Tutorial: TUPLE RELATIONAL CALCULUS

Introduction

TRC is a nonprocedural query language, where each query is of the form

$$\{t \mid P(t)\}$$

where t = resulting tuples,
P(t) = known as predicate and these are the conditions that are used to fetch t.
P(t) may have various conditions logically combined with OR ($\vee$), AND ($\wedge$), NOT($\neg$).

It also uses quantifiers:
$\exists t \in r(Q(t))$ = "there exists" a tuple in t in relation r such that predicate Q(t) is true.
$\forall t \in r(Q(t))$ = Q(t) is true "for all" tuples in relation r.

- $\{P \mid \exists S \in Students(S.CGPA > 8 \wedge P.name = S.sname \wedge P.age = S.age)\}$: returns the name and age of students with a CGPA above 8.

TRC - Example

Student

Fname	Lname	Age	Course
David	Sharma	27	DBMS
Aaron	Lilly	17	JAVA
Sahil	Khan	19	Python
Sachin	Rao	20	DBMS
Varun	George	23	JAVA
Simi	Verma	22	JAVA

Q.1 Obtain the first name of students whose age is greater than 21.

Solution:

$$\{t.Fname \mid Student(t) \wedge t.age > 21\}$$

$$\{t.Fname \mid t \in Student \wedge t.age > 21\}$$

$$\{t \mid \exists s \in Student(s.age > 21 \wedge t.Fname = s.Fname)\}$$

Fname
David
Varun
Simi

TRC- Example

Consider the relational schema
student($\underline{rollNo}$, *name*, *year*, *courseId*)
course($\underline{courseId}$, *cname*, *teacher*)

Q.2 Find out the names of all students who have taken the course name 'DBMS'.

- $\{t \mid \exists s \in student\ \exists c \in course(s.courseId = c.courseId \wedge c.cname = \text{'DBMS'} \wedge t.name = s.name)\}$
- $\{s.name \mid s \in student \wedge \exists c \in course(s.courseId = c.courseId \wedge c.cname = \text{'DBMS'}\ )\}$

Q.3 Find out the names of all students and their rollNo who have taken the course name 'DBMS'.

- $\{s.name, s.rollNo \mid s \in student \wedge \exists c \in course(s.courseId = c.courseId \wedge c.cname = \text{'DBMS'}\ )\}$
- $\{t \mid \exists s \in student\ \exists c \in course(s.courseId = c.courseId \wedge c.cname = \text{'DBMS'} \wedge t.name = s.name \wedge t.rollNo = s.rollNo)\}$

TRC - Example

Consider the following relations:
Flights(*flno*, *from*, *to*, *distance*, *departs*, *arrives*)
Aircraft(*aid*, *aname*, *cruisingrange*)
Certified(*eid*, *aid*)
Employees(*eid*, *ename*, *salary*)

Q.4. Find the eids of pilots certified for Boeing aircraft.

RA

$$\Pi_{eid}(\sigma_{aname='Boeing'}(Aircraft \bowtie Certified))$$

TRC

- $\{C.eid \mid C \in Certified \wedge \exists A \in Aircraft(A.aid = C.aid \wedge A.aname = \text{'Boeing'})\}$
- $\{T \mid \exists C \in Certified \exists A \in Aircraft(A.aid = C.aid \wedge A.aname = \text{'Boeing'} \wedge T.eid = C.eid)\}$

Source: Database Management System by Raghu Ramakrishnan

TRC - Example(Cont..)

Consider the following relations:
Flights(*flno*, *from*, *to*, *distance*, *departs*, *arrives*)
Aircraft(*aid*, *aname*, *cruisingrange*)
Certified(*eid*, *aid*)
Employees(*eid*, *ename*, *salary*)

Q.5. Find the names and salaries of certified pilots working on Boeing aircrafts.

RA

$$\Pi_{ename,salary}(\sigma_{aname='Boeing'}(Aircraft \bowtie Certified \bowtie Employees))$$

TRC

$$\{P \mid \exists E \in Employees\ \exists C \in Certified\ \exists A \in Aircraft(A.aid = C.aid \wedge A.aname = \text{'Boeing'} \wedge E.eid = C.eid \wedge P.ename = E.ename \wedge P.salary = E.salary)\}$$

Source: Database Management System by Raghu Ramakrishnan.

TRC - Example(Cont..)

Consider the following relations:
Flights(*flno*, *from*, *to*, *distance*, *departs*, *arrives*)
Aircraft(*aid*, *aname*, *cruisingrange*)
Certified(*eid*, *aid*)
Employees(*eid*, *ename*, *salary*)

Q.6 Identify the flights that can be piloted by every pilot whose salary is more than $100,000. (Hint: The pilot must be certified for at least one plane with a sufficiently large cruising range.)

- $\{F.flno \mid F \in Flights \wedge \exists A \in Aircraft \exists C \in Certified \exists E \in Employees(A.cruisingrange > F.distance \wedge A.aid = C.aid \wedge E.salary > 100,000 \wedge E.eid = C.eid)\}$

Source: Database Management System by Raghu Ramakrishnan.