Quiz #2

Variant 2

Part 1 (5 points)

Please answer the following MCQ:

1. Suppose relation R(A,C), S(B,C,D) has the following tuples:

|  |  |
| --- | --- |
| **A** | **C** |
| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| **B** | **C** | **D** |
| 2 | 4 | 6 |
| 4 | 6 | 8 |
| 4 | 7 | 9 |

Compute the natural join of R and S. Which of the following tuples is in the result? Assume each tuple has schema (A,B,C,D).

1. (3,4,6,9)



1. (5,6,4,6)



1. (1,2,4,6)



1. (5,6,7,8)



2. Suppose relation R(A,B,C) has the following tuples:

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **C** |
| 1 | 2 | 3 |
| 4 | 2 | 3 |
| 4 | 5 | 6 |
| 2 | 5 | 3 |
| 1 | 2 | 6 |

and relation S(A,B,C) has the following tuples:

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **C** |
| 2 | 5 | 3 |
| 2 | 5 | 4 |
| 4 | 5 | 6 |
| 1 | 2 | 3 |

Compute the intersection of the relations R and S. Which of the following tuples is in the result?

1. (2,2,6)



1. (2,5,4)



1. (4,2,3)



1. (2,5,3)



3. Consider the following query:

Select \* From Student, Apply, College

Where Student.sID = Apply.sID and Apply.cName = College.cName

And Student.GPA > 1.5 And College.cName < 'Cornell'

Suppose we are allowed to create two indexes, and assume all indexes are tree-based. Which two indexes do you think would be most useful for speeding up query execution?

1. Student.sID, College.cName



1. Student.sID, Student.GPA



1. Apply.cName, College.cName



1. Apply.sID, Student.GPA



4. Consider a database containing two relations

Borrower(customer-name, loan-number)

Loan(loan-number, amount)

We define a view loan-info as

CREATE VIEW *loan-info* as

SELECT *customer-name,amount*

FROM *Borrower, Loan*

WHERE *Borrower.loan-number=Loan.loan-number*

Consider the following insertions

1. INSERT INTO Borrower VALUES ('Johnson', null)

INSERT INTO Loan VALUES (null,1900)

1. INSERT INTO Borrower VALUES ('Johnson', 1209) INSERT INTO Loan VALUES (1209,1900)

Which of the above operations will have the effect of inserting tuple ("Johnson",1900) into *loan-info* (assuming it is not there previously)?

1. I only
2. II only
3. I and II
4. None of the above

5. Consider the following SQL table declaration:

CREATE TABLE R (a INT, b INT, c INT, CHECK( [fill-in] ));

Currently R contains the tuples (1,4,14), (2,3,15), and (3,3,16). Which of the following tuple-based CHECK constraints will cause the following insertion to be rejected?

INSERT INTO R VALUES (4,4,9);

1. b < (SELECT MIN(c) FROM R)



1. b > (SELECT AVG(a) FROM R)



1. c >= (SELECT SUM(b) FROM R)



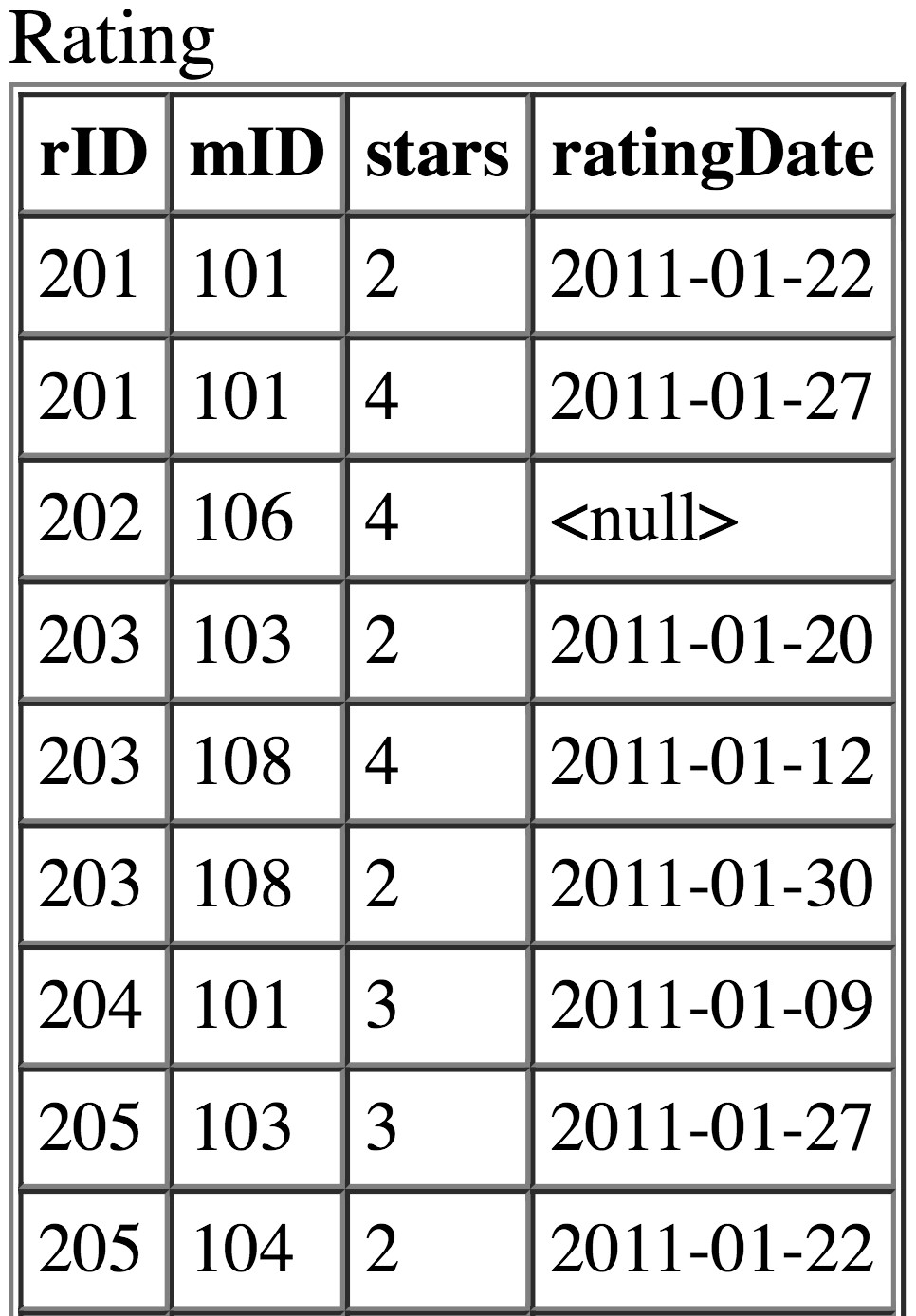
1. a <= ALL (SELECT c - b FROM R)



Part 2 (5 points)

Please write SQL queries for following tasks. Consider following schemas:





1. List movie titles and average ratings, from highest-rated to lowest-rated. If two or more movies have the same average rating, list them in alphabetical order.

SELECT title, AVG(stars) AS average

FROM movie

INNER JOIN Rating USING(mID)

GROUP BY mID

ORDER BY average DESC,title;

1. For each movie that has at least one rating, find the lowest number of stars that movie received. Return the movie title and number of stars. Sort by movie title.

SELECT title,MAX(stars)

FROM movie

INNER JOIN Rating USING(mID)

GROUP BY mID

ORDER BY title;

1. Some reviewers didn't provide a date with their rating. Find the names of all reviewers who have ratings with a NULL value for the date.

SELECT name

From Reviewer

INNER JOIN Rating USING(rID)

WHERE ratingDate IS NULL;

1. Create materialized view for the next statement. Find the titles of all movies that have no ratings.

SELECT title

FROM movie

WHERE mID NOT IN(SELECT mID FROM Rating);

1. Create role with any name and give select and update privileges on the previous view.