RunsAgainst AS Runs
FROM Scores
WHERE RunsFor > RunsAgainst
```

Question Explanation: The query selects only the rows where Team won their game, and renames the Team column to be Winner. The Opponent column is dropped, and the last two columns are summed and renamed Runs.

Winner	Runs
Dragons	8
Swallows	10
Bay Stars	3
Marines	8
Buffaloes	7
Golden Eagles	20

3. Question ID 570: The latest scores from the Japanese Baseball League are in the table with schema

Scores(Team, Opponent, RunsFor, RunsAgainst)

The data in this table is as follows:

Team	Opponent	RunsFor	RunsAgainst
Dragons	Tigers	5	3
Carp	Swallows	4	6
Bay Stars	Giants	2	1
Marines	Hawks	5	3
Ham Fighters	Buffaloes	1	6
Lions	Golden Eagles	8	12
Tigers	Dragons	3	5
Swallows	Carp	6	4
Giants	Bay Stars	1	2
Hawks	Marines	3	5
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

What is the result of executing on this data the query:

```
SELECT *
FROM Scores
WHERE RunsFor > 5
```

Question Explanation: The query selects only the rows where RunsFor is 6 or more. these are rows 6, 8, 11, and 12. Since there is a * in the SELECT clause, all columns are printed. Here is the result.

Team	Opponent	RunsFor	RunsAgainst
Lions	Golden Eagles	8	12
Swallows	Carp	6	4
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

4. Question ID 571: The latest scores from the Japanese Baseball League are in the table with schema

Scores(Team, Opponent, RunsFor, RunsAgainst)

The data in this table is as follows:

Team	Opponent	RunsFor	RunsAgainst
Dragons	Tigers	5	3
Carp	Swallows	4	6
Bay Stars	Giants	2	1
Marines	Hawks	5	3
Ham Fighters	Buffaloes	1	6
Lions	Golden Eagles	8	12
Tigers	Dragons	3	5
Swallows	Carp	6	4
Giants	Bay Stars	1	2
Hawks	Marines	3	5
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

What is the result of executing on this data the query:

```
SELECT Team, Opponent
FROM Scores
WHERE Team LIKE '% %' OR
    Opponent LIKE '_i%'
```

Question Explanation: The query asks for the first two columns of a row, whenever one or both of the following two conditions are met:

1. The Team in that row has a blank in its name (i.e., the Bay Stars, Ham Fighters, and Golden Eagles are the only teams with a blank in their name). This condition selects rows 3, 5, and 12.
2. The second letter in the Opponent name is "i" (i.e., the Giants, Lions, and Tigers). This condition selects rows 1, 3, and 12. The union of these rows is 1, 3, 5, and 12.

The resulting output is:

Team	Opponent
Dragons	Tigers
Bay Stars	Giants
Ham Fighters	Buffaloes

Golden Eagles	Lions

5. Question ID 585: Suppose relations R(A,B) and S(B,C,D) have the tuples shown below:

R =

A	B
1	2
3	4
5	6

S =

B	C	D
2	4	6
4	6	8
4	7	9

Compute the result of the join query:

```
SELECT A, R.B, C, D
FROM R, S
WHERE R.B = S.B
```

Question Explanation: This query is a standard join, in which the columns from R and S that have the same name (B) are equated. Tuple (1,2) from R(A,B) matches (2,4,6) from S(B,C,D), since they both have 2 in their B attributes. The resulting tuple, with schema (A,B,C,D), is (1,2,4,6). Similarly, (3,4) from S(A,B) matches both (4,6,8) and (4,7,9) from R(B,C,D), and yields tuples (3,4,6,8) and (3,4,7,9) for the result. Tuple (5,6) from R(A,B) matches nothing from S(B,C,D), so there are no more tuples in the result.

6. Question ID 586: Suppose relations R(A,B) and S(B,C,D) have the tuples shown below:

R =

A	B
1	2
3	4
5	6

S =

B	C	D
2	4	6
4	6	8
4	7	9

Compute the result of the join query:

```
SELECT A, R.B, S.B, C, D
FROM R, S
WHERE R.A < S.C AND R.B < S.D
```

Question Explanation: Consider tuple (1,2) from R(A,B). In order for a tuple (b,c,d) from S(B,C,D) to satisfy the condition R.A < S.C AND R.B < S.D, we must have 1 < c AND 2 < d. Each of the three tuples

from S satisfy that condition. Thus, we get for the resulting join tuples (1,2,2,4,6), (1,2,4,6,8), and (1,2,4,7,9).

Now consider tuple (3,4) from R. This tuple joins with tuples (b,c,d) from S that satisfy 3 < c AND 4 < d. Again all three tuples from S satisfy this condition and yield three more tuples for the result: (3,4,2,4,6), (3,4,4,6,8), and (3,4,4,7,9).

Finally, consider (5,6) from R. The condition for a successful join is 5 < c AND 6 < d. Tuple (2,4,6) from S(B,C,D) does not satisfy the condition; both 5 < 4 and 6 < 6 are false, in fact. However, (4,6,8) and (4,7,9) do satisfy the condition, yielding two more tuples for the result: (5,6,4,6,8) and (5,6,4,7,9).

7. Question ID 587: Here are three relations, R(a,b), S(a,b), and T(a,b). Their current values are:

R

a	b
0	0
0	1
1	0
1	1

S

a	b
0	0
0	1
1	0
1	1

T

a	b
0	0
0	1
1	0
1	1

Compute the result of the query:

```
SELECT R.a, R.b, S.b, T.b
FROM R, S, T
WHERE R.b = S.a AND S.b <> T.b
```

Question Explanation: The product of R, S, and T contains 64 tuples --- all sequences of six 0's and 1's. The WHERE condition R.b = S.a forces the second and third components to be the same, thus eliminating half the tuples and leaving 32. The condition S.b <> T.b eliminates half the remaining --- those that do not differ in the 4th and 6th components. The result is the 16 tuples of the form (w,x,x,y,z,y'), where *y*' is the complement of *y*, i.e., one is 0 and the other is 1.

The SELECT clause produces from each of these 16 tuples a row of the result: (w,x,y,y'). Since *z* can be either 0 or 1, each of these eight tuples is produced twice.

8. Question ID 591: The latest scores from the Japanese Baseball League are in the table with schema

Scores(Team, Opponent, RunsFor, RunsAgainst)

The data in this table is as follows:

Team	Opponent	RunsFor	RunsAgainst
Dragons	Tigers	5	3
Carp	Swallows	4	6
Bay Stars	Giants	2	1
Marines	Hawks	5	3
Ham Fighters	Buffaloes	1	6
Lions	Golden Eagles	8	12
Tigers	Dragons	3	5
Swallows	Carp	6	4
Giants	Bay Stars	1	2
Hawks	Marines	3	5
Buffaloes	Ham Fighters	6	1
Golden Eagles	Lions	12	8

What is the result of executing on this data the query:

```
SELECT S1.Team, S2.Team
FROM Scores S1, Scores S2
WHERE S1.Team < S2.Team AND
    (S1.RunsFor = S2.RunsFor
     OR S1.RunsAgainst = S2.RunsAgainst)
```

Remember that when strings are compared, "<" means "precedes in alphabetical order.

Question Explanation: The query asks for pairs of teams that either scored the same number of runs or had the same number scored against them, or both. The output must list the teams with the first preceding the second alphabetically.

For example, the Dragons and Marines scored the same number of runs --- 5. They also had the same number scored

against them, but that is not necessary in order for (Dragons, Marines) to be part of the output. Notice that we must list the Dragons first, because "D" precedes "M" alphabetically. The entire output is shown below. Notice that the first two rows are pairs of teams with the same number of runs both for and against. The next two rows are teams that scored the same number of runs but had different numbers scored against them, and the last two rows are teams that scored different numbers of runs, but had the same number scored against them.

S1.Team	S2.team
Dragons	Marines
Hawks	Tigers
Giants	Ham Fighters
Buffaloes	Swallows
Carp	Ham Fighters
Bay Stars	Buffaloes