Database Systems

Fall 2020

LAB – 08

The objective of this lab is to:

- DML (Create Table, Alter Table Statements)

Course & Lab Instructor: Sir Asif Sohail

Instructions:

- Work on this lab individually. Discussion is not allowed.
- Evaluation of tasks will be conducted in lab.
- Anyone caught being indulged in the act of plagiarism would be awarded an "F" grade in this lab.
- Evaluation will be considered final and you cannot debate for the marks. So, focus on performing the tasks when the time is given to you.
- **Allowed time: 1 hour and 20 minutes**
- Best of Luck!

Note: You will be using following tables in your lab tasks.

- EMP (EMPNO, ENAME, JOB, SAL, HIREDATE, COMM, MGR, DEPTNO)
- DEPT (DEPTNO, DNAME, LOC)
- SALGRADE (Grade, HISAL, LOSAL)

- **Perform the following tasks**

Task 01:

In continuation of the last lab, perform the following tasks:

1. Alter table **product** and add **qty_in_hand** field with data type NUMBER(5).

   Solution:

   Alter table product

   add qty_in_hand number(5);

```
Table altered.
```

2. Create the following tables:

Table Name: **vendor**		
Col Name	**Data Type**	**constraints**
v_code	CHAR(10)	Primary Key
v_name	VARCHAR2(15)	Not Null
v_addr	VARCHAR2(20)	

Solution:

```
Create table vendors (

   v_code char(10) constraint vendor_pk primary key,

   v_name varchar2(15) constraint vendor_nn not null,

   v_addr varchar2(20)

)
```

```
Table created.
```

Table Name: **shipment**		
Col Name	**Data Type**	**constraints**
shipment#	NUMBER(5)	Primary Key
ship_date	DATE	DEFAULT sysdate
prod#	NUMBER(5)	Primary Key
qty_delivered	NUMBER(4)	

Solution:

```
Create table shipment (

    shipment# number(5),

    shipdate date default sysdate,
    prod# number(5),

    qty_delivered number(4),

    constraint shipment_pk primary key (shipment#, prod#)

)
```

```
Table created.
```

3. Insert the following 3 rows in **customer** table.

CUST#	Customer Name	CITY	NIC	PHNO
54321	Afaq	Lahore	35202-25679022-3	0324-5227602
54355	Osama	Lahore	35202-13645022-3	0316-3242412
54225	Usama	Gulu	31211-13645111-1	0320-5201211

Solution:

[Optional Query]

```
alter table customer

rename column "Customer Name" to c_name
```

```
insert into customer(cust#, c_name, city, nic, phno) values (54321, 'Afaq', 'Lahore', '35202-25679022-3', '0324-5227602')
1 row(s) inserted.
```

```
insert into customer(cust#, c_name, city, nic, phno) values (54355, 'Osama', 'Lahore', '35202-13645022-3', '0316-3242412')
1 row(s) inserted.
```

```
insert into customer(cust#, c_name, city, nic, phno) values (54225, 'Usama', 'Gulu', '31211-13645111-1', '0320-5201211')
1 row(s) inserted.
```

OR

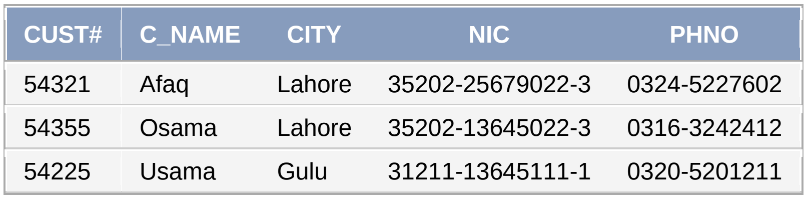
```
insert into customer(cust#, "Customer Name", city, nic, phno) values (54321, 'Afaq', 'Lahore', '35202-25679022-3', '0324-5227602')
1 row(s) inserted.
```

- insert into customer(cust#, "Customer Name", city, nic, phno) values (54355, 'Osama', 'Lahore', '35202-13645022-3', '0316-3242412')

```
1 row(s) inserted.
```

- insert into customer(cust#, "Customer Name", city, nic, phno) values (54225, 'Usama', 'Gulu', '31211-13645111-1', '0320-5201211')

```
1 row(s) inserted.
```



CUST#	C_NAME	CITY	NIC	PHNO
54321	Afaq	Lahore	35202-25679022-3	0324-5227602
54355	Osama	Lahore	35202-13645022-3	0316-3242412
54225	Usama	Gulu	31211-13645111-1	0320-5201211

4. Validate all the integrity constraints of **customer** table by attempting to insert a row that violates the integrity constraint.

   Solution:

   Insert into customer(cust#, c_name, city, nic, phno) values(54321,Null, Null, '35202-25678899734',0306-4416475)

   a) Unique constraint violated (Primary key must be unique) Customer.cust#

   b) C_name can’t be null (c_name not null constraint)

   c) City can’t be null (not null constraint) Lab07 Question 3

   d) Value larger than expected in column (cust#)

   ```
   value too large for column actual 17 characters max 16 allowed
   ```

   e) Datatype wrong number given instead of char

5. Insert your data in the **customer** table.

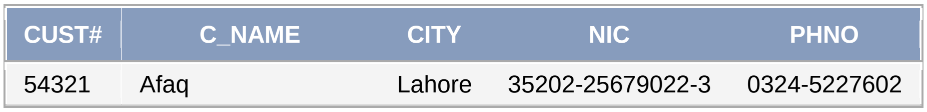
   Solution:

   I have already added my name record above, so I am going my fav. student name record :)

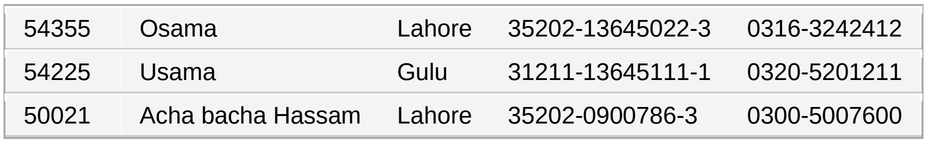
   Insert into customer (cust#, "Customer Name", city, nic, phno) values (50021, 'Acha bacha Hassam', 'Lahore', '35202-0900786-3', '0300-5007600')

6. View the contents of the **customer** table.

   Select * from customer



CUST#	C_NAME	CITY	NIC	PHNO
54321	Afaq	Lahore	35202-25679022-3	0324-5227602



54355	Osama	Lahore	35202-13645022-3	0316-3242412
54225	Usama	Gulu	31211-13645111-1	0320-5201211
50021	Acha bacha Hassam	Lahore	35202-0900786-3	0300-5007600

7. COMMIT

Solution:

```
Statement processed.
```

Permanently store all the DML operations uptil now.

8. Insert three rows in each of invoice, product, invoice_details, vendor, and shipment table. Use default values of the columns in two rows (where applicable). Use 'Mon dd,yyyy' date format in two rows for each inv_date and ship_date.

Product Table

INSERT into Product VALUES(01,'Mobile',50000,5)

```
1 row(s) inserted.
```

INSERT into Product VALUES(02,'Laptop',75000,2)

```
 1 row(s) inserted.
```

Invoice Table

Insert into invoice values(1,to_char(sysdate,'Mon dd,yyyy'),54321,'Credit Card')

```
1 row(s) inserted
```

Insert into invoice values(2,to_char(sysdate,'Mon dd,yyyy'),50021,'Credit Card')

```
1 row(s) inserted
```

Invoice_Details Table

Insert into invoice_details(invoice#, prod#, qty_ordered) values(1,2,3)

```
     1 row(s) inserted.
```

Insert into invoice_details(invoice#, prod#, qty_ordered) values(2,1,4)

```
     1 row(s) inserted.
```

Vendor Table

```
Insert into vendors values('vendor_01','Abdullah','Wapda Town Lahore')

 1 row(s) inserted.

 Insert into vendors values('vendor_02','Ahmed','DHA, Karachi')
 1 row(s) inserted.
```

Shipment Table

```
    insert into shipment(shipment#, shipdate, prod#, qty_delivered) values (100,
to_char(sysdate, 'Mon dd,yyyy'), 1,3)

1 row(s) inserted.
```

```
    insert into shipment(shipment#, shipdate, prod#, qty_delivered) values (101,
to_char(sysdate, 'Mon dd,yyyy'), 2,4)

1 row(s) inserted.
```

9. View the contents of the tables.

Solution:

Select * from customer

CUST#	C_NAME	CITY	NIC	PHNO
54321	Afaq	Lahore	35202-25679022-3	0324-5227602
54355	Osama	Lahore	35202-13645022-3	0316-3242412
54225	Usama	Gulu	31211-13645111-1	0320-5201211
50021	Acha bacha Hassam	Lahore	35202-0900786-3	0300-5007600

Select * from product

PROD#	P_NAME	PRICE	QTY_IN_HAND
1	Mobile	50000	5
2	Laptop	75000	2

Select * from invoice

INVOICE#	INV_DATE	CUST#	PAYMENT
1	05/25/2022	54321	Credit Card
2	05/25/2022	50021	Credit Card

Select * from invoice_details

INVOICE#	PROD#	QTY_ORDERED
1	2	3
2	1	4

Select * from vendors

V_CODE	V_NAME	V_ADDR
vendor_01	Abdullah	Wapda Town Lahore
vendor_02	Ahmed	DHA, Karachi

Select * from shipment

SHIPMENT#	SHIPDATE	PROD#	QTY_DELIVERED
100	05/25/2022	1	3
101	05/25/2022	2	4

10. COMMIT

    Solution:

    Store all the changes done by DML operations permanently

11. Delete a row from the **shipment** table using the primary key of the table

    Solution:

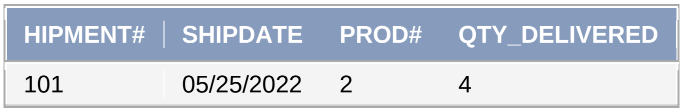
    Delete from shipment where shipment# = 100 and prod# = 1

```
      1 row(s) deleted.
```

12. View the contents of the **shipment** table

    Solution:

    Select * from shipment



HIPMENT#	SHIPDATE	PROD#	QTY_DELIVERED
101	05/25/2022	2	4

13. ROLLBACK

    Solution:

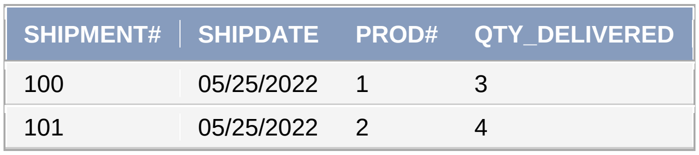
    Rollback the changes to previous commit (Q 10) (undo the DML operations of 11, 12)

```
      Statement processed.
```

14. View the contents of the **shipment** table.

    Solution:

    Select * from shipment



SHIPMENT#	SHIPDATE	PROD#	QTY_DELIVERED
100	05/25/2022	1	3
101	05/25/2022	2	4

15. Increase the price by 15% of the current price of all the products with price less than 100 in the **product** table.

    Solution:

```
update product
set price = price + price * 0.15
where price <100
```

```
0 row(s) updated.
Because I haven't any record having price less than 100 ☹
```

16. Set the price of a certain product equal to the price of some other product on the basis of prod#.

    Solution:

    ```
    update product
    set price = (Select price from product where prod#=1)
    where prod#=2
    ```

    ```
    1 row(s) updated.
    ```

17. SAVEPOINT S1

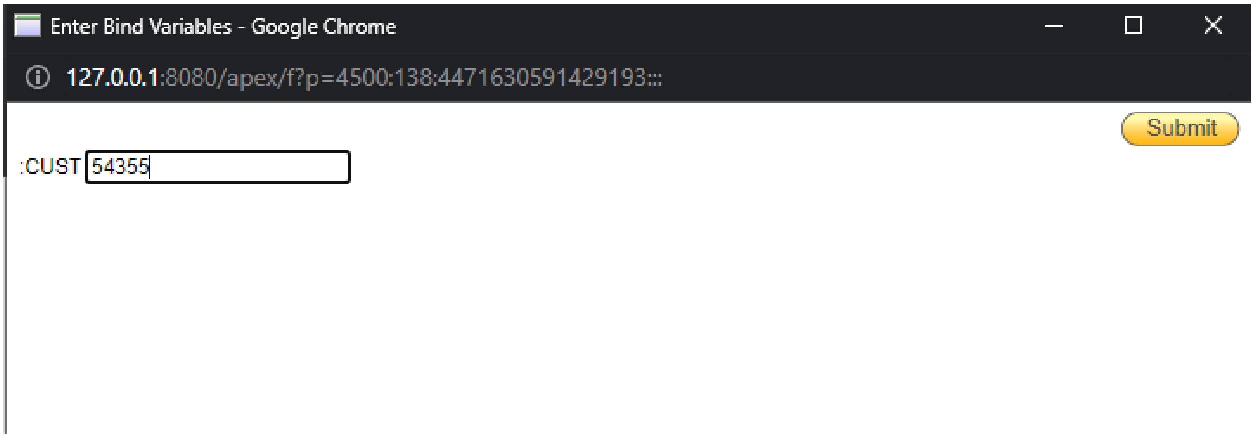
    Create a savepoint named ‘S1’

    Act as a marking point in a transaction (DML operations) so we can rollback to any savepoint in a transaction easily.

18. Change the phno of a customer for a given cust# in the customer table.

    Solution:

    ```
    Update customer set phno = 0900-78601
    Where cust#=:cust
    ```


Enter Bind Variables - Google Chrome
127.0.0.1:8080/apex/f?p=4500:138:4471630591429193:::
Submit

:CUST 54355

```
1 row(s) updated.
```

19. SAVEPOINT S2

    Solution:

    Create a savepoint s2 (act as a marker in a transaction so we can easily rollback to it)

20. Delete a row from the invoice_details table using the primary key of the table.

    Solution:

    ```
    Delete from invoice_details
    where invoice#=1 and prod#=2
    ```

21. ROLLBACK TO S1

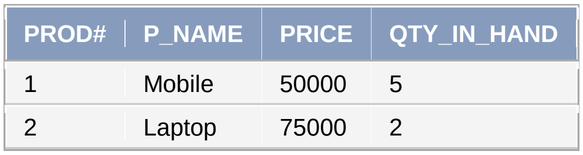
Solution:

```
        Statement processed
Unodo all the (DML Operations)changes upto the savepoint S1
```

22. View the contents of the **product, customer** and **invoice_details** table.

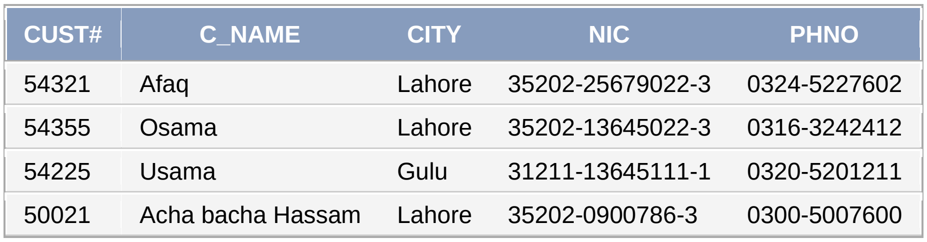
Solution:

Select * from product



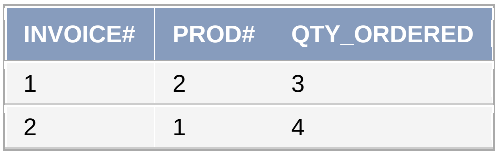
PROD#	P_NAME	PRICE	QTY_IN_HAND
1	Mobile	50000	5
2	Laptop	75000	2

Select * from customer



CUST#	C_NAME	CITY	NIC	PHNO
54321	Afaq	Lahore	35202-25679022-3	0324-5227602
54355	Osama	Lahore	35202-13645022-3	0316-3242412
54225	Usama	Gulu	31211-13645111-1	0320-5201211
50021	Acha bacha Hassam	Lahore	35202-0900786-3	0300-5007600

Select * from invoice_details



INVOICE#	PROD#	QTY_ORDERED
1	2	3
2	1	4

23. COMMIT.

Solution:

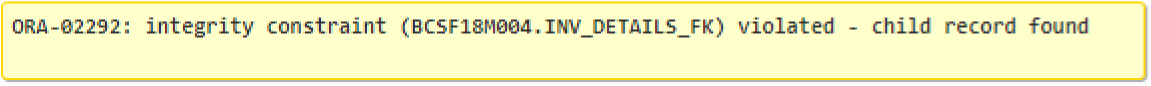
```
Statement processed
```

Save all the changes permanently done before savepoint s1

24. What is response when an attempt is made to delete a row from the **product** table that has matching rows in the **invoice_details** table.

Solution:

delete from product
where prod#=1



```
ORA-02292: integrity constraint (BCSF18M004.INV_DETAILS_FK) violated - child record found
```

Record can't be deleted from parent table product because it is referred in child table Invoice_details.

25. Alter the foreign key constraint on prod# in the **invoice_details** table and set it to cascaded deletion.

Solution:

Check constraint Name from user_constraints table:

```
Select * from user_constraints
where table_name = 'INVOICE_DETAILS'
```

OWNER	CONSTRAINT_NAME	CONSTRAINT_TYPE	TABLE_NAME	SEARCH_CONDITION	R_OWNER	R_CONSTRAINT_NAME	DELETE_RULE
BCSF18M004	QT_RANGE	C	INVOICE_DETAILS	qty_ordered between 1 and 500	-	-	-
BCSF18M004	INVOICE_DETAIL_PK	P	INVOICE_DETAILS	-	-	-	-
BCSF18M004	INV_DET_FK	R	INVOICE_DETAILS	-	BCSF18M004	SYS_C007111	NO ACTION
BCSF18M004	INV_DETAILS_FK	R	INVOICE_DETAILS	-	BCSF18M004	PRODUCT_PK	NO ACTION

Drop Constraint

```
alter table invoice_details
disable constraint inv_details_fk
```

```
 Table altered.
```

```
Add Constraint Again:
```

```
alter table invoice_details
add constraint inv_details_fk foreign key(prod#) references product(prod#) on delete cascade
```

```
Table altered.
```

26. Check the response when an attempt is made to delete a row from the product table that has matching rows in the **invoice_details** table.

Solution:

```
delete from product
where prod#=1
```

```
    1 row(s) deleted.
```

Record will be deleted from parent and child table

```
Select * from product
```

PROD#	P_NAME	PRICE	QTY_IN_HAND
2	Laptop	75000	2

Select * from invoice_details

INVOICE#	PROD#	QTY_ORDERED
1	2	3

27. ROLLBACK

Solution:

Undo all the changes done by dml operations upto the last commit

28. COMMIT.

Solution:

```
Statement processed.
```

0.00 seconds

No DML operation saved as it is already at last commit point

29. Write a query to display cust#, c_name, invoice#, inv_date, prod#, p_name, price, qty_ordered, and total. (total = price*qty_ordered)

Solution:

Select c.cust#, c.c_name, i.invoice#, i.inv_date, p.prod#, p.p_name, p.price, s.qty_ordered, p.price * s.qty_ordered as total

from customer c, invoice i, product p, invoice_details s

where c.cust# = i.cust# and p.prod# = s.prod#

CUST #	C_NAME	INVOICE #	INV_DATE	PROD #	P_NAME	PRICE	QTY_ORDERED	TOTAL
54321	Afaq	1	05/25/2022	2	Laptop	75000	3	225000
50021	Acha bacha Hassam	2	05/25/2022	2	Laptop	75000	3	225000

30. Write a query to drop the constraint added by you in task 25

Solution:

Alter table invoice_details

drop constraint inv_details_fk

Task 02: (Marks:10)

1. Delete all the employees from emp table belonging to RESEARCH department.

   Solution:

```
delete emp
where deptno=(select deptno from dept where dname='RESEARCH')
```

```
5 row(s) deleted.
```

2. Add a record of your name in the emp table and then update its
   - salary equal to president
   - same job to BLAKE

Solution:

a)

```
Insert Into emp
values(7000,'Osama','MANAGER',7839,Sysdate,25000,1000,10);
```

b)

```
update emp set sal =(select sal from emp where ename='PRESIDENT') , job = (Select job
from emp where ename like 'BLAKE')
WHERE ename='OSAMA';
```

3. Update sal and dept of JONES to that of blake.

Solution:

```
Update emp

Set (sal,deptno)=(select sal, deptno from emp where ename like 'BLAKE')

Where ename like 'JONES'
```

4. Make a table CLERK having the following using the following schema

   <u>CLERK(id, name, salary, hiredate)</u>

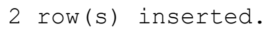
Table Name: CLERK		
Col Name	**Data Type**	**Constraints**
Id	NUMBER(4)	Primary Key
Name	VARCHAR2(10)	Not Null
Salary	Number(7,2)	
hiredate	Date	

Solution:

```
create table CLERK(
  id number(4) constraint clerk_pk PRIMARY KEY,
  NAME VARCHAR2(10) constraint clerk_nn NOT NULL,
  salary number(7,2),
  hiredate date
)
```

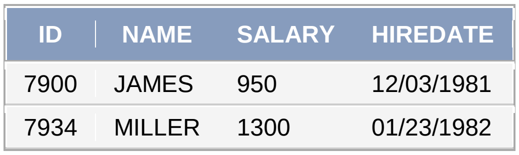
5. Add all the clerks from the EMP table to CLERK table using subquery-based insert.
   Solution:

```
insert into clerk (id,name,salary,hiredate)
select empno,ename,sal,hiredate from emp where job='CLERK'
```



```
2 row(s) inserted.
```

Select * from clerk



ID	NAME	SALARY	HIREDATE
7900	JAMES	950	12/03/1981
7934	MILLER	1300	01/23/1982