Formulate the following queries: (note, for some queries, you may need to use the combination of join and subqury or correlated subquery, together with group by and having clauses).

1. List the name of project sponsored by chen's division. (hint/think: find a project whose DID  equals to the DID of an employee whose name is chen)
2. List the name of employee who is working on the project whose budget is below the **divisional** average project budget (use correlated subquery).
3. List the name of project that some employee(s) who is/are working on it make less than **divisional** average salary (use correlated subquery).
4. List the total number of division that has 2 or more employees working on projects. For this query I built a framework of the code, you just need to fill in the right code in \_\_\_\_ , and then run the code to get the result.

     select count(did)  
     from \_\_\_\_\_  
     where did in  
         (select \_\_\_\_  
          from \_\_\_ , \_\_\_\_\_  
          where \_\_\_\_\_\_\_  
          group by \_\_\_  
          having \_\_\_\_\_\_\_\_ )

5. List the total number of projects 'accounting' division manager works on. (Note, if an employee is a division's manager, his/her empID is IN the Division table)

6. List the name of the employees (and his/her DID) who work on more projects than his/her ***divisional***colleagues. (hint: co-realated subquery, also use having , compare count() to count, use “ … having count (pid) >=ALL (select count (pid) …..)

7. List the name of the division that has more than one employee whose salary is greater than company's  average salary  (subquery,  group by, having)

8. List the name of the division that has more than one employee whose salary is greater than the divisional average salary  (corelated subquery, group by, having)

9. List the name of the employee that has the lowest salary in his division and list the total number of projects this employee is work on  (use corelated subquery)

10.  List the name of project Larry does not work on.

11. List the name of employee in Chen's division who works on a project that Chen  does NOT work on.

12. List the name of divisions that sponsors project(s)  Chen works on . (Namely, if there is a project 'chen' works on, find the name of the division that sponsors that project.)

13.  List the name of division (d) that has employee who work on a project (p) not sponsored by this division. (hint in a corelated subquery where d.did <> p.did)

14.  List the name of employee who work with Chen on some project(s).

15. Increase the salary of employees in engineering division by 10% if they work on more than 1 project.

16. Increase the budget of a project by 10% if it has more than two employees working on it.

**17. (bonus)**List the name of employee who work on all project (hint: Use NOT EXISTS predicate .  The logic of the code is  to find an employee that there NOT exists a project this employee does NOT work on,  Use NOT EXISTS twice, note, there is a very similar query in chapter 8).