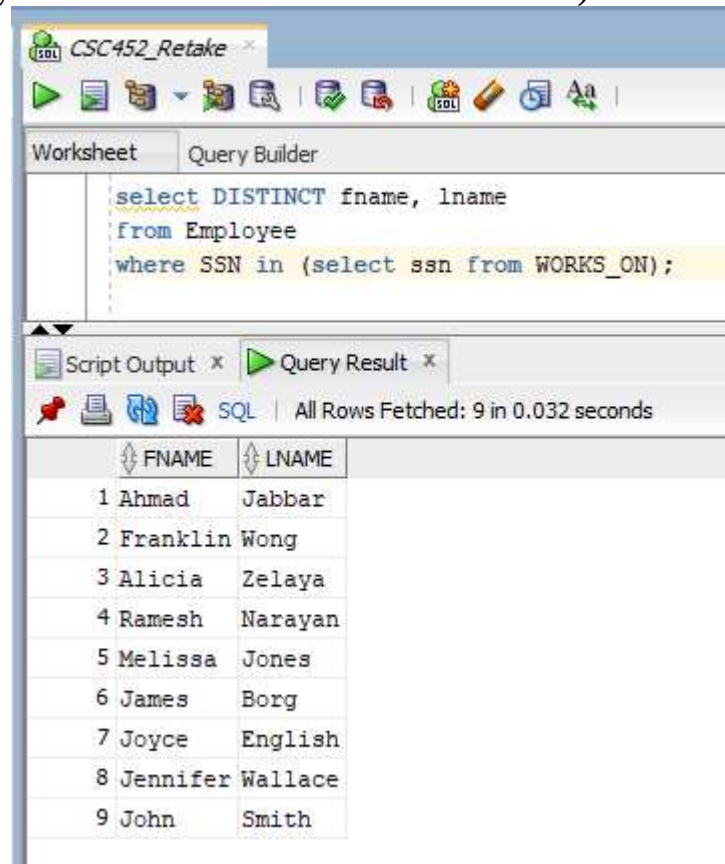


1. [2pt] Retrieve the names of all employees who work on at least one of the projects.  
(In other words, look at the list of projects given in the PROJECT table, and retrieve the names of all employees who work on at least one of them.)



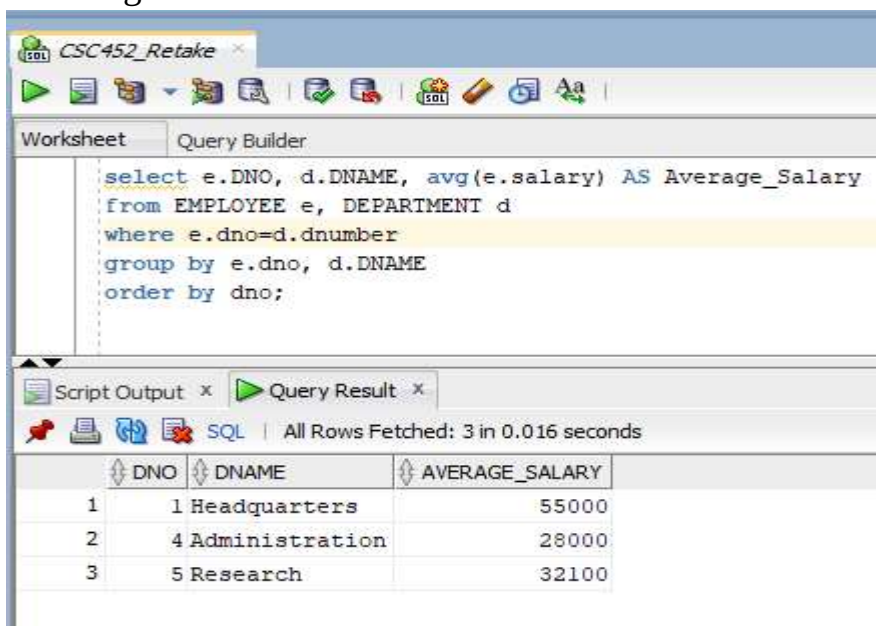
The screenshot shows the SQL Developer interface with a query window titled 'CSC452\_Retake'. The query is as follows:

```
select DISTINCT fname, lname
from Employee
where SSN in (select ssn from WORKS_ON);
```

The 'Query Result' tab shows the output of the query, displaying 9 rows of employee names. The columns are 'FNAME' and 'LNAME'.

	FNAME	LNAME
1	Ahmad	Jabbar
2	Franklin	Wong
3	Alicia	Zelaya
4	Ramesh	Narayan
5	Melissa	Jones
6	James	Borg
7	Joyce	English
8	Jennifer	Wallace
9	John	Smith

2. [2pt] For each department, retrieve the department name and the average salary of all employees working in that department. Order the output by department number in ascending order.



The screenshot shows the SQL Developer interface with a query window titled 'CSC452\_Retake'. The query is as follows:

```
select e.DNO, d.DNAME, avg(e.salary) AS Average_Salary
from EMPLOYEE e, DEPARTMENT d
where e.dno=d.dnumber
group by e.dno, d.DNAME
order by dno;
```

The 'Query Result' tab shows the output of the query, displaying 3 rows of department information. The columns are 'DNO', 'DNAME', and 'AVERAGE\_SALARY'.

	DNO	DNAME	AVERAGE_SALARY
1	1	Headquarters	55000
2	4	Administration	28000
3	5	Research	32100

3. [3pt] List the last names of all department managers who have no dependents.

The screenshot shows the SQL Developer interface with a query window titled 'DEPENDENT'. The query is as follows:

```
select e.lname
from employee e, department d
where e.ssn=d.MGR_SSN and e.ssn not in (select essn from dependent);
```

The 'Query Result' tab shows the following result:

LNAME
1 Borg

4. [3pt] Retrieve the names of all employees who work in the department that has the employee with the lowest salary among all employees.

The screenshot shows the SQL Developer interface with a query window titled 'DEPENDENT'. The query is as follows:

```
select e.DNO, e.fname, e.lname, d.DNAME, e.salary
from EMPLOYEE e, DEPARTMENT d
where e.DNO=d.DNUMBER and
      (e.dno, e.salary)in
      (select dno, MIN (salary)
       from employee
       group by dno
      );
```

The 'Query Result' tab shows the following results:

DNO	FNAME	LNAME	DNAME	SALARY
1	1 James	Borg	Headquarters	55000
2	4 Ahmad	Jabbar	Administration	22000
3	5 Joyce	English	Research	25000

5. [3pt] Find the total number of employees and the total number of dependents for every department (the dependents for the department are the dependents of all employees working for that department).

The screenshot shows the SQL Developer interface with a query window titled 'DEPENDENT'. The query is as follows:

```
select e.dno as Dept_number, count(*) as Emp_count
from dependent d
join employee e on d.essn=e.ssn
group by e.ssn, e.dno
order by e.dno;

/*(e.dno, e.salary)in
(select dno, MIN (salary)
```

Below the query window, the 'Query Result' tab shows the following data:

DEPT_NUMBER	EMP_COUNT
1	2
2	1
3	3
4	3

6. [2pt] Retrieve the names of employees whose salary is within \$20,000 of the salary of the employee who is paid the most in the company (e.g., if the highest salary in the company is \$80,000, retrieve the names of all employees that make at least \$60,000).

The screenshot shows the SQL Developer interface with a query window titled 'Query Builder'. The query is as follows:

```
select *
from employee
where salary >= (select MAX(salary) from employee) - 20000;
```

Below the query window, the 'Query Result' tab shows the following data:

FNAME	MINIT	LNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPER_SSN	DNO
1 James	E	Borg	888665555	10-NOV-37	450 Stone, Houston, TX	M	55000	(null)	1
2 Jennifer	S	Wallace	987654321	20-JUN-41	291 Berry, Bellaire, Tx	F	37000	888665555	4
3 Franklin	T	Wong	333445555	08-DEC-55	638 Voss, Houston, TX	M	40000	888665555	5
4 Ramesh	K	Narayan	666884444	15-SEP-20	975 Fire Oak, Humble, TX	M	38000	333445555	5