NYU, Tandon School of Engineering

October 7, 2024

CS-GY 6083

Principles of Database Systems

Section A, Fall 2024

Homework #2

Submitted by:

Khwaab Thareja

N15911999

Kt3180

1. **Write a tuple relational calculus (TRC) query to find the ID and name of each student in the Comp. Sci. department.**

   Answer: $\{ t \mid \exists\ s \in \text{Student} (t[ID] = s[ID] \wedge t[name] = s[name] \wedge s[dept\_name] = \text{'Comp. Sci.'}) \}$

2. **Write a relational algebra (RA) query to find the ID and name of each student in the Comp. Sci. department**

   Answer: $\Pi_{ID, Name} (\sigma_{dept\_name = \text{'Comp. Sci.'}} (Student))$

3. **Write an SQL query to find the ID and name of each student in the Comp. Sci. department.**

   Answer: select ID, name from student where dept_name='Comp. Sci.' ;

4. **Write a TRC query to find the ID of each instructor who has taught CS-101 along with the year in which they taught it.**

   Answer: $\{ t \mid \exists\ s \in \text{teaches} (t[ID] = s[ID] \wedge t[year] = s[year] \wedge s[course\_id] = \text{'CS-101'}) \}$

5. **Write an RA query to find the ID of each instructor who has taught CS-101 along with the year in which they taught it.**

   Answer: $\Pi_{ID, year} (\sigma_{course\_id = \text{'CS101'}} (teaches))$

6. **Write an SQL query to find the ID of each instructor who has taught CS-101 along with the year in which they taught it.**

   Answer: Select ID, year FROM teaches where course_id='CS-101';

7. **Write a TRC query to find the ID and name of each instructor who has taught CS-101 along with the year in which they taught it.**

   Answer: {r | ∃ i ∈ instructor (r[ID]=i[ID] ∧ r[name]=i[name] ∧ ∃ t ∈ teaches (i[ID]=t[ID] ∧ i[year]=t[year] ∧ t[course_id] = 'CS-101'))}

8. **Write a RA query to find the ID and name of each instructor who has taught CS-101 along with the year in which they taught it**.

   Answer: $\Pi_{instructor.id,\ teaches.year,\ instructor.name}\left(\sigma_{course\_id = 'CS\text{-}101'}\left(instructor \bowtie teaches\right)\right)$

9. **Write an SQL query to find the ID and name of each instructor who has taught CS-101 along with the year in which they taught it.**

   Answer:

```
select i.ID, i.NAME, t.year
FROM instructor i
left join  teaches t  on t.id=i.id
where t.course_id='CS-101' ;
```

10. **Write an SQL query to find the total number of credits the student with ID 12345 has taken in Fall 2009. (Do not worry about whether they have a passing grade for the course.)**

    Answer:

```
select  sum(c.credits) as credits  from takes t
left join course c on t.course_id=c.course_id
where t.ID='12345' and semester='Fall' and year='2009'
```

11. **Write an SQL query to find the ID and total number of credits taken by each student in Fall 2009. (Do not worry about whether they have a passing grade for the course.)**

    Answer:

    select t.ID,sum(c.credits) from takes t

    left join course c on t.course_id=c.course_id

    where semester='Fall' and year='2009'

    GROUP BY t.ID;

12. **Make up another question about the university data, write it in English, and write an SQL query to answer it. It should involve a join of at least two tables.**

    Answer:

    Write an SQL query to find the ID, Name for all the students taught by instructor mozart in spring 2010

    Solution:

    select s.ID, s.name

    from teaches te

    left join instructor i on i.id=te.id

    left join takes t on te.course_id=t.course_id

    left join student s on t.id=s.id

    where te.semester='Spring' and te.year='2010' and i.name='mozart'

13. **Find the productCode, productName and productLine of each product ordered by any customer who lives in the USA that has status “shipped”.**

    Answer:

```
select p.productCode, p.ProductName, productLine
from customers c
left join orders o on o.customerNumber =c.customerNumber
left join orderdetails od on od.orderNumber=o.orderNumber
left join products p on p.productCode=od.productCode
where c.country='USA' and o.status='Shipped';
```

14. **Find the total payments made by each customer who lives in the USA. The result should include the customer’s customerNumber, customerName, and their total payments.**

    Answer:

```
select c.customerNumber, customerName, sum(amount) as total_payments
from customers c
left join payments p on p.customerNumber=c.customerNumber
where c.country='USA'
and p.checknumber is not null
group by c.customerNumber, customerName;
```

15. **For each productCode, list the productCode, productName, and the maximum profit on that product, i.e. the maximum difference between the buyPrice and the priceEach paid for ordered items of that product. You don't need to list products for which there were no orders**

    Answer:

    Select pl.productCode, pl.productName,

    (select od.priceEach-pl.buyPrice) as MaxProfit

    from

    (select productCode,max(priceEach) as priceEach from orderdetails

    group by productCode) od

    left join products pl on pl.productCode=od.productCode;