



## Part 2:-

# SQL Queries & Relational Algebra

1) :: Select r.name from Researchers r  
 Join project researches pr ON r.Researcher\_id  
 = pr.researcher\_Id  
 Join  
 project p on pr.project\_id = p.project\_id  
 where p.budget > 500000;

:: Tname (S Budget > 500000 (project))  $\bowtie$  Project\_Researches  $\bowtie$  Researchers

2) :: Select u.name from university u  
 Join  
 Researcher r on u.university\_id = r.affiliation  
 group by u.university\_id, u.name  
 having count(r.researcher\_id)  $\geq 3$ ;

:: Tname (S count(researcher\_id)  $\geq 3$  (G university\_id, Name;  
 count(researcher\_id)) (Researcher  $\bowtie$  university)

3) :: Select p.project, pf.project\_funding from projects  
 join project\_funding pf on p.project\_id = pf.project\_id  
 join Fund\_Agency fa on pf.Agency\_id = FA.  
 Agency\_id

TICK Where fa.name = "NSF";

∴ Ⓛ Name = 'NSF' (Fund-Agency) ✪ project\_funding ↔ projects;

4) ∴ select p.title, count(Pb.Publication\_id)  
 as Publication\_count from projects p

left join Publication Pb on p.project\_id =  
~~p.project~~ pb.project\_id group by p.project\_id,  
 p.title;

∴ π title, count(publication\_id) (σ project\_id, title; count(publication\_id))  
 (project ↔ Publications)

5) ∴ Select r.name from researchers r Join  
 projects p on r.researcher\_id = p.principal\_invest\_id  
 Join publication pb on p.project\_id = pb.project\_id  
 Join Journals j on pb.journal = j.journal  
 where j.rank  $\leq 10$ ;

∴ π name (σ Ranking  $\leq 10$  (researchers ↔ Projects ↔ publication  
 ↔ journal))

6) :: Select f.name, sum(f.total\_fund) as funding\_provided from funding\_Agencies f join project\_funding pf on f.agency\_id = pf.Agency\_id  
Group by f.agency\_id, f.name;

:: π name, sum(Total\_funding) (G1.Agency\_id, Agency\_Name,  
sum(Total\_funding) (funding\_Agencies AS Project\_funding))

7) :: Select p.project from projects p left join  
Pub\_id pb on p.project\_id = pb.project\_id  
where pb.publication\_id is Null;

:: π name, project\_id (σ publication\_id is Null (Project left join  
Publication))

8) :: Select r.name from researchers r  
join universities u on r.Affiliation = u.university\_id  
join project\_researchers pr on r.researcher\_id =  
pr.researcher\_id  
join project p on pr.project\_id = p.project\_id  
where u.country = "USA" AND p.budget > 100000;

$\therefore \pi_{name} ( \& country = "USA" \wedge Budget > 100000 )$   
 researchers  $\bowtie$  University  $\bowtie$  Project - Researcher  $\bowtie$   
 (Project))

9)  $\therefore$  Select u.name, count(pf.project\_id)  
 as project\_funding from university u  
 Join researchers r on u.University\_id =  
 r.Affiliation  
 join project\_researcher pr on r.Researcher\_id =  
 pr.researcher\_id  
 join project p on pr.project\_id = p.project\_id  
 join project\_funding pf on p.project\_id =  
 pf.project\_id  
 Group by u.uni\_id, u.name  
 Order by project\_funding desc  
 Limit 3;

$\therefore \pi_{name, count(Project-id)} ( \& Uni_id, uni_name, count($   
 Project\_id)  
 (Universities  $\bowtie$  researchers  $\bowtie$  project\_researchers  $\bowtie$   
 project  $\bowtie$  project\_funding) Order by count(project\_id)  
 desc Limit 3;

**10):-**

∴ Select p.project\_title, from project p  
join Milestone m on P.project\_id = m.project\_id  
where m.deadline < sysdate AND m.status =  
"In progress";

∴  $\pi_{name, title} ( \& deadline < sysdate \wedge status = 'In progress' )$   
(projects  $\bowtie$  Milestones)

## ⇒ Part 3: Key Identification

∴ Project Table have

Project-id, Title, Budget, Start date, End date

Superkey :-

Multiple or single attribute combination that is unique.

{Project-id}

Unique ✓

{Project-id, Budget}

✓

{Project-id, Title}

✓

Candidate key :-

The key which is unique and also minimal like single attribute, or attribute should not be repeated.

{Project-id}

{Rname, Project\_title}

Primary key :-

in project table      Key which always unique . here  
                              {Project\_id}

Alternate key :-

The remaining key which not  
select for primary key called Alternate key.

{Name , Project-title}

Researcher Table

Researcher\_id, Name, Specialization, Affiliation, email

Superkey :-

{Researcher\_id}

{Researcher\_id, Email}

{Researcher\_id, Specialization}

Candidate key :-

{Researcher\_id}

{Email}

**Alternate key:-**

{Email}

**Primary key:-**

{Researcher\_id}

**Universities Table**

{University\_id, Name, Country, Ranking}

**Super key:-**

{uni\_id}  
 {uni\_id, uni\_name}  
 {uni\_id; country}

**Candidate keys:-**

{University\_id}, {Name, Country}

**Alternate key:-**

{Name, country} None

**Primary key:-**

{University\_id}

## ∴ Funding Agencies

{Agency\_id, Name, Country, Funding}

Super key :-

{Agency\_id}

{Agency\_id, Name, Country}

{Agency\_id, Country, Funding}

Candidate key :-

{Agency\_id}

~~{Agency\_id, Name}~~

{Name, Country, Funding}

Primary key :-

{Agency\_id}

Alternate key :-

None

## ∴ Publication Table :-

(Publication\_id, Title, Year, Journal, Project\_id)

Super key :-

{Pub\_id}

{Pub\_id, Title}

, {Pub\_id, Year, Jam}

Candidate keys:-

{Publication\_id}

Alternate keys:-

None

Primary key:-

{Publication\_id}