Day 3

Assignments

1. Create table with name of emp2 based on employees table (with no data)- (use insert with subquery) to populate the emp2 table using a select statement from the employees table for the employees in department 60.

```
-- Create emp2 table with no data

create table emp2
as select * from employees where 1=2

-- Insert data from employees in department 60

insert into emp2
select * from employees where DEPARTMENT_ID=60
------
select * from emp2
```

2. A) Create the DEPARTMENT table based on the following table instance chart. Create table command with 2 columns

COLUMN NAME	ID	NAME
Default value	1	
DATATYPE	Number	Varchar2
LENGTH	7	25

```
Create table DEPARTMENT
(ID Number(7)  default  '1',
    NAME Varchar2(25) )
```

b) Populate the DEPARTMENT table with data from departments table. Include only columns that you need. (insert using sub query)

```
insert into DEPARTMENT
select DEPARTMENT_ID, DEPARTMENT_NAME from DEPARTMENTS
select * from DEPARTMENT
```

c) Add column 'Loc_name' to table department. (varchar2 100)

```
alter table DEPARTMENT
add Loc_name Varchar2(100)
commit
```

d) Truncate table department.

```
Truncate table DEPARTMENT
```

3. **Create table employee_bkp based on the structure of the employees table (Structure with data).**
   Include only the employee_id, last_name, email, salary and department_id columns

Change using alter (Employee_id Primary key, email unique)

```
create table employee_bkp
as select EMPLOYEE_ID, LAST_NAME, EMAIL, SALARY, DEPARTMENT_ID from
employees where 1=2
insert into employee_bkp
select EMPLOYEE_ID, LAST_NAME, EMAIL, SALARY, DEPARTMENT_ID  from
employees
commit
alter table employee_bkp
modify EMPLOYEE_ID primary key
alter table employee_bkp
modify EMAIL unique
-----
alter  table employee_bkp
modify (EMPLOYEE_ID primary key,
EMAIL unique)

select * from employee_bkp
```

4. **Create a view called EMP_VU based on the employee number, employee last name, and department number from the EMPlOYEES table. Change the heading for the last name to title_name**

```
create or replace view EMP_VU
as
select EMPLOYEE_ID, LAST_NAME as tilte_name , DEPARTMENT_ID
from employees
select * from EMP_VU
```

5. **Create the following tables using ddl**
   Trainers [tr_id, tr_name, tr_mobile], Courses [crs_id, crs_name, crs_price]
   Use Many to Many relationship;
   Solve using create tables, then alter trainers and add email column
   then alter again to add unique constraints;
   Use insert to set those data
   Trainer [aly] > teach [php – oracle – java]
   Trainer [Mohamed] > teach [oracle]
   Trainer [Omar] > teach [oracle – java]
   Then select the data using inner join

```
create table Trainers
( tr_id Number(7) ,
tr_name  Varchar2(100),
tr_mobile Number(15) )
alter table Trainers
modify tr_id primary key
alter table Trainers
add email Varchar2(100)
commit
alter table Trainers
modify email unique
commit
----
select * from  Trainers
---
create table Courses
( crs_id Number(7),
crs_name  Varchar2(100),
crs_price Number(15))
alter table Courses
modify crs_id primary key
----
create table Courses_trainers
(crs_id Number(7),
 tr_id Number(7))
 select * from Courses_trainers
-----
insert into Trainers (tr_id,tr_name, tr_mobile)
values (1, 'Aly', '0145256')
insert into Trainers (tr_id,tr_name, tr_mobile)
values (2, 'Mohammed', '01456256')
insert into Trainers (tr_id,tr_name, tr_mobile)
values (3, 'Omar', '01756256')
commit
insert into Courses (crs_id, crs_name, crs_price)
values (1, 'php', '250')
insert into Courses (crs_id, crs_name, crs_price)
values (2, 'oracle', '300')
insert into Courses (crs_id, crs_name, crs_price)
values (3, 'java', '250')

select * from Courses_trainers

alter table Courses_trainers
modify tr_id constraint tr_id_fk references Trainers (tr_id)

alter table Courses_trainers
modify crs_id constraint crs_id_fk references Courses (crs_id)

select * from Trainers t, Courses c, Courses_trainers cs
```

```sql
where T.TR_ID=cs.tr_id
      and cs.crs_id=C.CRS_ID
```