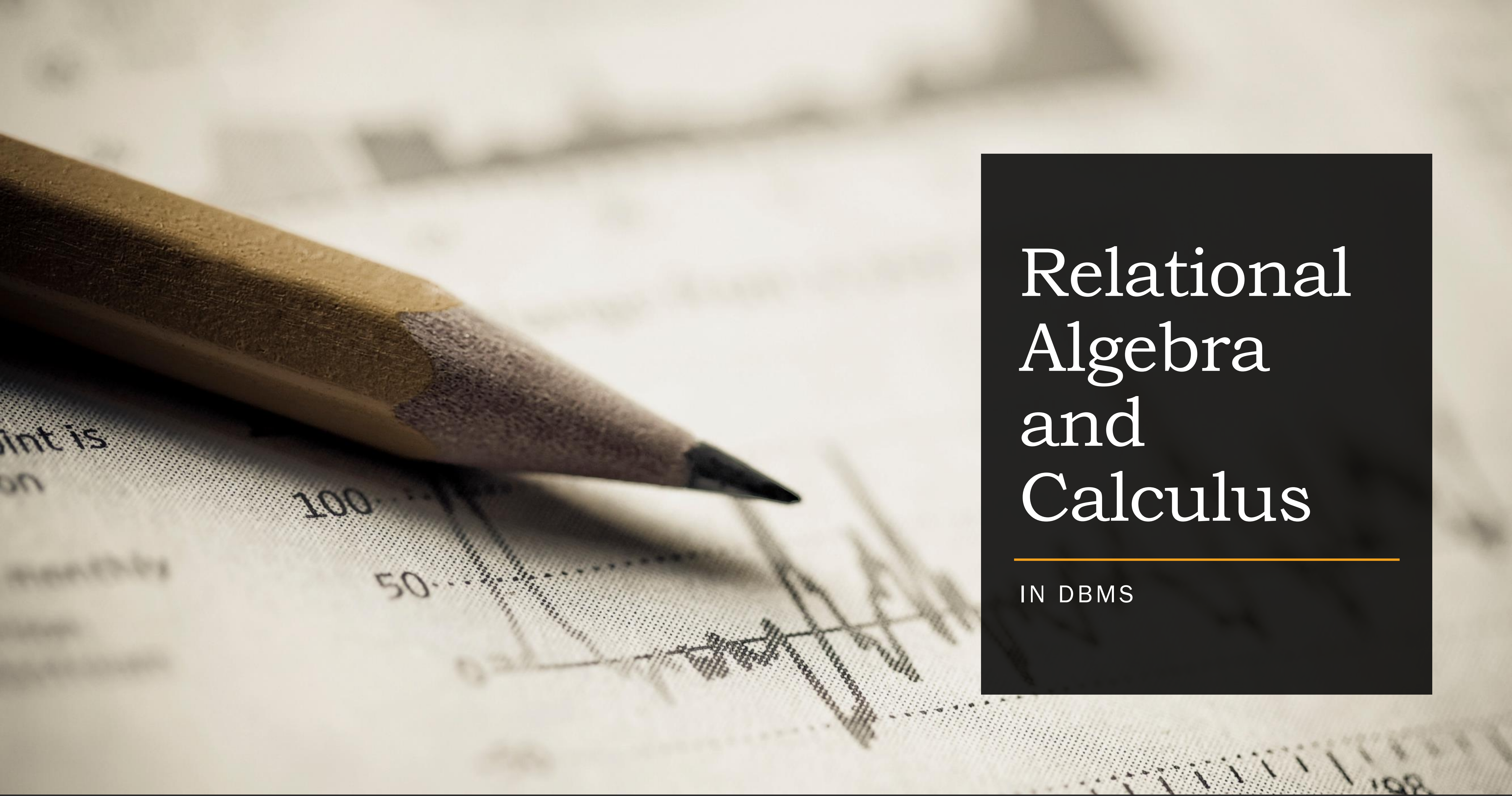


Relational Algebra and Calculus

IN DBMS

Relational Algebra

- A collection of operations to manipulate relations.
- A procedural language
- It specifies the operations to be performed on existing relations to derive result relations.
- It defines the complete scheme for each of the result relations.
- Relational algebraic operations can be divided into
  - Set-oriented operations
  - Relational- oriented operations

Basic operations

UNION (∪)

- ✈ Two relation are union compatible if they have the same arity and one-to-one correspondence of the attributes with the corresponding attributes defined over the same domain.
- ✈ Two relations P(P) and Q(Q) are said to be union compatible of both P and Q are of the same degree ***n*** and the domains of the corresponding ***n*** attributes are identical.

if P= $\{P_1, P_2, \ldots, Pn\}$ and Q= $\{Q_1, Q_2, \ldots, Qn\}$ then

$$Dom(Pi) = Dom(Qi) \;\; for\, i{=}\{1,2,\ldots,n\}$$

Where Dom(Pi) $represents\ the\ domain\ of\ the\ attribute\ Pi$.

- ✈ The result relation R contains tuples that are in either P or Q or in both of them.
- ✈The duplicates are eliminated.
- ✈The degree of the relations P, Q and R is the same.
- ✈The cardinality of the resultant relation depends on the duplication of the tuples in P and Q.

Basic operations

UNION (∪)

Consider two relations

Pi

Id	Name
101	Mahesh
102	Janki
103	Vishal
104	Shruti
108	Ayush
110	Priya

Qi

Id	Name
102	Janki
104	Shruti
107	Jitesh
109	Harris

P ∪Q

Id	Name
101	Mahesh
102	Janki
103	Vishal
104	Shruti
107	Jitesh
108	Ayush
109	Harris
110	Priya

Basic operations

DIFFERENCE (–)

- The difference operation removes the common tuples from the first relation.

R= P – Q

Pi

Id	Name
101	Mahesh
102	Janki
103	Vishal
104	Shruti
108	Ayush
110	Priya

Qi

Id	Name
102	Janki
104	Shruti
107	Jitesh
109	Harris

P – Q

Id	Name
101	Mahesh
103	Vishal
108	Ayush
110	Priya

Basic operations

DIFFERENCE (–)

- The difference operation removes the common tuples from the first relation.

R= P – Q

Pi

Id	Name
101	Mahesh
102	Janki
103	Vishal
104	Shruti
108	Ayush
110	Priya

Qi

Id	Name
102	Janki
104	Shruti
107	Jitesh
109	Harris

P – Q

Id	Name
101	Mahesh
103	Vishal
108	Ayush
110	Priya

Basic operations

INTERSECTION (∩)

- The intersection operation selects the common tuples from the two relations.

R= P ∩ Q

Pi

Id	Name
101	Mahesh
102	Janki
103	Vishal
104	Shruti
108	Ayush
110	Priya

Qi

Id	Name
102	Janki
104	Shruti
107	Jitesh
109	Harris

P ∩ Q

Id	Name
102	Janki
104	Shruti

Basic operations

CARTESIAN PRODUCT (×)

- The cartesian product of two relations is the concatenation of tuples belonging to the two relations.

R= P × Q

- A new resultant relation scheme is created consisting of all possible combinations of the tuples.

Pi

Id	Name
101	Mahesh
102	Janki
103	Vishal
104	Shruti
108	Ayush
110	Priya

Qi

Id	Name
102	Janki
104	Shruti
107	Jitesh
109	Harris

P × Q

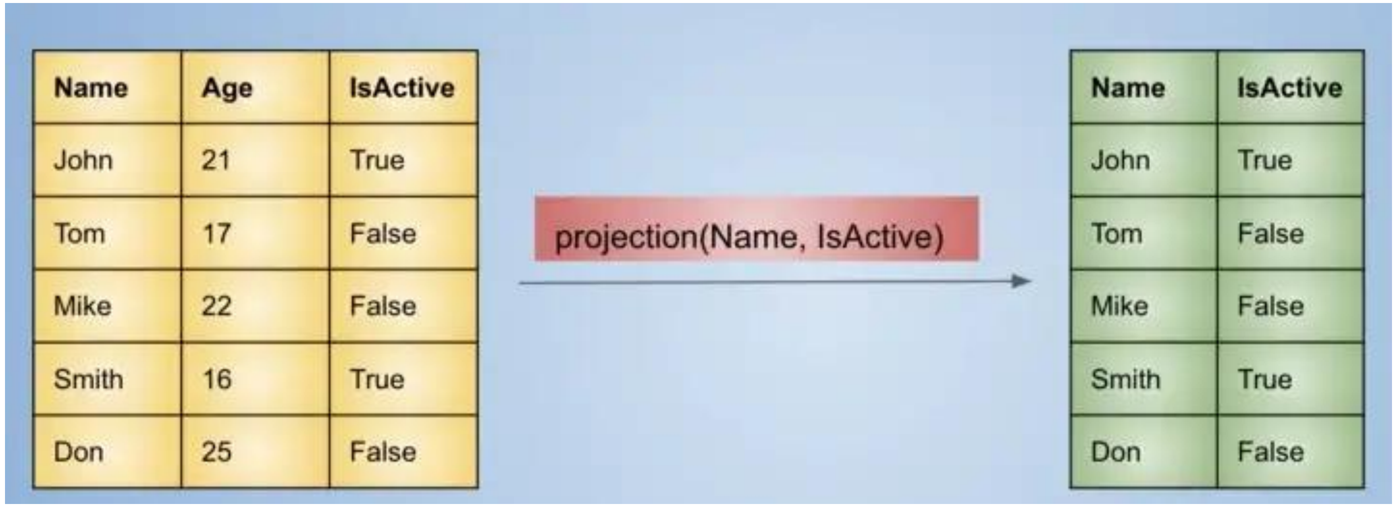
Id	Name
101	Mahesh
102	Janki
103	Vishal
104	Shruti
107	Jitesh
108	Ayush
109	Harris
110	Priya

Basic operations

PROJECTION (Π)

- Projection of all its tuples over some set of attributes (vertical subset of the relation).
- either to reduce the number of attributes or reorder the attributes.

$\Pi_{Name,IsActive}$ (Person)



Name	Age	IsActive
John	21	True
Tom	17	False
Mike	22	False
Smith	16	True
Don	25	False

projection(Name, IsActive)

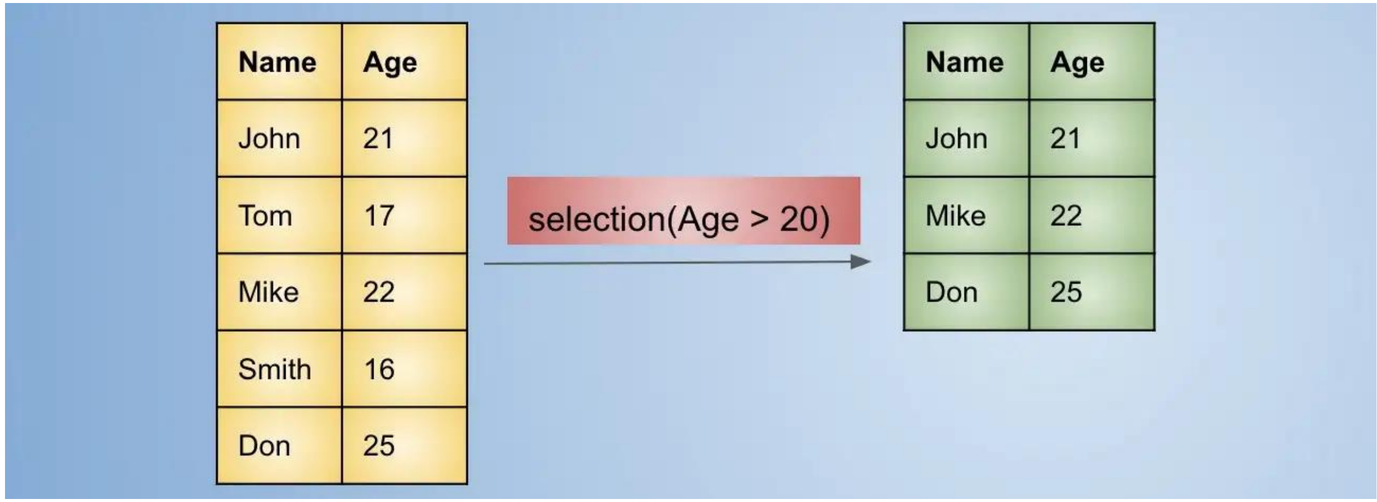
Name	IsActive
John	True
Tom	False
Mike	False
Smith	True
Don	False

Basic operations

SELECTION (σ)

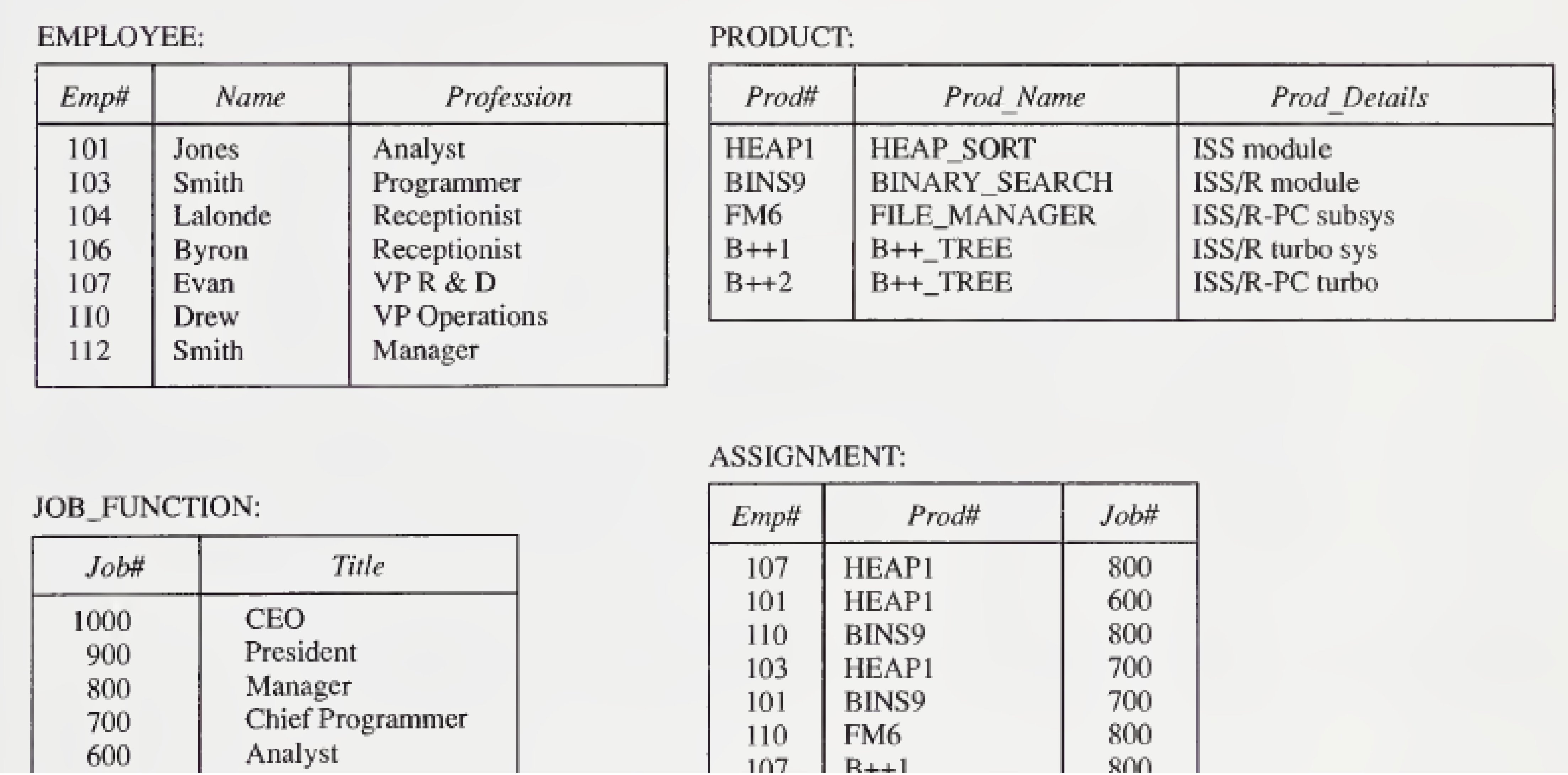
- Selection yields a horizontal subset of the relation.
- The action is defined over a complete set of attribute names but only a subset of the tuples are included in the result.

$\sigma_{Age>10}(Person)$



Name	Age
John	21
Tom	17
Mike	22
Smith	16
Don	25

Name	Age
John	21
Mike	22
Don	25



EMPLOYEE:

Emp#	Name	Profession
101	Jones	Analyst
103	Smith	Programmer
104	Lalonde	Receptionist
106	Byron	Receptionist
107	Evan	VP R & D
110	Drew	VP Operations
112	Smith	Manager

PRODUCT:

Prod#	Prod_Name	Prod_Details
HEAP1	HEAP_SORT	ISS module
BINS9	BINARY_SEARCH	ISS/R module
FM6	FILE_MANAGER	ISS/R-PC subsys
B++1	B++_TREE	ISS/R turbo sys
B++2	B++_TREE	ISS/R-PC turbo

ASSIGNMENT:

Emp#	Prod#	Job#
107	HEAP1	800
101	HEAP1	600
110	BINS9	800
103	HEAP1	700
101	BINS9	700
110	FM6	800
107	B++1	800

JOB_FUNCTION:

Job#	Title
1000	CEO
900	President
800	Manager
700	Chief Programmer
600	Analyst

Basic operations

JOIN (⋈)

✈Combing two relations to form a single relation.

Consider the following relations

Assignment (Emp#,Prod#,job#)

Job_function(Job#, Title)

Get product number of assignments whose development teams have a chief programmer.

1. Compute the cartesian product of ***Assignment*** and ***Job_function*** relations. (name it as TEMP)
2. Select the tuples where title is *chief programmer* and ***Job#*** is same in ***Assignment*** and ***Job_function***.

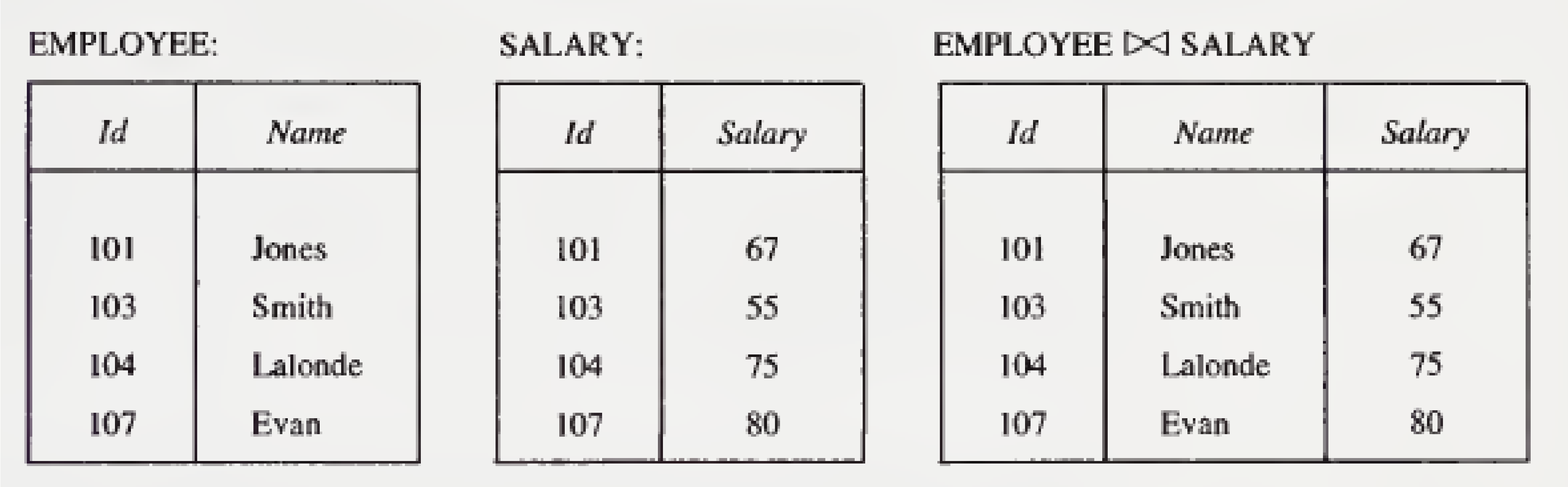
$\text{TEMP} = (\text{Assignment} \times \text{Job\_function})$

$$\Pi_{\text{Prod\#}} (\sigma_{\text{Title="Chief Programmer"} \land \text{ Assignment.Job\# =Job\_Function. Job\#}} (\text{TEMP})$$

Basic operations

The natural join of EMPLOYEE and SALARY relations;

$$\pi_{(Name,Salary)}\ (\text{EMPLOYEE} \bowtie \text{SALARY})$$



EMPLOYEE:

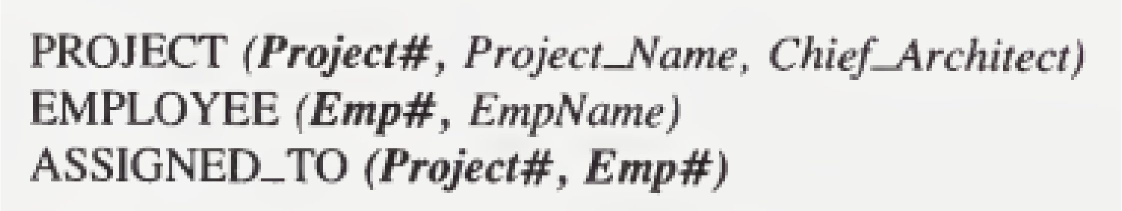
Id	*Name*
101	Jones
103	Smith
104	Lalonde
107	Evan

SALARY:

Id	*Salary*
101	67
103	55
104	75
107	80

EMPLOYEE $\bowtie$ SALARY

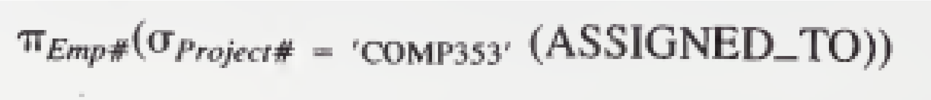
Id	*Name*	*Salary*
101	Jones	67
103	Smith	55
104	Lalonde	75
107	Evan	80



PROJECT (***Project#***, *Project_Name*, *Chief_Architect*)
EMPLOYEE (***Emp#***, *EmpName*)
ASSIGNED_TO (***Project#***, ***Emp#***)

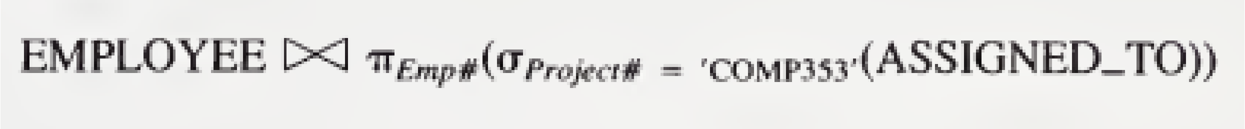
“Get Emp# of employees working on project COMP353.”

- select those tuples of relation ASSIGNED_TO such that the value of the Project# attribute is COMP353.
- then project the result on the attribute Emp# to get the response relation.
- The query and the response relation are shown below



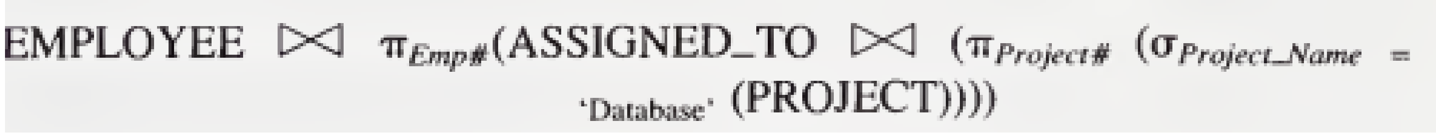
$$\pi_{Emp\#}(\sigma_{Project\# \, = \, \text{'COMP353'}}\ (\text{ASSIGNED\_TO}))$$

“Get details of employees (both number and name) working on project COMP353.”



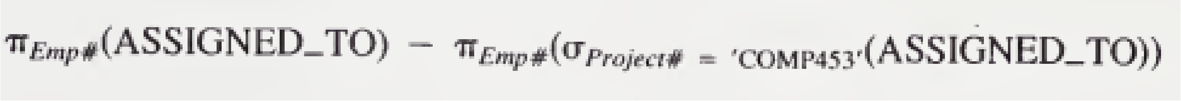
$$\text{EMPLOYEE} \bowtie \pi_{Emp\#}(\sigma_{Project\# \, = \, \text{'COMP353'}}(\text{ASSIGNED\_TO}))$$

“Obtain details of employees working on the Database project.”



$$\text{EMPLOYEE} \bowtie \pi_{Emp\#}(\text{ASSIGNED\_TO} \bowtie (\pi_{Project\#}\ (\sigma_{Project\_Name \, = \, \text{‘Database’}}\ (\text{PROJECT}))))$$

“Find the employee numbers of employees who do not work on project COMP453.”



$$\pi_{Emp\#}(\text{ASSIGNED\_TO}) - \pi_{Emp\#}(\sigma_{Project\# \, = \, \text{'COMP453'}}(\text{ASSIGNED\_TO}))$$

Exercise

Write the queries in relational algebra to retrieve the data from the relational schema

1. Student regno and name in data science major.
2. Name of the course registered by a student with regno 101.
3. Enrolment details of all students with score exceeds 90.

STUDENT (regno, name, major, bdate)

COURSE (courseno, cname, dept)

ENROLL (regno, courseno, sem, marks)