DBMS: Revision Session(Week 1- 8)

By Piyush Wairale

20

Week10

60%

40% → 9-12

Week 1 → Intro to DBMS

week 2 & 3

week 4

week 5 & 6

50%

Week 7 → Pymm-DB

→ web fundamentals

week 12 → 12.1 & 12.2 → PA GA

slides

week 9-11 → GA PA

Week 8 → PA GA

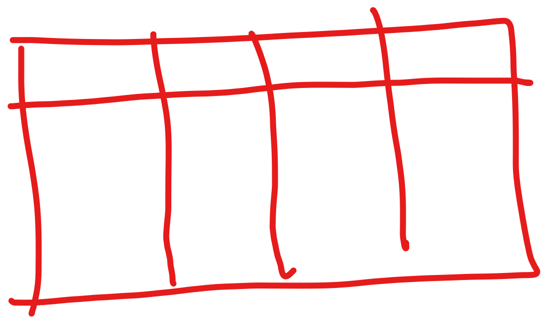
BST

magnetic disk

Week 1

Week 2 & 3

⇒ SQL



→ Create table

CREATE TABle table_name (col_datay consti,

→ Datatype :– Char(n), Varchar(n), Int →, Date → DD/mm/yyyy, Numeric

'RAM'

Char (x) → fixed storage

Varchar (5) → Variable stg

'n' 'RAM_ _' → fixed char length

'RAM'

Numer (5,2) → 100.2 1.20 100.20 1.200 1000.2

Numer(a,b) = _ _ . _ _ a–b . b

+ Constraint

1. unique →
2. NOT NULL →
3. Primary key (Not NULL + unique)
4. Check

5. Foreign key

Check (age>21)

21f1

unique

Unique

~~Student~~ Student (Roll , Name, age)

unique

course (cid , coursename, Name

21f2

duplicate

DBMS	21f1	
JAVA	2f	NULL

8. Execute the following SQL statements in the given order. [MCQ: 2 Points]

Rolln		
2	Asim	21
3	Samir	22

1.
```
CREATE TABLE employee(
roll_no INT PRIMARY KEY,
name VARCHAR(6) UNIQUE,
age INT CHECK(age > 18)
);
```
2. INSERT INTO employee VALUES(1, 'Hemanta', 20); → 7
3. INSERT INTO employee(name, roll_no, age) VALUES('Asim', 2, 21);
4. INSERT INTO employee VALUES(3, 'Samir', 22);
5. INSERT INTO employee(roll_no, name, age) VALUES(4,'Aamir', 18); → violat age con
6. INSERT INTO employee VALUES(5, 'Asim', 25);

What will be the output of the below SQL query?

```
SELECT roll_no from employee;
```

4. Consider the following SQL statement: [MSQ: 2 Points]

```
CREATE TABLE Student(
        Roll_no varchar(8) primary key,
        Name varchar(10),
        Dept_name varchar(10),
        Semester varchar(10),
        check (Semester in ('Fall', 'Winter', 'Summer')));
```

Identify the correct INSERT statement for table **Student**.

Answer: Option [redacted] & Optio[redacted]

- ○ `INSERT INTO Student values('CS101','Rakesh','CS','Winter')`
- ○ `INSERT INTO Student(Roll_no,Name,Dept_name,Semester) values('CS102','Ram','CS','Summer')`
- ○ `INSERT INTO Student(Roll_no,Name,Dept_name,Semester) values('CS104','Shyam','CS','Spring')`
- ○ `INSERT INTO Student('CS106','Mohan','CS','Winter')`

syntax error

8. Consider the SQL query shown below.

```
create table person (
        ID char(10) primary key,
        name char(40) NOT NULL,
        father char(10),
        foreign key father references person(ID)
        );
```

Student

ID	Nam	age
21f1	Ram	
21f2	Raj	

cour

cw	cid	Ral
		21f3

With respect to the above query, which constraint does the table person violate?

21F3

ID	name	father
P001	Rudra	NULL
P002	Advik	P001
P003	Raghab	P006
P004	Krishna	P002
P005	Rudra	P004

Table 4: person

- ◯ NOT NULL
- ◯ PRIMARY KEY
- ◯ FOREIGN KEY
- ◯ None of the above

13. Consider the following relational schemas. [MCQ: 3 Points]

employee(<u>*emp_id*</u>, *emp_name*, *dob*, *dept_id*, *desg_id*)
department(<u>*dept_id*</u>, *dept_name*)
designation(<u>*desg_id*</u>, *desg_name*, *salary*)

Choose the correct options to fill in the blanks of the given query so that it returns the highest salary in 'Computer Science' department.

Order of execution

```
SELECT ___A___(de.salary)
FROM employee AS e
INNER JOIN department AS d ON e.dept_id = d.dept_id
INNER JOIN designation AS de ON e.desg_id = de.desg_id
___B___ BY d.dept_name
___C___ d.dept_name = 'Computer Science'
```

FROM(Joins)
where
Group By
HAVING
SELECT
Order By
limit

Answer: Op[redacted]

- ◯ A:MAX, B:GROUP, C:WHERE
- ◯ A:MAX, B:GROUP, C:HAVING
- ◯ A:MAX, B:ORDER, C:WHERE
- ◯ A:MAX, B:ORDER, C:HAVING

10. Let **ABC** (a, b, c) and **BQR** (b, q, r) be two relations with instances shown below:

a	b	c
2	3	5
4	3	1
2	4	1
1	1	2
5	2	5

Table 1: ***ABC***

b	q	r
1	4	2
2	3	4
5	1	2
2	7	1

Table 1: ***BQR***

What will be the number of tuples fetched by the given relational algebra operation?
$\Pi_{q,r}(\sigma_{c>a}(ABC \bowtie BQR))$

4. Consider the table **player** and predict the output of the query that follows.

player_id	name	jersey_no
P001	Rudra	10
P002	Advik	20
P003	Raghab	30
P004	Krishna	40
P005	Rudra	80

Table 3: **player**

```
SELECT COUNT(name)
FROM (( SELECT player_id, name
        FROM player) as P
        NATURAL JOIN ( SELECT name, jersey_no
                       FROM player) as J)
```

3. Consider Table 1 and predict the output of the query that follows.
[NAT: 3 Points]

id	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000.00
12121	Wu	Finance	90000.00
15151	Mozart	Music	40000.00
22222	Einstein	Physics	95000.00
32343	El Said	History	60000.00
33456	Gold	Physics	87000.00
45565	Katz	Comp. Sci.	75000.00
58583	Califieri	History	62000.00
76543	Singh	Finance	80000.00
76766	Crick	Biology	72000.00
83821	Brandt	Comp. Sci.	92000.00
98345	Kim	Elec. Eng.	80000.00

Table 1: **instructor**

```
SELECT COUNT(*) FROM instructor AS a
WHERE a.salary > SOME(SELECT b.salary
                 FROM instructor AS b
                 where b.dept_name='Biology')
AND a.salary  >  ALL(SELECT c.salary
                 FROM instructor AS c
                 WHERE c.dept_name='Accountancy');
```

Answer: 7

5. Consider the following relations as shown in Table 2 and Table 3 [NAT: 4 Points]

shop_no	name
SH01	Tea stall
SH02	Modern Store
SH03	Balaji Store
SH04	Modern Store

Table 2: **Shop**

shop_no	item_name	price
SH01	Sugar	200
SH01	Tea leaf	500
SH02	Cookies	800
SH02	Namkeen	400
SH03	Mustard oil	700
SH04	Cookies	500

Table 3: **Shop_order**

```
SELECT name,AVG(price)
FROM Shop
NATURAL JOIN
Shop_order
GROUP BY name
HAVING AVG(price)>400
```

The number of tuples returned by the above SQL query.

Answer: 2

Week 4

15. Consider the two relational schemas **Faculty**(*f_id*, *name*, *dept_name*) and **Student**(*s_id*, *name*, *dept_name*) as shown in the Figure 4. [MCQ: 4 Points]

f_id	name	dept_name
F001	Marry	Biology
F003	Abhi	Zoology
F007	Harry	Physics
F002	Sunil	Biology
F009	Rose	Zoology

s_id	name	dept_name
S001	Shima	Physics
S002	Rose	Zoology
S003	Henry	Zoology
S004	Abhi	Biology
S005	Abhi	Physics

Figure 4: **Faculty and Student**

What will be the total numbers of tuples resulting from the following relational algebra expression?

$$\Pi_{name,dept\_name}(Faculty \bowtie Student)$$

Answer: Option C

- ◯ 3
- ◯ 2
- ◯ 1
- ◯ 4

6. Consider the following relations: [MSQ: 4 Points]

auto_part($\underline{pid}$, $pname$, $color$)
auto_suppliers($\underline{sid}$, $sname$, $location$)
catalog($\underline{pid, sid}$,$price$)

Choose the correct relational algebra expressions to find the suppliers ID (sid) who supply auto parts of both the 'Red' and 'Black' colors.
Answer: Option C

- ○ $\Pi_{sid}(\sigma_{color='Red'}(auto\_part \bowtie catalog)) \wedge \Pi_{sid}(\sigma_{color='Black'}(auto\_part \bowtie catalog))$
- ○ $\Pi_{sid}(\sigma_{color='Red' \wedge color='Black'}(auto\_part \bowtie catalog))$
- ○ $\Pi_{sid}(\sigma_{color='Red'}(auto\_part \bowtie catalog)) \cap \Pi_{sid}(\sigma_{color='Black'}(auto\_part \bowtie catalog))$
- ○ $\Pi_{sid}(\sigma_{color='Red'}(auto\_part \bowtie auto\_suppliers)) \cup \Pi_{sid}(\sigma_{color='Black'}(auto\_part \bowtie auto\_suppliers))$
- ○ $\Pi_{sid}(\sigma_{color='Red'}(auto\_part \bowtie auto\_suppliers)) \cap \Pi_{sid}(\sigma_{color='Black'}(auto\_part \bowtie catalog))$

customer(<u>cid</u>, ~~cname~~, age)
product(<u>pid</u>, pname, price)
order(<u>oid</u>, <u>cid</u>, <u>pid</u>)

TRC

$\{t \mid \exists t \in ty\ (\ \rightarrow\ )\}$

SQL

DRC → week4 → week 4

→ Quiz 2 → Max

1. Find the name of the customer who has ordered the product named 'Belts'.
2. Find the name and price of the product that was ordered by the customer whose name is 'Josh' ~~and~~ OR whose age is greater than 25.

Student ⇔ SQL ⇔ RA ⇔ TRC ⇔ DRC

TRC and DRC expressions

(t.cname = c.cname) ∧ t.pname = p.pname

a. $\{< b >\mid \exists a,b,c(< a,b,c >\in customer) \wedge \exists d,e,f(< d,e,f >\in product \wedge e = \text{`Belts'}) \wedge \exists g,h,i(< g,h,i >\in order \wedge a = h \wedge d = i)\}$

b. $\{p \mid \exists c \in customer\ \exists p \in product\ \exists o \in order(c.cid = o.cid \wedge p.pid = o.pid \wedge p.pname = \text{`Belts'})\}$

c. $\{t \mid \exists c \in customer\ \exists p \in product\ \exists o \in order(t.pname = p.pname \wedge t.price = p.price \wedge c.age > 25 \wedge c.cname = \text{`Josh'})\}$

∧ ∨ < >

d. $\{< e,f >\mid \exists a,b,c(< a,b,c >\in customer \wedge b = \text{`Josh'} \vee c > 25) \wedge \exists d,e,f(< d,e,f >\in product) \wedge \exists g,h,i(< g,h,i >\in order \wedge a = h \wedge d = i)\}$

e. $\{t \mid \exists c \in customer\ \exists p \in product\ \exists o \in order(t.pname = p.pname \wedge t.price = p.price \wedge c.age > 25 \wedge c.cname = \text{`Josh'} \wedge c.cid = o.cid \wedge p.pid = o.pid)\}$