**Inventory Case Study.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Customer | |  | | |  | | |  | | |  | |  | |
| customerid | | customername | | | dateofreg | | | userid | | | password | | Location | |
| Customer Purchase | | | | | |  | | |  | | |  | |
| customerid | | | itemid | | | qtypurchased | | | billId | | | netprice | |
| item |  | | |  | | |  | | |
| itemid | itemname | | | unitprice | | | class | | |

|  |  |  |
| --- | --- | --- |
| Supplier |  |  |
| supplierid | suppliername | suppliercontactno |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Itemorder |  |  |  |  |  |
| Itemid | supplierid | qtyordered | orderdate | deliverystatus | deliverydate |

**Using joins solve:**

1. **Get the name of the customer and the name of the item purchased.**

select c.customername, i.itemname

from customer c join customer purchase cp on c.customerid=cp.customerid

join item i on cp.itemid=i.itemid;

1. **Display the details of the suppliers who have been ordered to supply item, but the delivery status is ‘Not Delivered’.**

select s.\* from supplier s join itemorder io on s.supplierid=io.supplierid

where deliverystatus = 'Not Delivered';

1. **Display the name of the item, name of the supplier and quantity ordered if the supplier has been ordered to supply that item.**

select i.itemname, s.suppliername, io.qtyordered

from item i join itemorder io on i.itemid=io.itemid

join supplier on io.supplierid=s.supplierid;

1. **Display the customer name, item name and unit price of every item purchased by the customer.**

select c.customername, i.itemname, i.unitprice

from customer c join customer purchase cp

on c.customerid=cp.customerid

join item i on cp.itemid=i.itemid;

1. **Amar is a customer who has placed the order, can you please help him get the contact no. of the person who supplies the item which he has ordered.**

select s.suppliercontactno

from customer c join customer purchase cp

on c.customerid=cp.customerid

join itemorder io on cp.itemid=io.itemid

join supplier s on io.supplierid=s.supplierid

where c.customername='Amar'

and io.deliverystatus='Not Delivered';

1. **Mohan has placed an order, can you help him find on which date will he be receiving the order.**

select io.deliverydate

from customer c join customer purchase cp

on c.customerid=cp.customerid

join itemorder io on cp.itemid=io.itemid

where c.customername='Mohan';

1. **For every item class find out the total quantity purchased.**

select i.class, sum(io.qtypurchased) as "Total Quantity Purchased"

from item i join customerpurchase cp on i.itemid=cp.itemid

group by i.class;

1. **For every customer location get the total no. of items ordered.**

select c.location, count(cp.itemid) as "Total no. of Items Ordered"

from customer c join customer purchase cp on c.customerid=cp.customerid

group by c.location;

1. **Display the details of the customer who has purchased more than 1 item.**

select \* from customer c

where c.customerid in (select c.customerid

from customer c join customerpurchase cp

on c.customerid=cp.customerid

group by c.customerid, cp.billid

having count(cp.itemid)>1);

1. **Display the details of the supplier who has been ordered to supply more than 1 item.**

select \* from supplier s

where s.supplierid in (select s.supplierid

from supplier s join itemorder io

on s.supplierid=io.supplierid

group by s.supplierid

having count(io.itemid)>1);

**Using subqueries solve:**

1. **List the name of item whose unit price is greater than the unit price of item id BAK101.**

select itemname from item

where unitprice > (select unitprice from item

where itemid = 'BAK101');

1. **List the details of the items whose unit price is maximum.**

select \* from item

where unitprice = (select max(unitprice) from item);

1. **List the second highest Unit price from the item table.**

select unitprice from item

where unitprice = (select max(unitprice) from item

where unitprice < (select max(unitprice) from item));

1. **List the details of the item whose unit price is more than the average Unit price of items.**

select \* from item

where unitprice > (select avg(unitprice) from item group by class);

1. **List the id of the customer and SUM of total purchase amount for those customers whose SUM of total purchase amount is more than the average of the SUM of total purchase amount of all the customers.**

select customerid, sum(netprice) as "Total Purchase Amount"

from customerpurchase group by customerid

having sum(netprice) > (select avg(sum(netprice))

from customerpurchase group by customerid);

1. **List the details of the customer(s) who have paid a bill of maximum amount.**

select \* from customer

where customerid in (select customerid from customerpurchase

group by (customerid, billid)

having sum(netprice) > (select max(sum(netprice))

from customerpurchase group by (customerid, billid));