Lab Assignment # 1

DDL, DML, Constraints and Transaction Processing

Charu Bisht

Task 1:

Create a sequence object with the name **my_seq**. It should start with 1 and increase by 1. The sequence method NEXTVAL returns a numeric data type.

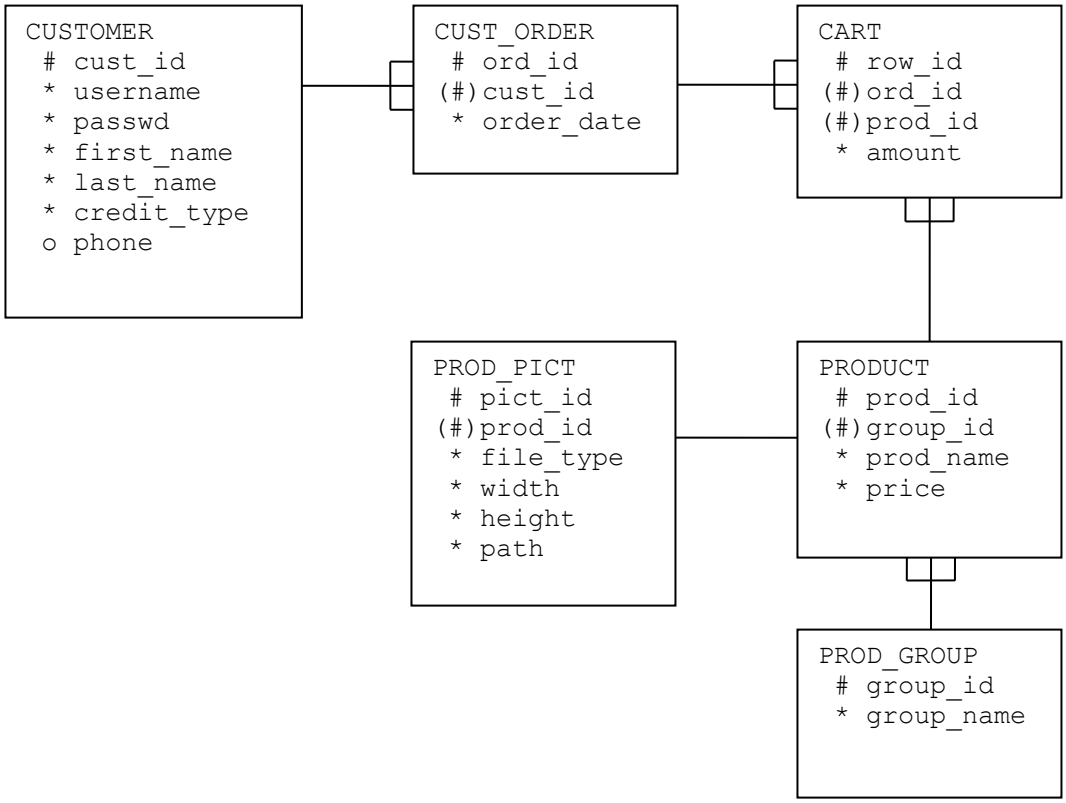
Solution 1:

```
Create sequence with start value 1, increment by 1
```

```
create sequence my_seq start with 1 increment by 1;
```

Task 2:

Create a table structure according to the drawing below:



Explanation of notation:

= Primary key
(#)= Foreign key
* = Mandatory (must contain a value => NOT NULL)
o = Optional (must not contain a value can be NULL)

customer.credit_type CHECK ('high','average','low')
prod_pict.file_type CHECK ('gif','jpg')
cust_order.ord_id (generated by the sequence my_seq)
cart.row_id (generated by the sequence my_seq)
cust_order.order_date (data type = DATE, SYSDATE)
customer.username (should be unique, constraint UNIQUE)
All Foreign Key columns should have the column constraint **NOT NULL**

Declare all constraints except NOT NULL at the table level!
Suggestion for a constraint naming convention: **table_column_constraint**.
You can use the following abbreviations if you like: **CK** = CHECK, **PK** = PRIMARY KEY, **FK** = FOREIGN KEY and finally **UQ** = UNIQUE, or whatever you like as long as you are consistently.
For the customer table above, a primary key constraint would be named: **customer_cust_id_pk**

Solution 2:

Create CUSTOMER table with required columns

```
create table customer (
 cust_id number(10),
 username varchar2(20) not null,
 passwd varchar2(20) not null,
 first_name varchar2(30) not null,
 last_name varchar2(30) not null,
 credit_type varchar2(12) not null,
 phone number(10));
```

Alter CUSTOMER table to add required constraint

```
alter table customer
add constraint customer_cust_id_pk primary key(cust_id)
add constraint customer_credit_type_ck check (credit_type in
('high','average','low'))
add constraint customer_username_uq unique(username);
```

Create CUST ORDER table with required columns

```
create table cust_order(
 ord_id number(10),
 cust_id number(7) not null,
 order_date date not null);
```

Alter CUST ORDER table to add required constraint

```
alter table cust_order
add constraint cust_order_ord_id_pk primary key(ord_id)
add constraint cust_order_cust_id_fk foreign key(cust_id) references
customer(cust_id);
```

Create PROD GROUP table with required columns

```
create table prod_group(
 group_id number(10),
 group_name varchar2(50) not null);
```

Alter PROD GROUP table to add required constraint

```
alter table prod_group
add constraint prod_group_group_id_pk primary key(group_id);
```

Create PRODUCT table with required columns

```
create table product(
 prod_id number(10),
 group_id number(10) not null,
 prod_name varchar2(50) not null,
 price number(9,2) not null);
```

Alter PRODUCT table to add required constraint

```
alter table product
add constraint product_prod_id_pk primary key(prod_id)
add constraint product_group_id_fk foreign key(group_id) references
prod_group(group_id);
```

Create CART table with required columns

```
create table cart(
 row_id number(10),
 ord_id number(10) not null,
 prod_id number(10) not null,
```

```
 amount number(9,2) not null);
```

Alter CART table to add required constraint

```
alter table cart
add constraint cart_row_id_pk primary key(row_id)
add constraint cart_ord_id_fk foreign key(ord_id) references cust_order(ord_id)
add constraint cart_prod_id_fk foreign key(prod_id) references product(prod_id);
```

Create PROD PICT table with required columns

```
create table prod_pict(
 pict_id number(10),
 prod_id number(10) not null,
 file_type varchar2(5) not null,
 width number(10) not null,
 height number(10) not null,
 path varchar2(50) not null);
```

Alter PROD PICT table to add required constraint

```
alter table prod_pict
add constraint prod_pict_pict_id_pk primary key(pict_id)
add constraint prod_pict_prod_id_fk foreign key(prod_id) references
product(prod_id)
add constraint prod_pict_file_type_ck check (file_type in ('gif','jpg'));
```

Task 3:

Insert three rows in the **customer** table.

Solution 3

Inserting 3 rows

```
 insert into customer values(1,'billy','billy121','billy','jones','high',454515);
 insert into customer values(2,'tim','tim828','tim','singh','low',145454);
 insert into customer values(3,'jason','jase3','jason','brad','average', null);
```

Task 4:

Insert two rows in the **prod_group** table.

Solution 4:

Inserting 2 rows

```
insert into prod_group values(1000,'Accessories');
insert into prod_group values(2000,'Clothing');
```

Task 5:

Insert two rows in the **product** table.

Solution 5:

```
insert into product values(10001,1000,'Jeans',150.90);
insert into product values(10002,2000,'Watch',599.00);
```

Task 6:

Perform a sale by creating **one row** in the **cust_order** table and **two rows** in the **cart** table.
Remember to use the sequence to generate primary key in the tables.
NOTE that when you have created the cust_order you must check what value the sequence put in the ord_id column (i.e. the Primary Key value). Then take that number and use it in the insert on the cart table FK-column. **DO NOT USE** the sequence to generate a number to the foreign key ord_id in the cart table!

Solution 6:

```
Insert 1 row into cust order table using seq created in Task 1
insert into cust_order values(my_seq.nextval,1,sysdate);

Insert 1 record into cart table using my seq sequence for PK
insert into cart values(my_seq.nextval,1,10001,1);
insert into cart values(my_seq.nextval,1,10002,2);
```

Task 7:

Increase the price on all articles by 12%.

Solution 7:

```
Update column price by 12%
update product set price = price*1.12;
```

Task 8:

Update the phone number for an optional customer.

Solution 8:

```
Update column phone
update customer set phone = 446548 where cust_id = 1;
```

Task 9:

Delete all rows from the cust_order table, by using DML. **What happens and why!**

Solution 9:

Deleting all rows from the cust_order

```
delete from cust_order;
```

Output:

We face below error when attempting to delete data from cust_order table.

```
ORA-02292: integrity constraint (SQL_IQGXEAECUVYGBCVCBWIGMTFTQ.CART_ORD_ID_FK) violated - child record found ORA-06512: at "SYS.DBMS_SQL", line 1721
```

The error suggests that the foreign key constraint was violated. According to the error, we cannot drop values from a parent table until all corresponding values in the child table are dropped. This property is also referred to as referential integrity.
To get rid of the error we will have to first drop the values from the child table followed by the parent table.