Database Management Systems

Vijaya Saradhi

IIT Guwahati

Fri, 14th Feb 2020

Example Database

	Sailors						
sid sname rating ag							
22	Dustin	7	45.0				
29	Brutus	1	33.0				
31	Lubber	8	55.5				
32	Andy	8	25.5				
58	Rusty	10	35.0				
64	Horatio	7	35.0				
71	Zorba	10	16.0				
74	Horatio	9	35.0				
85	Art	3	25.5				
95	Bob	3	63.5				

	Reserves							
sid	bid	day						
22	101	10-Oct-2019						
22	102	10-Oct-2019						
22	103	08-Oct-2019						
22	104	07-Oct-2019						
31	102	10-Nov-2019						
31	103	06-Nov-2019						
31	104	12-Nov-2019						
64	101	05-Sep-2019						
64	102	08-Sep-2019						
74	103	08-Sep-2019						

	Boats	
bid	bname	color
101	Interlake	blue
102	Interlanke	red
103	Clipper	green
104	Marine	red

Queries

- Q1 Find the names of the Sailors who have reserved Boat 103
- Q2 Find the names of the Sailors who reserved a red boat
- Q3 Find the colors of boats reserved by Lubber
- Q4 Find the names of Sailors who have reserved at least one boat
- Q5 Find the names of Sailors who have reserved a red or a green Boat
- Q6 Find the names of Sailors who have reserved a red AND a green Boat
- Q7 Find the names of Sailors who have reserved at least two boats
- Q8 Find the sids of Sailors with age over 20 who have not reserved a red

3/64

- Q9 Find the names of sailors who have reserved all boats
- Q10 Find the names of sailors who have reserved <u>all</u> boats with name Interlake

Q7: Find the names of Sailors who have reserved at least two boats

 $\rho(Reservations, \pi_{sid,sname,bid}(Sailors \bowtie Reserves))$

Reservations							
	Sailors					erves	
sid	sid sname rating age		sid	bid	day		
22	Dustin	7	45.0	22	101	10-Oct-2019	
29	Brutus	1	33.0	22	102	10-Oct-2019	
31	Lubber	8	55.5	22	103	08-Oct-2019	
32	Andy	8	25.5	22	104	07-Oct-2019	
58	Rusty	10	35.0	31	102	10-Nov-2019	
64	Horatio	7	35.0	31	103	06-Nov-2019	
71	Zorba	10	16.0	31	104	12-Nov-2019	
74	Horatio	9	35.0	64	101	05-Sep-2019	
85	Art	3	25.5	64	102	08-Sep-2019	
95	Bob	3	63.5	74	103	08-Sep-2019	

	Reservations	5
sid	sname	bid
22	Dustin	101
22	Dustin	102
22	Dustin	103
22	Dustin	104
31	Lubber	102
31	Lubber	103
31	Lubber	104
64	Horatio	101
64	Horatio	102
74	Horatio	103

Q7: Find the names of Sailors who have reserved at least two boats

	Reservations			Reservation	s
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

		Two_Res	ervation	s	
sid_1	$sname_1$	bid_1	sid ₂	sname ₂	bid_2
22	Dustin	101	22	Dustin	101
22	Dustin	101	22	Dustin	102
22	Dustin	101	22	Dustin	103
22	Dustin	101	22	Dustin	104
22	Dustin	101	31	Lubber	102
22	Dustin	101	31	Lubber	103
22	Dustin	101	31	Lubber	104
22	Dustin	101	64	Horatio	101
22	Dustin	101	64	Horatio	102
22	Dustin	101	74	Horatio	103

Q7: Find the names of Sailors who have reserved at least two boats

	Reservations			Reservation	s
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

		- -			
		Two_Res		5	
sid_1	$sname_1$	bid_1	sid ₂	$sname_2$	bid_2
22	Dustin	102	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	102	22	Dustin	103
22	Dustin	102	22	Dustin	104
22	Dustin	102	31	Lubber	102
22	Dustin	102	31	Lubber	103
22	Dustin	102	31	Lubber	104
22	Dustin	102	64	Horatio	101
22	Dustin	102	64	Horatio	102
22	Dustin	102	74	Horatio	103

Q7: Find the names of Sailors who have reserved at least two boats

	Reservations			Reservation	s
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

	Two_Reservations						
sid ₁	sname ₁	bid_1	sid ₂	sname ₂	bid ₂		
22	Dustin	103	22	Dustin	101		
22	Dustin	103	22	Dustin	102		
22	Dustin	103	22	Dustin	103		
22	Dustin	103	22	Dustin	104		
22	Dustin	103	31	Lubber	102		
22	Dustin	103	31	Lubber	103		
22	Dustin	103	31	Lubber	104		
22	Dustin	103	64	Horatio	101		
22	Dustin	103	64	Horatio	102		
22	Dustin	103	74	Horatio	103		

Q7: Find the names of Sailors who have reserved at least two boats

	Reservation	s		Reservation	s
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

		Two_Res	ervation	S	
sid_1	$sname_1$	bid_1	sid ₂	sname ₂	bid_2
22	Dustin	104	22	Dustin	101
22	Dustin	104	22	Dustin	102
22	Dustin	104	22	Dustin	103
22	Dustin	104	22	Dustin	104
22	Dustin	104	31	Lubber	102
22	Dustin	104	31	Lubber	103
22	Dustin	104	31	Lubber	104
22	Dustin	104	64	Horatio	101
22	Dustin	104	64	Horatio	102
22	Dustin	104	74	Horatio	103

Q7: Find the names of Sailors who have reserved at least two boats

	Reservations			Reservations		
sid	sname	bid	sid	sname	bid	
22	Dustin	101	22	Dustin	101	
22	Dustin	102	22	Dustin	102	
22	Dustin	103	22	Dustin	103	
22	Dustin	104	22	Dustin	104	
31	Lubber	102	31	Lubber	102	
31	Lubber	103	31	Lubber	103	
31	Lubber	104	31	Lubber	104	
64	Horatio	101	64	Horatio	101	
64	Horatio	102	64	Horatio	102	
74	Horatio	103	74	Horatio	103	

		Two_Res			
				•	
sid_1	$sname_1$	bid_1	sid ₂	$sname_2$	bid_2
31	Lubber	102	22	Dustin	101
31	Lubber	102	22	Dustin	102
31	Lubber	102	22	Dustin	103
31	Lubber	102	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	102	31	Lubber	103
31	Lubber	102	31	Lubber	104
31	Lubber	102	64	Horatio	101
31	Lubber	102	64	Horatio	102
31	Lubber	102	74	Horatio	103

Q7: Find the names of Sailors who have reserved at least two boats

Reservations				Reservation	s
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

		Two_Res	ervations	S	
sid ₁	sname ₁	bid_1	sid ₂	sname ₂	bid ₂
31	Lubber	103	22	Dustin	101
31	Lubber	103	22	Dustin	102
31	Lubber	103	22	Dustin	103
31	Lubber	103	22	Dustin	104
31	Lubber	103	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	103	31	Lubber	104
31	Lubber	103	64	Horatio	101
31	Lubber	103	64	Horatio	102
31	Lubber	103	74	Horatio	103

Q7: Find the names of Sailors who have reserved at least two boats

	Reservation	s		Reservation	s
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

Two_Reservations									
sid ₁	sname ₁	bid_1	sid ₂	sname ₂	bid ₂				
31	Lubber	104	22	Dustin	101				
31	Lubber	104	22	Dustin	102				
31	Lubber	104	22	Dustin	103				
31	Lubber	104	22	Dustin	104				
31	Lubber	104	31	Lubber	102				
31	Lubber	104	31	Lubber	103				
31	Lubber	104	31	Lubber	104				
31	Lubber	104	64	Horatio	101				
31	Lubber	104	64	Horatio	102				
31	Lubber	104	74	Horatio	103				

Q7: Find the names of Sailors who have reserved at least two boats

Reservations				Reservation	s
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

		Two_Res	ervations	5	
sid_1	$sname_1$	bid_1	sid ₂	sname ₂	bid_2
64	Horatio	101	22	Dustin	101
64	Horatio	101	22	Dustin	102
64	Horatio	101	22	Dustin	103
64	Horatio	101	22	Dustin	104
64	Horatio	101	31	Lubber	102
64	Horatio	101	31	Lubber	103
64	Horatio	101	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	101	64	Horatio	102
64	Horatio	101	74	Horatio	103

Q7: Find the names of Sailors who have reserved at least two boats

	Reservations			Reservation	S
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

	Reservations			Reservations		
sid_1	$sname_1$	bid_1	sid ₂	sname ₂	bid_2	
64	Horatio	102	22	Dustin	101	
64	Horatio	102	22	Dustin	102	
64	Horatio	102	22	Dustin	103	
64	Horatio	102	22	Dustin	104	
64	Horatio	102	31	Lubber	102	
64	Horatio	102	31	Lubber	103	
64	Horatio	102	31	Lubber	104	
64	Horatio	102	64	Horatio	101	
64	Horatio	102	64	Horatio	102	
64	Horatio	102	74	Horatio	103	

Q7: Find the names of Sailors who have reserved at least two boats

Reservations			Reservations		
sid	sname	bid	sid	sname	bid
22	Dustin	101	22	Dustin	101
22	Dustin	102	22	Dustin	102
22	Dustin	103	22	Dustin	103
22	Dustin	104	22	Dustin	104
31	Lubber	102	31	Lubber	102
31	Lubber	103	31	Lubber	103
31	Lubber	104	31	Lubber	104
64	Horatio	101	64	Horatio	101
64	Horatio	102	64	Horatio	102
74	Horatio	103	74	Horatio	103

Reservations			Reservations		
sid_1	sname ₁	bid_1	sid ₂	sname ₂	bid ₂
74	Horatio	103	22	Dustin	101
74	Horatio	103	22	Dustin	102
74	Horatio	103	22	Dustin	103
74	Horatio	103	22	Dustin	104
74	Horatio	103	31	Lubber	102
74	Horatio	103	31	Lubber	103
74	Horatio	103	31	Lubber	104
74	Horatio	103	64	Horatio	101
74	Horatio	103	64	Horatio	102
74	Horatio	103	74	Horatio	103

Q8: Find the sids of Sailors with age over 20 who have not reserved a red boat

- Compute Sailors whose age is more than 20 years
- $\pi_{sid}(\sigma_{age>20}(Sailors))$
- Compute Sailors who have reserved red boat
- $\pi_{sid}((\sigma_{color='red'}(Boats)) \bowtie Reserves \bowtie Sailors)$
- Obtain set difference between the above two results
- $\pi_{sid}(\sigma_{age>20}(Sailors)) \pi_{sid}((\sigma_{color='red'}(Boats)) \bowtie Reserves \bowtie Sailors)$

Q9: Find the names of sailors who have reserved all boats

- To answer queries involving all we have to obtain two relations A, B; A having two attributes; B having one attribute
- One relation is: $\pi_{(sid,bid)}(Reserves)$
- Another relation is: $\pi_{(bid)}(Boats)$
- Now we can apply division operator on the above two relations

Q9: Find the names of sailors who have reserved all boats

- $\pi_{(sid,bid)}(Reserves)/\pi_{(bid)}(Boats)$
- $\rho(Tempsids, \pi_{(sid,bid)}(Reserves)/\pi_{(bid)}(Boats))$
- π_{sname} (Tempsids \bowtie Sailors)

Q10: Find the names of sailors who have reserved all boats with name Interlake

- $\pi_{(sid,bid)}(Reserves)/\pi_{(bid)}(\sigma_{bname='Interlake'}(Boats))$
- $\rho(Tempsids, \pi_{(sid,bid)}(Reserves)/\pi_{(bid)}(Boats))$
- π_{sname} (Tempsids \bowtie Sailors)

SQL

Overview

- DDL Subset of SQL support creation, deletion and modification of tables and views
- DML Subset of SQL that allows users to pose queries, insert, delete and modify tuples
- Triggers, Events & Adv. Constraints Performs operations based on actions or time
- Embedded SQL SQL statements can be included in various programming languages such as C, C++, Java, python and/or php

SQL

Overview

Transaction Management Various commands allow user to explicitly control aspects of how a transaction is to be executed

Security provide mechanism to control user's access to tables and views

Programming Constructs such as control statements, loops, exceptions, error handling statements

DDL

Overview

- Used in creating tables that is entities, relations
- Defining domains for each column that is attribute
- Express constraints on tables
- Modify tables
- Modify constraints
- Delete tables, columns within tables and constraints
- User need to have privileges for performing these operations
- Typically database administrator perform these operations
- Database users perform DML

DDL - Create

Permanent table

```
CREATE TABLE student (
    roll_number CHAR(20),
    name CHAR(30),
    login CHAR(20),
    age INT);
```

DDL - Create temporary table

Temporary table

```
CREATE TEMPORARY TABLE student(
    roll_number CHAR(20),
    name CHAR(30),
    login CHAR(20),
    age INT);
```

DDL - Expressing - Keys

Expressing Keys

```
CREATE TABLE student(
    roll_number CHAR(20),
    name CHAR(30),
    login CHAR(20),
    age INT,
    UNIQUE(login),
    PRIMARY KEY(roll_number));
```

```
CREATE TABLE student(
    roll_number CHAR(20),
    name CHAR(30) NOT NULL,
    login CHAR(20),
    age INT NOT NULL,
    UNIQUE(login),
    PRIMARY KEY(roll_number));
```

```
CREATE TABLE course(
    cid CHAR(6),
    title CHAR(20) NOT NULL,
    credits INT NOT NULL,
    PRIMARY KEY(cid));
```

```
CREATE TABLE registers(
    rn CHAR(20),
    course_id CHAR(6),
    PRIMARY KEY(rn, course_id),
    FOREIGN KEY(rn) REFERENCES student(roll_number),
    FOREIGN KEY(course_id) REFERENCES course(cid)
);
```

Specify actions

- What happens when a student row gets deleted?
- What happens when a student row gets updated?
- What happens when a course row gets deleted?
- What happens when a course row gets updated?

```
CREATE TABLE registers (
    rn CHAR(20),
    course_id CHAR(6),
    PRIMARY KEY(rn, course_id),
    FOREIGN KEY(rn) REFERENCES student(roll_number)
    ON DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY(course_id) REFERENCES course(cid)
    ON DELETE CASCADE ON UPDATE CASCADE
);
```

```
CREATE TABLE manager(
    supervisor_id CHAR(20),
    supervisee_id CHAR(20),
    PRIMARY KEY(supervisor_id, supervisee_id),
    FOREIGN KEY(supervisor_id) REFERENCES employee(eid)
    FOREIGN KEY(supervisee_id) REFERENCES employee(eid)
);
```

- A constraint is checked at the end of every SQL statement
- Checks for constraint violations
- SQL statements gets rejected in the case of constraint violations
- Some times this causes inflexibility

Table 1

```
CREATE TABLE student(
    roll_number CHAR(20),
    name CHAR(30),
    login CHAR(20),
    age INT,
    honors CHAR(10) NOT NULL,
    UNIQUE(login),
    PRIMARY KEY(roll_number),
    FOREIGN KEY (honors) REFERENCES courses(cid)
);
```

Table 2

```
CREATE TABLE course(
   cid CHAR(6),
   title CHAR(20) NOT NULL,
   credits INT NOT NULL,
   grader CHAR(20) NOT NULL,
   PRIMARY KEY(cid),
   FOREIGN KEY(grader) REFERENCES student(roll_number)
);
```

Deffer constraint

Disable foreign key checks

```
SET foreign_key_checks = 0;
```

Enable constraint check

Enable foreign key checks

```
SET foreign_key_checks = 1;
```

DDL - Naming Constraint

Naming constraints

- Every constraint can be given a name
- Names are useful in creating, modifying and deleting constraints on tables
- Every constraint is prefixed with syntax CONSTRAINT [symbol] followed by the actual constraint
- In the following example, c1, c2 and c3 are the names given to each
 of the constraint

```
CONSTRAINT c1 UNIQUE(login)
CONSTRAINT c2 PRIMARY KEY(roll_number)
CONSTRAINT c3 FOREIGN KEY (honors) REFERENCES courses(cid)
```

DDL - Creating Indexes

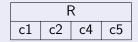
Indexes

- Provide handle on adding indexes to existing tables
- Following example creates an index using the first 10 characters of the name column

```
CREATE INDEX part_of_student_name ON student(name(10));
```

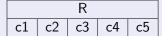
DDL - Adding a column

Altering Table



• Adding a column between c2 and c4

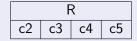
ALTER TABLE R ADD COLUMN c3 INT AFTER c2;



37 / 64

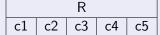
DDL - Adding a column at the beginning

Altering Table



Adding a column c1 at the beginning

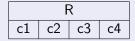
ALTER TABLE R ADD COLUMN c1 INT FIRST: 2



38 / 64

DDL - Adding a column at the end

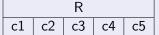
Altering Table



Adding a column c1 at the end

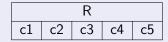
ALTER TABLE R ADD COLUMN c5 INT:

2



DDL - Dropping a column

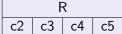
Altering Table



• Dropping the column c1

```
1 ALTER TABLE R DROP COLUMN c1;
```

2



DDL - Adding Constraints

```
Primary Key
CREATE TABLE R(c1 INT, c2 INT, c3 INT, c4 INT);
                              c2 | c3 |
                                       c4

    Adding a primary key c1

    ALTER TABLE R ADD CONSTRAINT my_c1 PRIMARY KEY(c1);
2
```

DDL - Adding Constraints

```
Foreign Key
CREATE TABLE R(c1 INT, c2 INT, c3 INT, c4 INT, PRIMARY KEY(c1
   ));
CREATE TABLE S(s1 INT, s2 INT, PRIMARY KEY(s1);
```

Adding a primary key c2 to R

```
ALTER TABLE R ADD CONSTRAINT my_c2_fkey FOREIGN KEY(c2)
1
       REFERENCES S(s1);
```

DDL - Dropping Constraints

Primary Key

ALATER TABLE R DROP CONSTRAINT my_c1;

Foreign Key

ALTER TABLE R DROP CONSTRAINT my_c2_fkey;

DDL - Changing Domains

Altering Attribute Domains

ALTER TABLE R CHANGE c3 c3 CHAR(20);

ALTER TABLE R CHANGE c3 new_c3 CHAR(20);

One has to be carful while changing the domains when with columns are either primary key or foreign key constraints.

DDL - Default Constraint

Expressing Default Constraint

CREATE TABLE R(c1 INT, c2 INT DEFAULT 441, PRIMARY KEY(c1))

Primary key vs temporal key

Example Schema

- eid and pcn stand for primary key
- Only in the absence of timed attributes
- start_date and end_date are included in the relation
- No employee can have a particular position twice at the same time.
- eid, pcn, start_date, end_date not a primary key

eid	pcn	start_date	end_date
123	900225	01-Jan-1996	01-June-1996
123	900225	01-Apr-1996	01-Oct-1996

Primary key vs temporal key

```
CREATE TABLE Incumbents (eid INT, pcn INT, start_date date,
    end_date date.
   CHECK (
       NOT EXISTS (
           SFLFCT *
           FROM Incumbents as 11
           WHERE 1 <
            (SELECT COUNT(eid)
            FROM Incumbents as 12
            WHERE 11.eid = 12.eid
            AND I1.pcn = I2.pcn
            AND I1.start_date < I2.end_date
            AND | 12.start_date < | 11.end_date )
       AND NOT EXISTS (
           SFLFCT *
           FROM Incumbents AS I1
           WHERE II.eid is null OR II.pcn is null
```

SELECT

Overview

- Consists of SIX clauses
- Combines selection and projection operators
- Optionally the following are specified
 - Extended operations
 - Groupy
 - sort (order by)

SELECT list of attributes

FROM list of tables

WHERE Condition

GROUP BY list of attributes

HAVING CONDITION

ORDER BY list of attributes

Vijava Saradhi (IIT Guwahati)

48 / 64

Algebraic Operators and SQL

Overview

```
\sigma, \pi SELECT, FROM, WHERE
```

× comma separated table list after FROM clause

× table_1 CROSS JOIN table_2

★ table_1 JOIN table_2

Theta Join table_1 JOIN table_2 ON Condition

Re-naming AS: SELECT bname AS boat_name FROM Boats

Algebraic Operators and SQL

Operators

- **UNION**
- INTERSECTION (not available in all DBs)
- EXCEPT (not available in all DBs)

Selection

```
SELECT attr1, attr2, attr3
FROM table1
WHERE attr3 >= 6;
```

Selection

$\sigma_{attr3 \geq 6}(table1)$

table1				
attr1 attr2 attr3				
1	2	5		
3	4	6		
1	2	7		
1	2	7		

```
SELECT
FROM
       table1
WHERE
       attr3 >= 6:
```

Projection

$\pi_{attr1,attr2}(table1)$

table1				
attr1	attr2	attr3		
1	2	5		
3	4	6		
1	2	7		
1	2	7		

```
SELECT
        attr1, attr2
FROM
        table1;
```

Projection

$\pi_{attr3}(table1)$

table1				
attr1 attr2 attr3				
1	2	5		
3	4	6		
1	2	7		
_1	2	7		

```
SELECT
        attr3
FROM
        table1;
```

Selection AND Projection

```
\pi_{attr2}(\sigma_{attr3 \geq 6}(table1))
```

table1				
attr1	attr2	attr3		
1	2	5		
3	4	6		
1	2	7		
1	2	7		

```
SELECT attr2
FROM table1
WHERE attr3 >= 6;
```

Cross Product

$table1 \times table2$

table1		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

ible1 >	< tabl	e2
В	В	D
2	2	3
2	4	5
2	4	5
2	2	3
2	4	5
2	4	5
	B 2 2 2 2 2 2	2 2 2 4 2 4 2 2 2 4

```
SELECT
FROM table1
CROSS JOIN table2;
```

Cross Product

table1		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

ta	ble1 :	< tabl	e2
Α	В	В	D
1	2	2	3
1	2	4	5
1	2	4	5
1	2	2	3
1	2	4	5
1	2	4	5

```
SELECT
FROM
        table1, table2;
```

Cross Product - Projecting out duplicate columns

$\pi_{A,B,D}(table1 \times table2)$

table1		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

tabl	$e1 \times$	table2
Α	В	D
1	2	3
1	2	5
1	2	5
1	2	3
1	2	5
1	2	5

```
SELECT A, table 1.B, D
FROM table1
CROSS JOIN table2;
```

Natural Join

table1		tab	le2
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

```
table1 ⋈ table2
         2
```

```
SELECT
FROM
        table1
JOIN
        table2
ON
        table1.B = table2.B;
```

Natural Join - Projecting out Duplicate Columns

$\pi_{A,table1.B,D}(table1 \bowtie table2)$

table		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

```
SELECT A, table1.B, C
FROM table1

JOIN table2
ON table1.B = table2.B;
```

Natural Join - Projecting out Duplicate Columns

table1		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

tabl	e1 ⋈	table2
Α	В	D
1	2	3
1	2	3

```
SELECT A, table2.B, C
FROM table1
JOIN table2
ON table1.B = table2.B;
```

Theta Join

table1		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

ta	ble1	⊲ tabl	e2
Α	В	В	D
1	2	4	5
1	2	4	5
1	2	4	5
1	2	4	5

```
SELECT
       A, table1.B, C
FROM
       table1
JOIN
       table2
WHERE
       table1.B < table2.B;
```

Natural Join AND Theta Join

table1	×	table2
	table1.B = table2.B&table1.A < table2.D	

table		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

```
SELECT
       A, table1.B, C
FROM
       table1
JOIN table 2
ON table1.B = table2.B
WHERE table1.A < table2.D;
```

Re-naming

 $\rho(RESULT(A1, B1, B2, D1), table1$ table2) table1.B=table2.B&table1.A<table2.D

table		table2	
Α	В	В	D
1	2	2	3
1	2	4	5
		4	5

```
SELECT A AS A1, table 1.B AS B1,
       table 2.B AS B2, D AS D1
FROM
       table1
       table2
JOIN
ON table1.B = table2.B
WHERE table1.A < table2.D;
```

64 / 64