IS 2511 – Fundamentals of Database Systems

Lab assignment # 4

Answer the following questions by using **(EMP and DEPT) script** then copy your code to a text document (with print screens) .

A- Run (EMP and DEPT) script.

B- Then apply the following queries:

1- List out the Employee names with their department names **, And display the** Employee names **alphabetically.**

2- List out the Employee names with their department names **for employees** who got salary **less than 3000$.**

3- List out the departments names with number of employee who works on, start from the biggest department (which have the largest number of employees).

4- List out the departments number with number of employee who works on, only departments with more than 5 employees will be returned.

5- Return the number of unique departments .

6- What is the maximum salary.

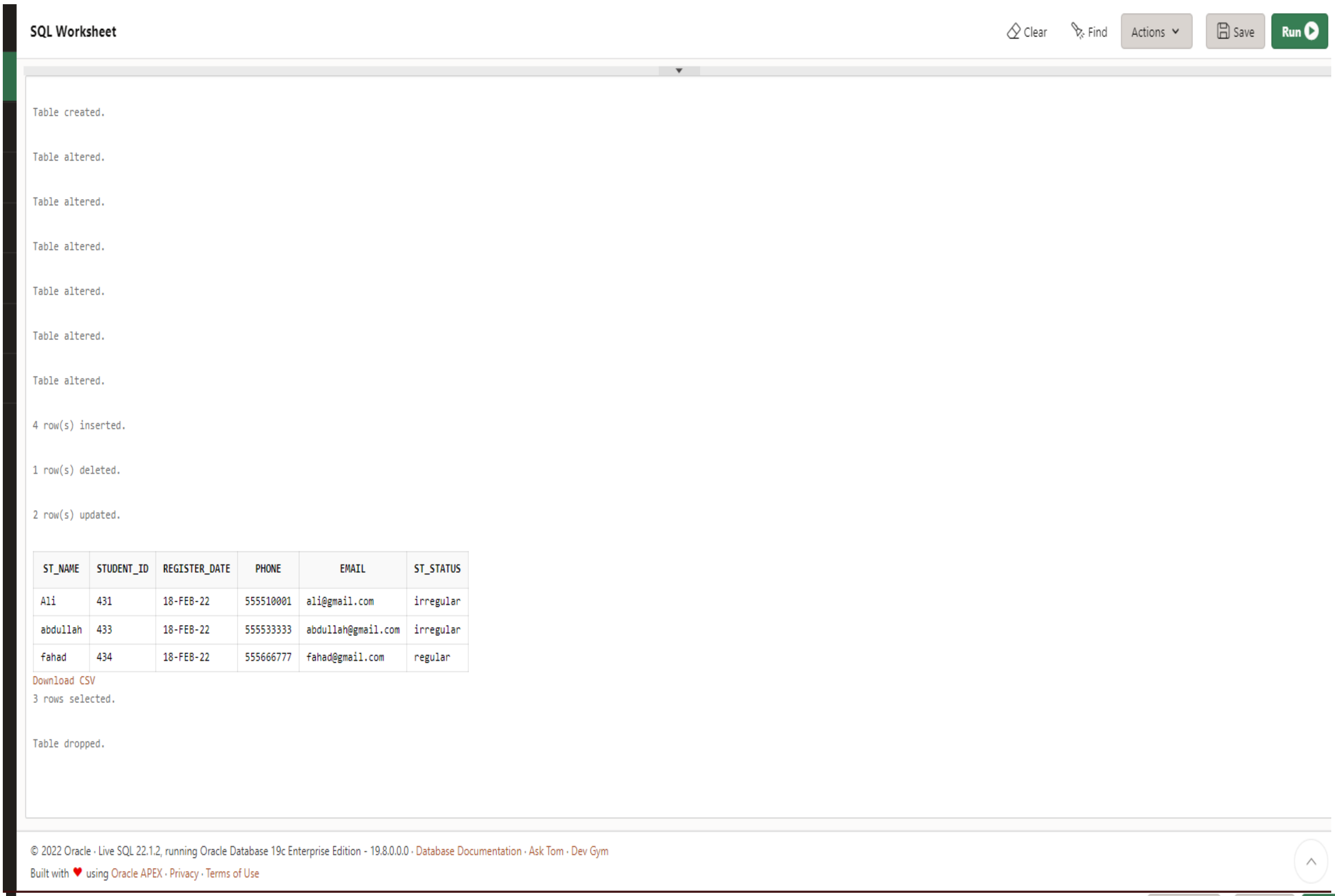
7- What is the maximum salary for each department.

8- Return the name and the salary of the employee who has the minimum salary.

9- Return the name of department and the minimum salary in the department, only those departments whose minimum salary is less than 2000 $.

10- Create a new table called emp_loc contains the employee names and locations where they works on.

```
select ENAME , DNAME from emp inner join dept on emp.DEPTNO = dept.DEPTNO   order by ENAME ;

select ENAME , DNAME from emp inner join dept on emp.DEPTNO = dept.DEPTNO  where SAL < 3000 ;

select DNAME , count(*)  from emp , dept where emp.DEPTNO = dept.DEPTNO group by DNAME order by count(*) Desc ;

select DEPTNO , count(*)  from emp  group by DEPTNO  having count(*)>5;

select count(distinct DEPTNO)  from dept ;

select max( SAL )  from emp ;

select DEPTNO , max( SAL )  from emp  group by DEPTNO;

select ENAME , SAL   from emp  where SAL = ( select min(SAL) from emp );

select DNAME, min(SAL)  from dept, emp where dept.DEPTNO = emp.DEPTNO group by DNAME  having min(SAL) < 2000;

create table emp_loc  as (select emp.ENAME, dept.LOC from emp, dept where emp.DEPTNO = dept.DEPTNO);
create table student (
st_Name varchar2(30) ,
st_ID number(3)   not null ,
Register_date date default  sysdate
);

alter table student
add birth_date date ;

alter table student
add ( phone number(8),
email varchar2(25) ,
st_status varchar2(25) ) ;

alter table student
modify ( email varchar2(50) ,
phone number(10) );

alter table student
drop column birth_date ;

alter table student
rename column st_ID to student_ID ;

alter table student
rename to students ;

insert all
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'Ali' , 431 ,
sysdate , 0555510001 , 'ali@gmail.com' , 'regular' )
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'ahmad' ,
432 , sysdate , 054411177 , 'ahmad@gmail.com' , 'irregular' )
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'abdullah' ,
433 , sysdate , 0555533333 , 'abdullah@gmail.com' , 'regular' )
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'fahad' ,
434 , sysdate , 0555666777 , 'fahad@gmail.com' , 'regular' )
select * from dual ;

delete from students
where st_status =  'irregular' ;

update students
set st_status = 'irregular'
where st_Name = 'Ali' or phone = 0555533333 ;

select * from students ;

drop table students ;
```



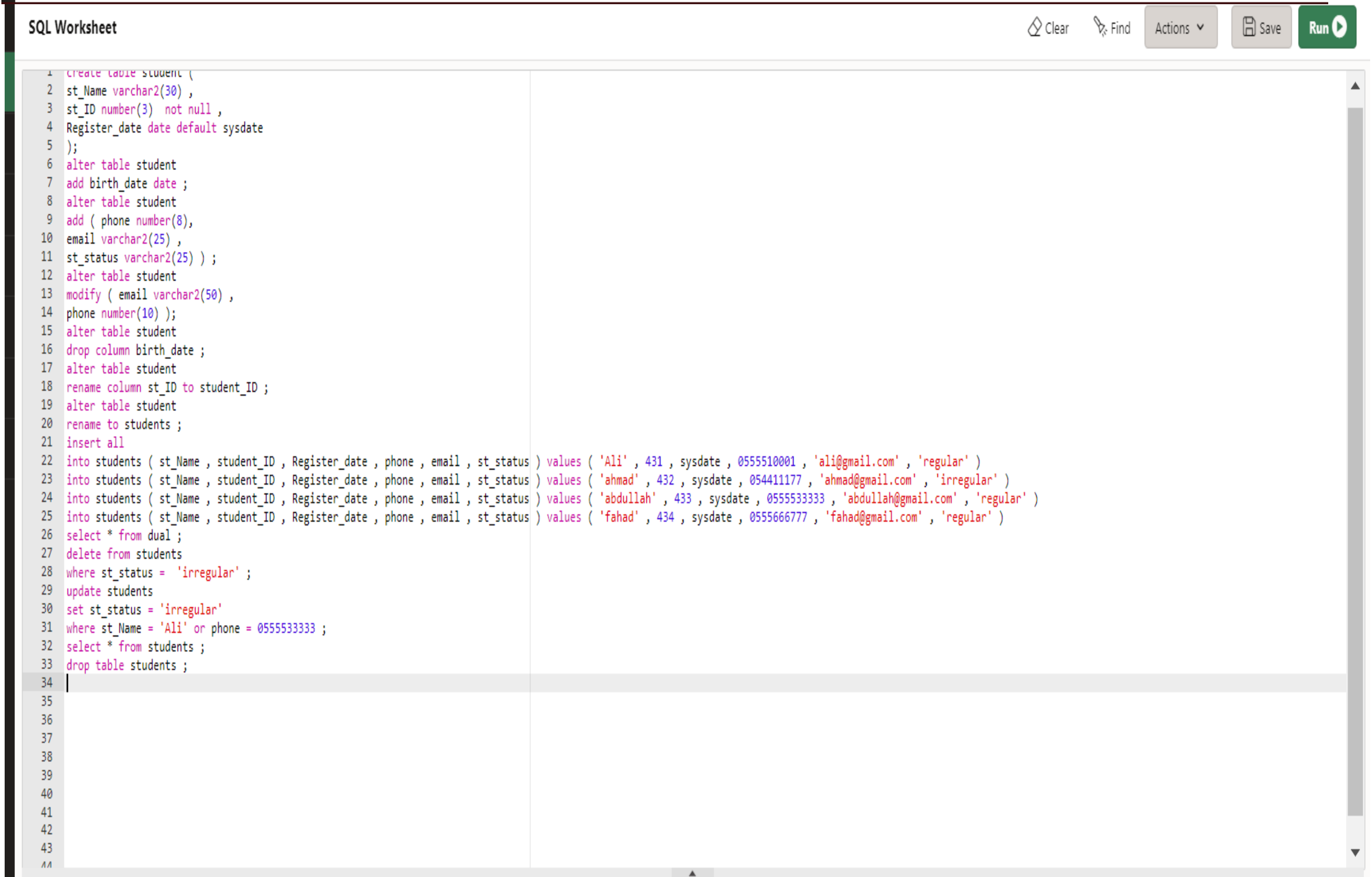
```
Table created.

Table altered.

Table altered.

Table altered.

Table altered.

Table altered.

Table altered.

4 row(s) inserted.

1 row(s) deleted.

2 row(s) updated.
```

ST_NAME	STUDENT_ID	REGISTER_DATE	PHONE	EMAIL	ST_STATUS
Ali	431	18-FEB-22	555510001	ali@gmail.com	irregular
abdullah	433	18-FEB-22	555533333	abdullah@gmail.com	irregular
fahad	434	18-FEB-22	555666777	fahad@gmail.com	regular

Download CSV

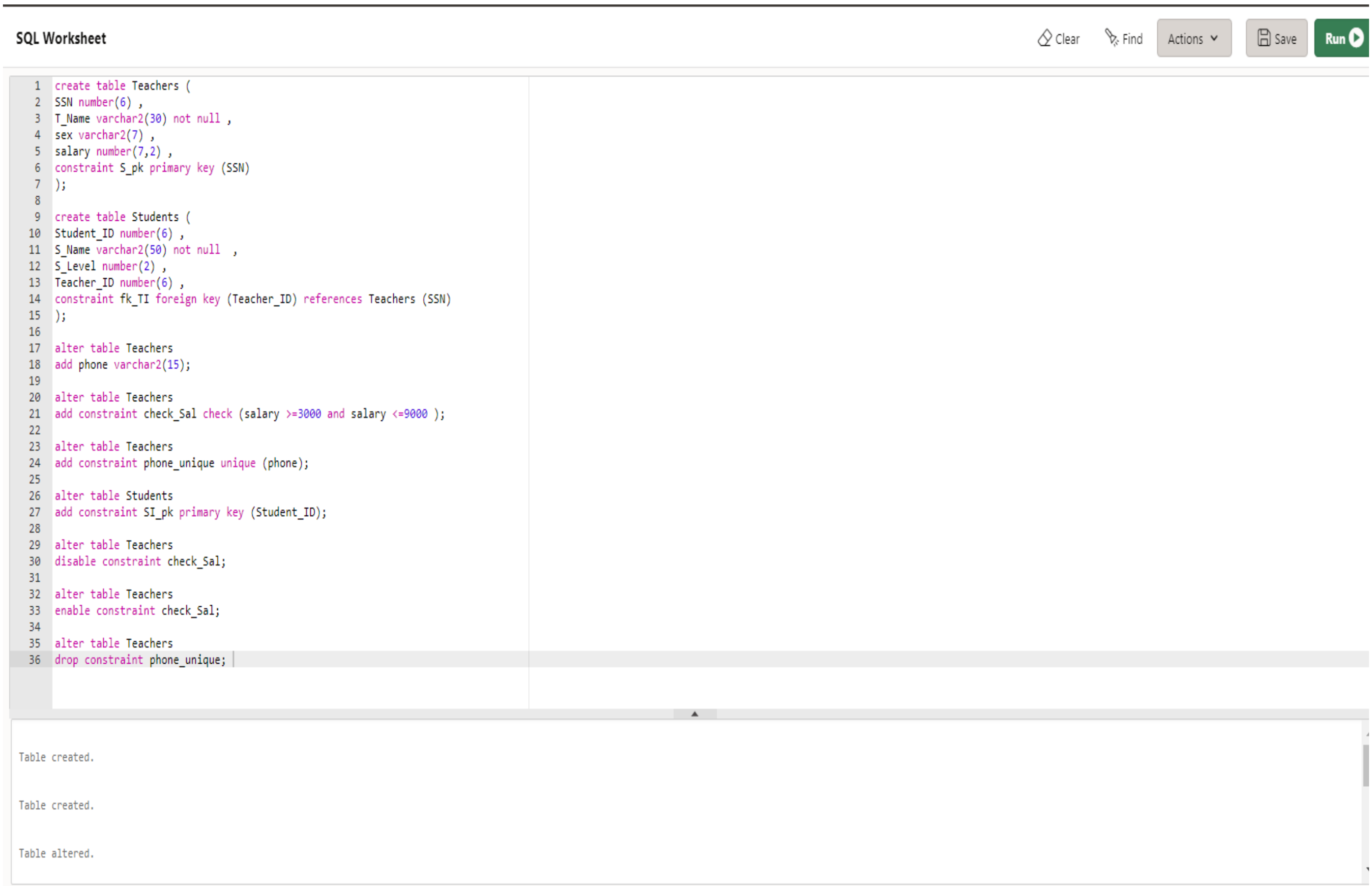
```
3 rows selected.

Table dropped.
```

© 2022 Oracle · Live SQL 22.1.2, running Oracle Database 19c Enterprise Edition - 19.8.0.0.0 · Database Documentation · Ask Tom · Dev Gym
Built with ♥ using Oracle APEX · Privacy · Terms of Use



```
create table student (
st_Name varchar2(30) ,
st_ID number(3)  not null ,
Register_date date default sysdate
);
alter table student
add birth_date date ;
alter table student
add ( phone number(8),
email varchar2(25) ,
st_status varchar2(25) ) ;
alter table student
modify ( email varchar2(50) ,
phone number(10) );
alter table student
drop column birth_date ;
alter table student
rename column st_ID to student_ID ;
alter table student
rename to students ;
insert all
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'Ali' , 431 , sysdate , 0555510001 , 'ali@gmail.com' , 'regular' )
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'ahmad' , 432 , sysdate , 054411177 , 'ahmad@gmail.com' , 'irregular' )
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'abdullah' , 433 , sysdate , 0555533333 , 'abdullah@gmail.com' , 'regular' )
into students ( st_Name , student_ID , Register_date , phone , email , st_status ) values ( 'fahad' , 434 , sysdate , 0555666777 , 'fahad@gmail.com' , 'regular' )
select * from dual ;
delete from students
where st_status =  'irregular' ;
update students
set st_status = 'irregular'
where st_Name = 'Ali' or phone = 0555533333 ;
select * from students ;
drop table students ;
```

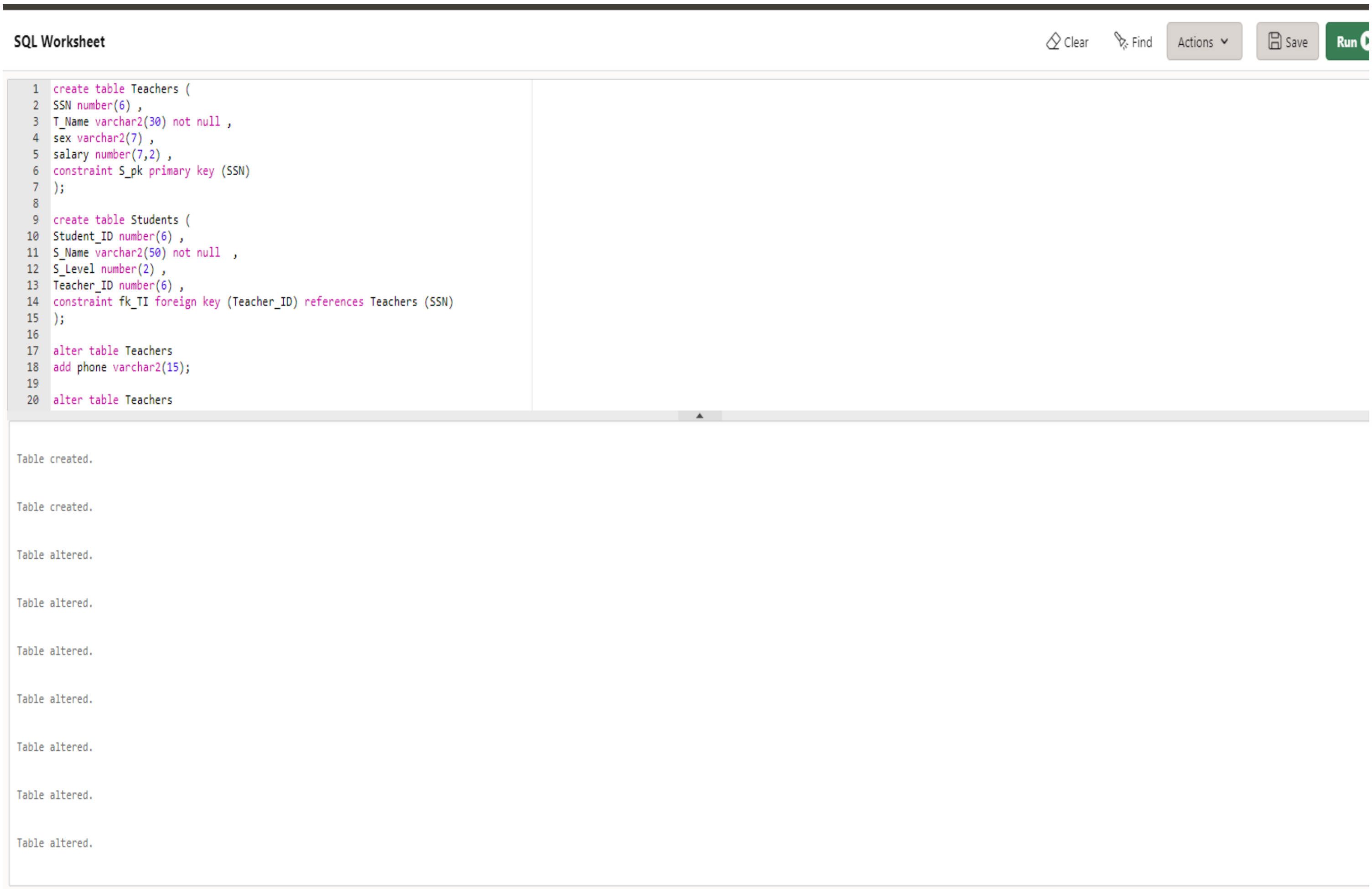
```
create table Teachers (
SSN number(6) ,
T_Name varchar2(30) not null ,
sex varchar2(7) ,
salary number(7,2) ,
constraint S_pk primary key (SSN)
);

create table Students (
Student_ID number(6) ,
S_Name varchar2(50) not null  ,
S_Level number(2) ,
Teacher_ID number(6) ,
constraint fk_TI foreign key (Teacher_ID) references Teachers (SSN)
);

alter table Teachers
add phone varchar2(15);

alter table Teachers
add constraint check_Sal check (salary >=3000 and salary <=9000 );

alter table Teachers
add constraint phone_unique unique (phone);

alter table Students
add constraint SI_pk primary key (Student_ID);

alter table Teachers
disable constraint check_Sal;

alter table Teachers
enable constraint check_Sal;

alter table Teachers
drop constraint phone_unique;
```



SQL Worksheet

```
create table Teachers (
SSN number(6) ,
T_Name varchar2(30) not null ,
sex varchar2(7) ,
salary number(7,2) ,
constraint S_pk primary key (SSN)
);

create table Students (
Student_ID number(6) ,
S_Name varchar2(50) not null  ,
S_Level number(2) ,
Teacher_ID number(6) ,
constraint fk_TI foreign key (Teacher_ID) references Teachers (SSN)
);

alter table Teachers
add phone varchar2(15);

alter table Teachers
add constraint check_Sal check (salary >=3000 and salary <=9000 );

alter table Teachers
add constraint phone_unique unique (phone);

alter table Students
add constraint SI_pk primary key (Student_ID);

alter table Teachers
disable constraint check_Sal;

alter table Teachers
enable constraint check_Sal;

alter table Teachers
drop constraint phone_unique;
```

```
Table created.


Table created.


Table altered.
```



SQL Worksheet

```
create table Teachers (
SSN number(6) ,
T_Name varchar2(30) not null ,
sex varchar2(7) ,
salary number(7,2) ,
constraint S_pk primary key (SSN)
);

create table Students (
Student_ID number(6) ,
S_Name varchar2(50) not null  ,
S_Level number(2) ,
Teacher_ID number(6) ,
constraint fk_TI foreign key (Teacher_ID) references Teachers (SSN)
);

alter table Teachers
add phone varchar2(15);

alter table Teachers
```

```
Table created.


Table created.


Table altered.


Table altered.


Table altered.


Table altered.


Table altered.


Table altered.


Table altered.
```

IS 2511 – Fundamentals of Database Systems

Lab Home Work # 3

Answer the following questions then copy your code to a text document (with print screens) and upload it on Blackboard before the due date.

A- Create this **Employees** table

First_name	Last_name	salary	Department_id
Saad	Almahmod	1500	66
Saleh	Alnaser	10500	50
Abdullah	Al_badr	2000	50
Mohammad	Alabdukarem	18000	50
Nasser	Almasoud	11000	70

B- Then apply the following queries:

1- Retrieve all the employees who got salary more than 2000$

2- Retrieve all the employees who got salary **in the range** 2000$ to 11000$

3- Retrieve all the employees who got salary **out of the range** 2000$ to 11000$

4- Retrieve all the employees who got salary 2000$, 10500$ or 17000$ using the **IN** operator

5- Retrieve all the employees who got salary **other than** 2000$, 10500$ or 17000$ using the **IN** operator

6- Retrieve employee first name, last name and display his Full name which is the concatenation of his first name and last name together.

7- Retrieve the first name, last name and salary for all employees whose first name start with the letter **S**.

8- Retrieve the first name, last name and salary for all employees whose last name include the letter **s.**

9- Retrieve the first name and last name for all employees whose first name consist of 4 letters.

10- Retrieve the first name and last name for all employees whose first name OR last name contains _ (underscore symbol).

11- Retrieve the first_name , last_name, department_id and salary for all employees who is working in department no **50** sorted by their **salary** in descending order.

12- Retrieve the first_name , last_name, department_id and salary for all employees sorted by their department_id and their salary in descending order.

13- List out the departments' numbers where employees work.

```
CREATE TABLE Employees (
 First_name varchar2(20) not null,
 Last_name varchar2(35) not null,
 salary number(10) not null,
 Department_id number (6) not null,
 CONSTRAINT Fn_pk PRIMARY KEY (First_name)
 );
 INSERT INTO Employees (First_name, Last_name, salary, Department_id) VALUES ('Saad', 'Almahmod', 1500, 66);
 INSERT INTO Employees (First_name, Last_name, salary, Department_id) VALUES ('Saleh','Alnaser', 10500, 50);
 INSERT INTO Employees (First_name, Last_name, salary, Department_id) VALUES ('Abdullah', 'Al_badr', 2000, 50);
 INSERT INTO Employees (First_name, Last_name, salary, Department_id) VALUES ('Mohammad', 'Alabdukarem', 18000, 50);
 INSERT INTO Employees (First_name, Last_name, salary, Department_id) VALUES ('Nasser', 'Almasoud', 11000, 70);
 SELECT * FROM Employees WHERE salary > 2000;
 SELECT * FROM Employees WHERE salary BETWEEN 2000 AND 11000;
 SELECT * FROM Employees WHERE salary NOT BETWEEN 2000 AND 11000;
 SELECT * FROM Employees WHERE salary IN (2000, 10500, 17000);
SELECT * FROM Employees WHERE salary NOT IN (2000, 10500, 17000);

select First_name ||' '|| Last_name as "full name" from Employees;

select first_name , last_name , salary from Employees where first_name like 'S%';

select first_name , last_name , salary from Employees where first_name like '%s%' ;

SELECT First_name,Last_name FROM Employees WHERE First_name LIKE '____' ;

SELECT First_name,Last_name FROM Employees WHERE First_name LIKE '%\_%' OR Last_name LIKE '%\_%' ESCAPE '\';

SELECT * FROM Employees WHERE department_id = 50 ORDER BY salary DESC;

SELECT First_name, Last_name, Department_id, salary from Employees order by Department_id DESC , salary DESC;

SELECT DISTINCT Department_id FROM Employees;
```

Lab Assignment#1

Answer the following questions in **Oracle live sql**.

1- Create a table called **table1** with the following specifications

Column	Data Type	Default value	
C1	Varchar2(30)		Not null
C2	Number(3)	2	
C3	Date	sysdate	

:Hint

- The SYSDATE function is used to retrieve the current database system time in Oracle and MySQL. A common use of SYSDATE is to get today's date.

2- Insert in to table1 following info

C1	C2	C3
aa	11	sysdate
bb	22	sysdate

3-show table1 info using sql statement:
(select * from table1;)

4-Modify Table1 to make the following changes :

a) Add new column C4 number(2) .

b) Change the size of C1 to be varchar2(50)

c) Drop the column C2 .

5- show table1 info using sql statement:
(select * from table1;)

6- Rename the Table name to be **table2** instead of table1 .

7- update C4 to 77 , where the C1 is 'aa'.

8- delete all records in the table with C1='bb'

9- show table1 info using sql statement:
(select * from table2;)

10- Drop the Table.

```
create table table1 (
C1 varchar2(30) not null ,
C2 number(3) default(2) ,
C3 date default sysdate
);

insert all
into table1 ( C1 , C2 , C3 ) values ( 'aa' , 11 , sysdate )
into table1 ( C1 , C2 , C3 ) values ( 'bb' , 22 , sysdate )
select * from dual ;

select * from table1 ;

alter table table1
add C4 number(2);
alter table table1
modify C1 varchar2(50);
alter table table1
drop column C2 ;

select * from table1 ;

alter table table1
rename to table2 ;

update table2
set C4 = 77
where C1 ='aa' ;

delete from table2
where C1 ='bb';

select * from table2 ;
drop table table2 ;
```

Lab Assignment#2

Answer the following questions in **Oracle live sql**.

1- Create a table called **table1** with the following specifications

Column	Data Type	Constraint
C1	Varchar2(30)	Primary key
C2	Number(3)	Not null
C3	Date	unique

2- Add constraint that checks the column C2 to be between 100 and 400.
3- drop the unique constraint.
4- create **table2**

Column	Data Type	Constraint
A1	Varchar2(30)	foreign key referenced c1 in table1, with delete cascade
A2	Number(3)	Primary key

5- disable the primary key of **table2** .

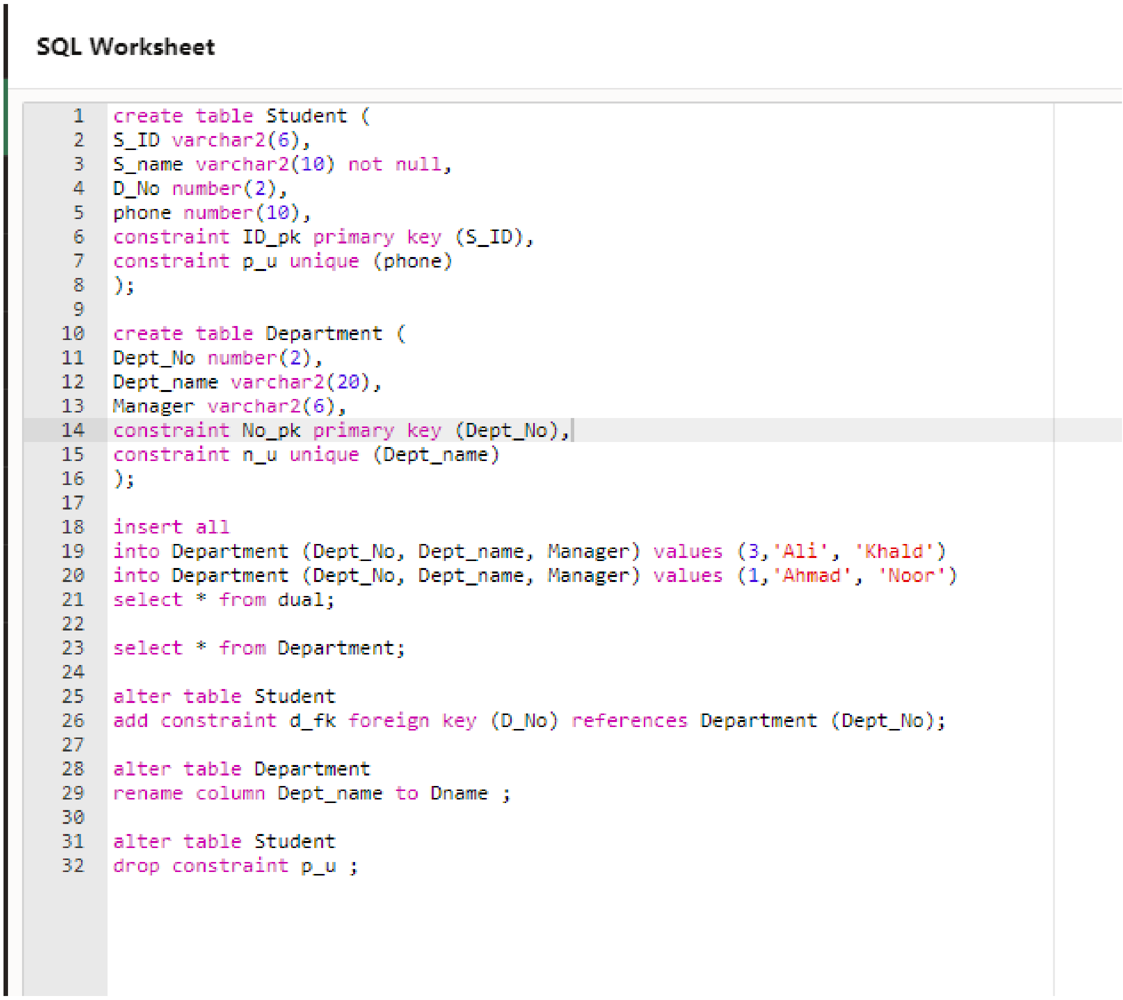
```
create table table1 (

C1 varchar2(30) ,

C2 number(3) not null ,

C3 date ,

constraint date_unique unique (C3),

constraint c_pk primary key (C1)

);
```

```
alter table table1
add constraint check_num check (C2 >=100 and C2 <=400 );

alter table table1
drop constraint date_unique ;

create table table2 (
A1 varchar2(30) ,
A2 number(3) ,
constraint a_pk primary key (A2) ,
constraint fk_a1 foreign key (A1) references table1 (C1) on delete cascade
);

alter table table2
disable constraint a_pk ;
```

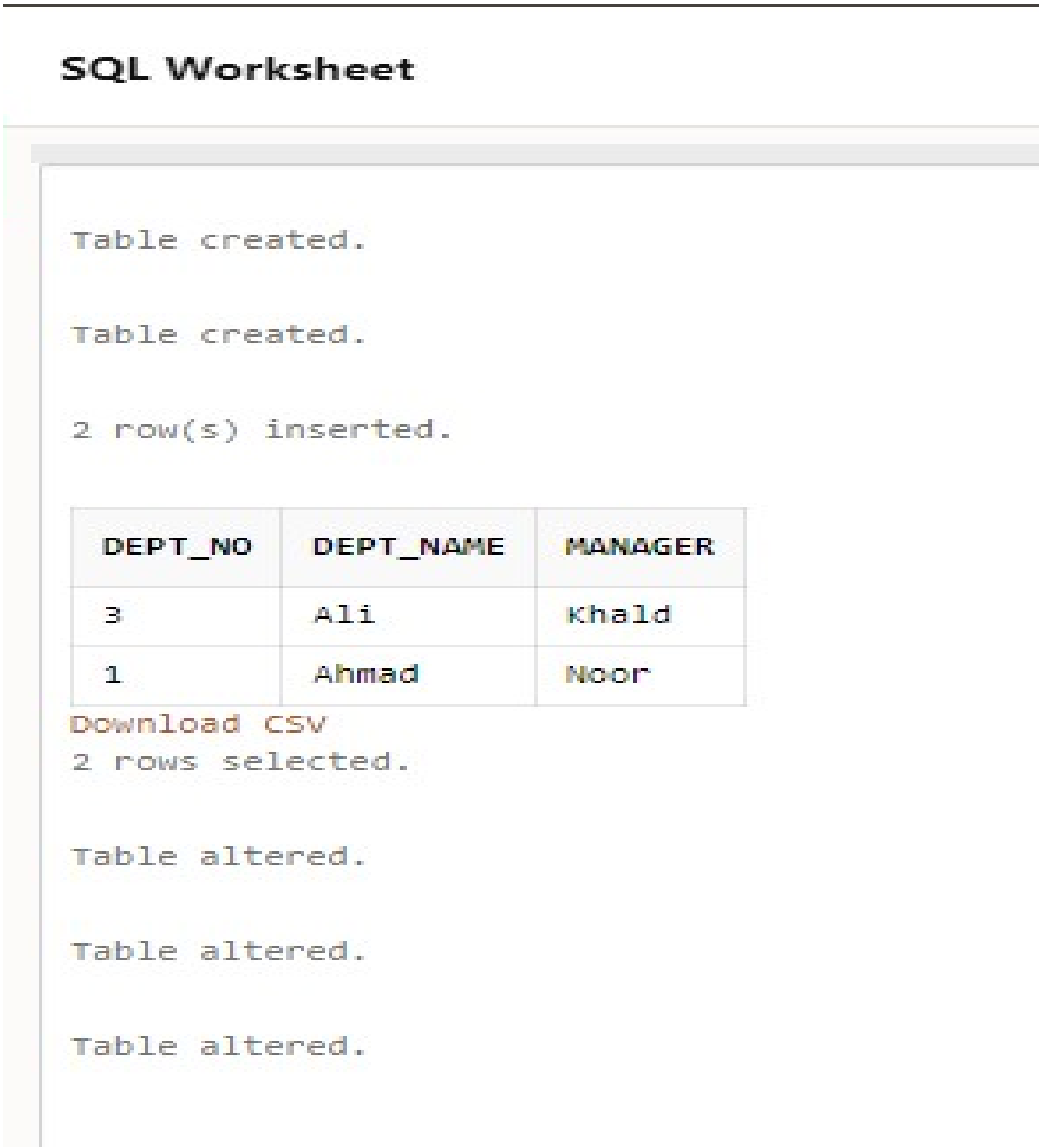
```
create table Student (
S_ID varchar2(6),
S_name varchar2(10) not null,
D_No number(2),
phone number(10),
constraint ID_pk primary key (S_ID),
constraint p_u unique (phone)
);

create table Department (
Dept_No number(2),
Dept_name varchar2(20),
Manager varchar2(6),
constraint No_pk primary key (Dept_No),
constraint n_u unique (Dept_name)
);

insert all
into Department (Dept_No, Dept_name, Manager) values (3,'Ali', 'Khald')
into Department (Dept_No, Dept_name, Manager) values (1,'Ahmad', 'Noor')
select * from dual;

select * from Department;

alter table Student
add constraint d_fk foreign key (D_No) references Department (Dept_No);

alter table Department
rename column Dept_name to Dname ;
```

alter table Student

drop constraint p_u ;



```
create table Student (
S_ID varchar2(6),
S_name varchar2(10) not null,
D_No number(2),
phone number(10),
constraint ID_pk primary key (S_ID),
constraint p_u unique (phone)
);

create table Department (
Dept_No number(2),
Dept_name varchar2(20),
Manager varchar2(6),
constraint No_pk primary key (Dept_No),
constraint n_u unique (Dept_name)
);

insert all
into Department (Dept_No, Dept_name, Manager) values (3,'Ali', 'Khald')
into Department (Dept_No, Dept_name, Manager) values (1,'Ahmad', 'Noor')
select * from dual;

select * from Department;

alter table Student
add constraint d_fk foreign key (D_No) references Department (Dept_No);

alter table Department
rename column Dept_name to Dname ;

alter table Student
drop constraint p_u ;
```



SQL Worksheet

```
Table created.

Table created.

2 row(s) inserted.
```

DEPT_NO	DEPT_NAME	MANAGER
3	Ali	Khald
1	Ahmad	Noor

```
Download CSV
2 rows selected.

Table altered.

Table altered.

Table altered.
```

IS 2511 – Fundamentals of Database Systems

Lab assignment # 3

Answer the following questions by using **(EMP and DEPT) script** then copy your code to a text document (with print screens) .

A- Run (EMP and DEPT) script.

B- Then apply the following queries:

1- Retrieve all the **names** of **employees** who got salary **less than 3000$.**

2- Retrieve all the employees who got salary **in the range 1000$ to 2000$.**

3- Retrieve all **the numbers and salaries** of **employee** for who **got salaries out of the range 2000$ to 3000$** .

4- Retrieve all the employees who got **job: 'ANALYST' , 'SALESMAN' or 'MANAGER'** using the **IN operator**.

5- Retrieve all the employees who **got job other than : 'ANALYST' , 'SALESMAN' or 'MANAGER'** using the **IN operator.**

6- Retrieve all the employees who **in department number 10 or 30 and got salary greater than 1500$.**

7- Retrieve **employee job, first name**, and display as **Position** which is the **concatenation of his job and first name** together.

```
select ename from emp
where sal<3000;

select * from emp
where sal between 1000 and 2000;

select sal, empno from emp
where sal not between 2000 and 3000;

select * from emp
where job in ('ANALYST','SALESMAN','MANAGER');

select * from emp
where job not in ('ANALYST','SALESMAN','MANAGER');

select * from emp
where (deptno = 10 or deptno = 30 )
and(sal > 1500 );

select job ||' '|| ename as position
from emp;
```