

## Database Schema for a customer-sale scenario

Customer(Cust id : integer, cust\_name: string)

Item(item\_id: integer, item\_name: string, price: integer)

Sale(bill\_no: integer, bill\_data: date, cust\_id: integer,  
item\_id: integer, qty\_sold: integer)

### 1. Create the tables with the appropriate integrity constraints.

```
create table CUSTOMER(  
  cust_id int,  
  cust_name varchar(50),  
  primary key(cust_id)  
);  
create table ITEM(  
  item_id int primary key,  
  item_name varchar(50),  
  price int not null  
);  
create table SALE(  
  bill_no int primary key,  
  bill_date date,  
  cust_id int not null,  
  item_id int not null,  
  qty_sold int,  
  foreign key(cust_id) references CUSTOMER(cust_id),  
  foreign key(item_id) references ITEM(item_id)  
);
```

Output: Table created

### 2. Insert around 10 records in each of the tables.

```
Insert into CUSTOMER values (101,  
'bhavini');  
Insert into CUSTOMER values (102,  
'karunya');  
Insert into CUSTOMER values (103,  
'yashika');  
Insert into CUSTOMER values (104,  
'aditya');  
Insert into CUSTOMER values (105,  
'nehal');  
Insert into CUSTOMER values (106,  
'ishita');  
Insert into CUSTOMER values (107,
```

'titiksha');

Insert into CUSTOMER values (108,

'anshee');

Insert into CUSTOMER values (109, 'anubhuti');

Insert into CUSTOMER values (110, 'manav');

SELECT \* FROM CUSTOMER;

	cust_id	cust_name
1	101	bhavini
2	102	karunya
3	103	yashika
4	104	aditya
5	105	nehal
6	106	ishita
7	107	titiksha
8	108	anshee
9	109	anubhuti
10	110	manav

Insert into ITEM values (201, 'Notebook', 50);

Insert into ITEM values (202, 'Register', 10);

Insert into ITEM values (203, 'Geometry box', 10);

Insert into ITEM values (204, 'Pencil', 20);

Insert into ITEM values (205, 'Pencil box', 50);

Insert into ITEM values (206, 'Pen', 100);

Insert into ITEM values (207, 'Graph paper', 100);

Insert into ITEM values (208, 'Fevicol', 20);

Insert into ITEM values (209, 'Glaze paper', 40);

Insert into ITEM values (210, 'Rubber', 60);

SELECT \* FROM ITEM;

	item_id	item_name	price
1	201	Notebook	50
2	202	Register	10
3	203	Geometry box	10
4	204	Pencil	20
5	205	Pencil box	50
6	206	Pen	100
7	207	Graph	100
8	208	Fevicol	20
9	209	Glaze paper	40
10	210	Rubber	60

Insert into SALE values (301, '22-SEP-2020', 110,201, 2);  
 Insert into SALE values (302, '21-SEP-2020', 109,202, 5);  
 Insert into SALE values (303, '20-SEP-2020', 108,203, 2);  
 Insert into SALE values (304, '19-SEP-2020', 107,204, 3);  
 Insert into SALE values (305, '18-SEP-2020', 106,205, 5);  
 Insert into SALE values (306, '17-SEP-2020', 105,206, 3);  
 Insert into SALE values (307, '16-SEP-2020', 104,207, 1);  
 Insert into SALE values (308, '15-SEP-2020', 103,208, 2);  
 Insert into SALE values (309, '14-SEP-2020', 102,209, 3);  
 Insert into SALE values (310, '13-SEP-2020', 101,210, 1);  
 SELECT \* FROM SALE;

	bill_no	bill_date	cust_id	item_id	qty_sold
1	301	22-SEP-2020	110	201	2
2	302	21-SEP-2020	109	202	5
3	303	20-SEP-2020	108	203	2
4	304	19-SEP-2020	107	204	3
5	305	18-SEP-2020	106	205	5
6	306	17-SEP-2020	105	206	3
7	307	16-SEP-2020	104	207	1
8	308	15-SEP-2020	103	208	2
9	309	14-SEP-2020	102	209	3
10	310	13-SEP-2020	101	210	1

**3. List all the bills for the current date with the customer names and item numbers.**

```

SELECT c.cust_name, i.item_id, s.bill_no
FROM CUSTOMER c, ITEM i, SALE s
WHERE c.cust_id = s.cust_id AND
  
```

i.item\_id=s.item\_id  
AND s.bill\_date = to\_char(sysdate);

PRICE	QTY_SOLD	TOTAL
50	2	100
10	5	50
10	2	20
20	3	60
50	5	250
100	3	300
100	1	100
20	2	40
40	3	120
60	1	60

**4. List the total Bill details with the quantity sold, price of the item and the final amount.**

Select i.price,s.qty\_sold,(i.price\*s.qty\_sold)  
Total from Item i, Sale s where i.item\_id=s.item\_id;

PRICE	QTY_SOLD	TOTAL
50	2	100
10	5	50
10	2	20
20	3	60
50	5	250
100	3	300
100	1	100
20	2	40
40	3	120
60	1	60

**5. List the details of the customer who have bought a product which has a price>200.**

```

Select c.cust_id, c.cust_name
from Customer c, Sale s, Item i
Where i.price>200
And c.cust_id=s.cust_id and i.item_id=s.item_id;

```

no data found

**6. Give a count of how many products have been bought by each customer**

```

Select cust_id, count(item_id)
From Sale group by cust_id;

```

CUST_ID	COUNT (ITEM_ID)
108	1
107	1
110	1
109	1
105	1
104	1
103	1
101	1
106	1
102	1

**7. Give a list of products bought by a customer having cust\_id as 5.**

```

Select i.item_name
From Item i, Sale s
Where s.cust_id=5 and i.item_id=s.item_id;

```

**8. List the item details which are sold as of today**

```

Select i.item_id, i.item_name
From Item I, Sale s
Where i.item_id=s.item_id
and s.bill_date=to_char(sysdate);

```

no data found

**9. Create a view which lists out the bill\_no, bill\_date, cust\_id, item\_id, price, qty\_sold, amount**

Create view cust as

(select s.bill\_no, s.bill\_date, c.cust\_id,  
i.item\_id, i.price, s.qty\_sold

From Customer c, Sale s, Item i

Where c.cust\_id=s.cust\_id and

i.item\_id=s.item\_id);

Select \* from cust;

BILL_NO	BILL_DATE	CUST_ID	ITEM_ID	PRICE	QTY_SOLD
301	20-SEP-20	110	201	50	2
302	20-SEP-20	109	202	10	5
303	19-SEP-20	108	203	10	2
304	18-SEP-20	107	204	20	3
305	17-SEP-20	106	205	50	5
306	16-SEP-20	105	206	100	3
307	15-SEP-20	104	207	100	1
308	15-SEP-20	103	208	20	2
309	14-SEP-20	102	209	40	3
310	13-SEP-20	101	210	60	1

**10. Create a view which lists the daily sales date wise for the last one week**

create view view1 as

select (i.price\*s.qty\_sold)total, s.bill\_date

from Customer c,Sale s, Item i

where c.cust\_id=s.cust\_id and i.item\_id=s.item\_id

and bill\_date<='20-SEP-20'