SQL

1. Which PROJECT has maximum number of EMPLOYEES?

select max(countno),project\_id  
from(select count(emp\_id) as countno,project\_id   
from allocation group by project\_id);

2. Which EMPLOYEE has not yet been allocated to any PROJECT?

3. Which role played by the employee 'E03' frequently?

select role\_id,emp\_id from allocation where project\_id=3;

4. Which is the costliest Project?   
 select max(high),project\_id   
 from (select sum(amount\_per\_day)  
 as high,project\_id   
 from allocation   
 group by project\_id);

5. How many employees were there in costliest Project?   
 select distinct count(emp\_id) from allocation where project\_id=1

6. Which is the cheapest Project in the year 2012?  
 select min(lowcost),project\_id  
 from (select sum(amount\_per\_day)  
 as lowcost,project\_id  
 from allocation where to\_date=2012  
 group by project\_id)

7. What is the salary of employee, who played maximum roles in Project 'P07'?  
Select empid,count(role\_id),project\_id from ALLOCATION wher project\_id =101 group by role\_id

8. select count(project\_id) as noofproject,emp\_name,min(hire\_datetime) from eight group by emp\_name limit 1

9. select sum(amount\_per\_day),emp\_id   
from allocation where project\_id is null group by emp\_id

10. select emp\_id,count(project\_id),to\_date from allocation group by emp\_id

11. select sum(countno) as noofemployees from(select count(emp\_id) as countno   
from allocation   
group by project\_id)  
 where countno <=10

12. select count(emp\_id) as noofemployee from allocation   
where role\_id=2 and project\_id=4

13. create view thirteen as  
select \* from allocation left outer join t\_project   
on allocation.project\_id=t\_project.project\_id  
select client\_name,count(project\_id) as noofprojects,project\_id   
from thirteen group by project\_id order by noofprojects desc limit 1

14. select emp\_id from allocation where project\_id is null and from\_date=10

19. Which Project has taken maximum duration?   
Select project from T\_PROJECT where deadline = max(values)

21. create view newview as  
select \* from allocation left outer join t\_project on allocation.project\_id=t\_project.project\_id;  
select \* from newview;  
select count(emp\_id) as NoOfEmployees,project\_name as ProjectName   
from newview group by project\_name;

22. create view newviews as   
select \* from allocation left outer join  
 role on allocation.role\_id=role.role\_id;  
select \* from newviews;  
select count(emp\_id) as NoOfEmployee,role\_title as RoleName   
from newviews group by role\_title;

23. create view new as  
select \* from allocation left outer join employee   
on employee.emp\_id=allocation.emp\_id;   
select \* from new;  
select emp\_name as EmpName,count(project\_id)   
as NoOfProjects from new group by emp\_name;

24. create view new as  
select \* from allocation left outer join employee   
on employee.emp\_id=allocation.emp\_id;   
select \* from new;  
select emp\_name as EmpName,count(role\_id)   
as NoOfRoles from new group by emp\_name;

25. create view twofive as  
select \* from allocation left outer join role  
on role.role\_id=allocation.role\_id;   
select \* from twofive;  
select role\_title as RoleName,count(emp\_id)   
as NoOfEmployees from twofive group by role\_title;

26. create view twofive as  
select \* from allocation left outer join role  
on role.role\_id=allocation.role\_id;   
select \* from twofive;  
select role\_title as RoleName,count(project\_id)   
as NoOfProjects from twofive group by role\_title;

27. create view twosix as  
select emp\_name,role\_title,project\_id   
from employee e inner join allocation a   
on e.emp\_id=a.emp\_id inner join role r   
on r.role\_id=a.role\_id;  
select \* from twosix;  
select emp\_name,role\_title,count(project\_id) from twosix group by emp\_name;

28. create view twoseven as  
select project\_name,role\_title,emp\_id  
from t\_project p inner join allocation a   
on p.project\_id=a.project\_id inner join role r   
on r.role\_id=a.role\_id;  
drop view if exists twoseven;  
select \* from twoseven;  
select project\_name as NameOfProject ,role\_title as  
 RoleTitle,count(emp\_id) as NoOfEmployee from twoseven group by project\_name;

29. create view twoeight as  
select emp\_name,role\_title,project\_id   
from employee e inner join allocation a   
on e.emp\_id=a.emp\_id inner join role r   
on r.role\_id=a.role\_id;  
drop view if exists twosix;  
select \* from twoeight;  
select role\_title as RoleName,emp\_name as NameOfEmployee,count(project\_id)   
as NoOfProjects from twosix group by emp\_name;

30. select dept\_id as DepartmentID,count(emp\_id) as NoOfEmployee from department left outer join employee   
on employee.dept\_id=department.dept\_id group by dept\_name;

31. select mgr\_id,count(emp\_id) from employee group by mgr\_id;

32. create view thirtytwo as  
select \* from allocation a   
inner join employee e on e.emp\_id=a.emp\_id   
inner join role r on r.role\_id=a.role\_id   
inner join t\_project p on p.project\_id=a.project\_id;   
select emp\_name,role\_title,project\_name from thirtytwo;

33. select project\_id,emp\_id,sum(amount\_per\_day) as amount\_collected from allocation   
group by emp\_id;

34. select emp\_id,role\_id,sum(amount\_per\_day)   
as amount\_collected from allocation group by amount\_per\_day

35. select emp\_id,role\_id,project\_id,sum(amount\_per\_day)   
as amount\_collected from allocation group by amount\_per\_day

36. select emp\_id,mgr\_id,case when mgr\_id is not null then 'has manager'   
else 'no manager' end as comment from employee