Lab File

Database Management System

(CSE 201)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Submitted to:	Submitted by:
Dr Richa Gupta	Shaina Mehta
Ast. Professor	A2305219268
CSE Department, ASET	B.tech. C.S.E.
	5CSE-4Y

AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY

AMITY UNIVERSITY UTTAR PRADESH

NOIDA -201301

Exp No	Assignment Category	Code	Name of Experiment	Date of Allotment	Date of Evaluation	Max Marks	Marks Obtained	Faculty Sign
1	Mandatory Experiment		Lab Assignment 1	29-07-2021	29-09-2021			
2			Lab Assignment 2	05-08-2021	29-09-2021			
3			Lab Assignment 3	12-08-2021	29-09-2021			
4			Lab Assignment 4	19-08-2021	29-09-2021			
5			Lab Assignment 5	26-08-2021	29-09-2021			
6			Lab Assignment 6	02-09-2021	29-09-2021			
7			Lab Assignment 7	09-09-2021	29-09-2021			
8			Lab Assignment 8	16-09-2021	29-09-2021			
9			Lab Assignment 9	06-10-2021	27-10-2021			
10			Lab Assignment 10	13-10-2021	27-10-2021			
11			Lab Assignment 11	20-10-2021	27-10-2021			
12			Lab Assignment 12	27-10-2021	10-11-2021			
	Viva	Viva						

Lab Assignment 1

Q1) Create the following tables:

1. **client_master**

columnname	datatype	size
client_no	varchar2	6
name	varchar2	20
address1	varchar2	30
address2	varchar2	30
city	varchar2	15
state	varchar2	15
pincode	number	6
bal_due	number	10,2

2. **Product_master**

Columnname	datatype	size
Product_no	varchar2	
Description	varchar2	
Profit_percent	number	
Unit_measure	varchar2	
Qty_on_hand	number	
Reoder_lvlnumber		
Sell_price	number	
Cost_price number		

Q2) Insert the following data into their respective tables:

1. **Data for client_master**

Clientno	Name	city	pincode	state	bal.due
0001	Ivan	Bombay	400054	Maharashtra	15000
0002	Vandana	Madras	780001	Tamilnadu	0
0003	Pramada	Bombay	400057	Maharashtra	5000
0004	Basu	Bombay	400056	Maharashtra	0
0005	Ravi	Delhi	100001		2000
0006	Rukmini	Bombay	400050	Maharashtra	0

2. **Data for Product_master**

Product No.	Desciption price	Profit % Percent	Unit measured	Qty on hand	Reorder lvl	Sell price	Cost price
P00001	1.44floppies	5	piece	100	20	525	500
P03453	Monitors	6	piece	10	3	12000	11200
P06734	Mouse	5	piece	20	5	1050	500
P07865	1.22 floppies	5	piece	100	20	525	500
P07868	Keyboards	2	piece	10	3	3150	3050
P07885	CD Drive	2.5	piece	10	3	5250	5100
P07965	540 HDD	4	piece	10	3	8400	8000
P07975	1.44 Drive	5	piece	10	3	1050	1000
P08865	1.22 Drive	5	piece	2	3	1050	1000

Q3) On the basis of above two tables answer the following Questionries:

1. Find out the names of all the clients.
2. Retrieve the list of names and cities of all the clients.
3. List the various products available from the product_master table.
4. List all the clients who are located in Bombay.
5. Display the information for client no 0001 and 0002.
6. Find the products with description as '1.44 drive' and '1.22 Drive'.
7. Find all the products whose sell price is greater then 5000.
8. Find the list of all clients who stay in in city 'Bombay' or city 'Delhi' or 'Madras'.
9. Find the product whose selling price is greater than 2000 and less than or equal to 5000.
10. List the name, city and state of clients not in the state of 'Maharashtra'.

Queries:

```
CREATE TABLE client_master(client_no VARCHAR(6) PRIMARY KEY, name
VARCHAR(20), address_1 VARCHAR(30), address_2 VARCHAR(30), city
VARCHAR(15), pincode NUMERIC(6), state VARCHAR(15), bal_due
NUMERIC(10,2));
CREATE TABLE product_master(product_no VARCHAR(6) PRIMARY KEY,
description VARCHAR(20), profit_percent NUMERIC, unit_measure
VARCHAR(10), qty_on_hand NUMERIC, reorder_lv1 NUMERIC, sell_price
NUMERIC, cost_price NUMERIC);
insert into client_master values('0001', 'Ivan', '', '', 'Bombay',
400054, 'Maharashtra', 1500);
insert into client_master values('0002', 'Vandana', '', '', 'Madras',
780001, 'Tamilnadu', 0);
insert into client_master values('0003', 'Pramada', '', '', 'Bombay',
400057, 'Maharashtra', 5000);
insert into client_master values('0004', 'Basu', '', '', 'Bombay',
400056, 'Maharashtra', 0);
insert into client_master values('0005', 'Ravi', '', '', 'Delhi',
100001, ' ' , 2000);
```

```sql
insert into client_master values('0006', 'Rukmini', '', '', 'Bombay',
400050, 'Maharashtra', 0);
insert into product_master values('P00001', '1.44floppies', 5,
'piece', 100, 20, 525, 500);
insert into product_master values('P03453', 'Monitors', 6, 'piece',
10, 3, 12000, 11200);
insert into product_master values('P06734', 'Mouse', 5, 'piece', 20,
5, 1050, 500);
insert into product_master values('P07865', '1.22 floppies', 5,
'piece', 100, 20, 525, 500);
insert into product_master values('P07868', 'Keyboards', 2, 'piece',
10, 1500, 3150, 3050);
insert into product_master values('P07885', 'CD Drive', 2.5, 'piece',
10, 3, 5250, 5100);
insert into product_master values('P07965', '540 HDD', 4, 'piece',
10, 3, 8400, 8000);
insert into product_master values('P07975', '1.44 Drive', 5, 'piece',
10, 3, 1050, 1000);
insert into product_master VALUES('P08865', '1.22 Drive' , 5, 'piece',
2, 3, 1050, 1000);

select * from client_master;
select * from product_master;
select name from client_master;
select name, city from client_master;
select description from product_master;
select name from client_master where city='Bombay';
select * from client_master where client_no='0001' OR
client_no='0002';
select * from product_master where description='1.44 Drive' OR
description='1.22 Drive';
select * from product_master where sell_price > 5000;
select * from client_master where city='Bombay' or city='Delhi' or
city='Madras';
select * from product_master where sell_price > 2000 and sell_price
<= 5000;
select name, city, state from client_master where not state =
'Maharashtra';
```

Output:

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0001	Ivan			Bombay	400054	Maharashtra	1500.00
2	0002	Vandana			Madras	780001	Tamilnadu	0.00
3	0003	Pramada			Bombay	400057	Maharashtra	5000.00
4	0004	Basu			Bombay	400056	Maharashtra	0.00
5	0005	Ravi			Delhi	100001		2000.00
6	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P00001	1.44floppies	5	piece	100	20	525	500
2	P03453	Monitors	6	piece	10	3	12000	11200
3	P06734	Mouse	5	piece	20	5	1050	500
4	P07865	1.22 floppies	5	piece	100	20	525	500
5	P07868	Keyboards	2	piece	10	1500	3150	3050
6	P07885	CD Drive	3	piece	10	3	5250	5100
7	P07965	540 HDD	4	piece	10	3	8400	8000
8	P07975	1.44 Drive	5	piece	10	3	1050	1000
9	P08865	1.22 Drive	5	piece	2	3	1050	1000

	name
1	Ivan
2	Vandana
3	Pramada
4	Basu
5	Ravi
6	Rukmini

	name	city
1	Ivan	Bombay
2	Vandana	Madras
3	Pramada	Bombay
4	Basu	Bombay
5	Ravi	Delhi
6	Rukmini	Bombay

	description
1	1.44floppies
2	Monitors
3	Mouse
4	1.22 floppi...
5	Keyboards
6	CD Drive
7	540 HDD
8	1.44 Drive
9	1.22 Drive

	name
1	Ivan
2	Pramada
3	Basu
4	Rukmini

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0001	Ivan			Bombay	400054	Maharashtra	1500.00
2	0002	Vandana			Madras	780001	Tamilnadu	0.00

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P07975	1.44 Drive	5	piece	10	3	1050	1000
2	P08865	1.22 Drive	5	piece	2	3	1050	1000

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P03453	Monitors	6	piece	10	3	12000	11200
2	P07885	CD Drive	3	piece	10	3	5250	5100
3	P07965	540 HDD	4	piece	10	3	8400	8000

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0001	Ivan			Bombay	400054	Maharashtra	1500.00
2	0002	Vandana			Madras	780001	Tamilnadu	0.00
3	0003	Prama...			Bombay	400057	Maharashtra	5000.00
4	0004	Basu			Bombay	400056	Maharashtra	0.00
5	0005	Ravi			Delhi	100001		2000.00
6	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P07868	Keyboards	2	piece	10	1500	3150	3050

	name	city	state
1	Vandana	Madras	Tamilnadu
2	Ravi	Delhi	

Lab Assignment 2

Q2) Using the table client master and product master answer the following Questionnaires.

1. Change the selling price of '1.44 floppy drive to Rs.1150.00
2. Delete the record with client 0001 from the client master table.
3. Change the city of client_no'0005' to Bombay.
4. Change the bal_due of client_no '0001, to 1000.
5. Find the products whose selling price is more than 1500 and also find the new selling price as original selling price *15.
6. Find out the clients who stay in a city whose second letter is a.
7. Find out the name of all clients having 'a' as the second letter in their names.
8. List the products in sorted order of their description.
9. Count the total number of orders
10. Calculate the average price of all the products.
11. Calculate the minimum price of products.
12. Determine the maximum and minimum prices . Rename the tittle as 'max_price' and min_price respectively.
13. Count the number of products having price greater than or equal to 1500.

Queries:

```
update product_master set sell_price = 1150 where description =
'1.44floppies';
select * from product_master;
delete from client_master where client_no = '0001';
select * from client_master;
update client_master set city = 'Bombay' where client_no = '0005';
select * from client_master;
insert into client_master values('0001', 'Ivan', '', '', 'Bombay',
400054, 'Maharashtra', 1500);
select * from client_master order by client_no;
update client_master set bal_due = 1000 where client_no = '0001';
select * from client_master order by client_no;
select *, sell_price*15 as original_price from product_master where
sell_price > 1500;
select * from client_master where city LIKE '_a%';
select name from client_master where name LIKE '_a%';
select * from product_master order by description;
select sum(qty_on_hand) from product_master;
select avg(sell_price), avg(cost_price) from product_master;
select min(sell_price), min(cost_price) from product_master;
select min(sell_price) as min_price, max(sell_price) from
product_master;
select count(*) from product_master where sell_price > 1500;
```

Output:

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P00001	1.44floppies	5	piece	100	20	1150	500
2	P03453	Monitors	6	piece	10	3	12000	11200
3	P06734	Mouse	5	piece	20	5	1050	500
4	P07865	1.22 floppies	5	piece	100	20	525	500
5	P07868	Keyboards	2	piece	10	1500	3150	3050
6	P07885	CD Drive	3	piece	10	3	5250	5100
7	P07965	540 HDD	4	piece	10	3	8400	8000
8	P07975	1.44 Drive	5	piece	10	3	1050	1000
9	P08865	1.22 Drive	5	piece	2	3	1050	1000

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0002	Vandana			Madras	780001	Tamilnadu	0.00
2	0003	Pramada			Bombay	400057	Maharashtra	5000.00
3	0004	Basu			Bombay	400056	Maharashtra	0.00
4	0005	Ravi			Delhi	100001		2000.00
5	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0002	Vandana			Madras	780001	Tamilnadu	0.00
2	0003	Pramada			Bombay	400057	Maharashtra	5000.00
3	0004	Basu			Bombay	400056	Maharashtra	0.00
4	0005	Ravi			Bombay	100001		2000.00
5	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0001	Ivan			Bombay	400054	Maharashtra	1500.00
2	0002	Vandana			Madras	780001	Tamilnadu	0.00
3	0003	Pramada			Bombay	400057	Maharashtra	5000.00
4	0004	Basu			Bombay	400056	Maharashtra	0.00
5	0005	Ravi			Bombay	100001		2000.00
6	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0001	Ivan			Bombay	400054	Maharashtra	1000.00
2	0002	Vandana			Madras	780001	Tamilnadu	0.00
3	0003	Prama...			Bombay	400057	Maharashtra	5000.00
4	0004	Basu			Bombay	400056	Maharashtra	0.00
5	0005	Ravi			Bombay	100001		2000.00
6	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price	original_price
1	P03453	Monitors	6	piece	10	3	12000	11200	180000
2	P07868	Keyboards	2	piece	10	1500	3150	3050	47250
3	P07885	CD Drive	3	piece	10	3	5250	5100	78750
4	P07965	540 HDD	4	piece	10	3	8400	8000	126000

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0002	Vandana			Madras	780001	Tamilnadu	0.00

	name
1	Vandana
2	Basu
3	Ravi

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P08865	1.22 Drive	5	piece	2	3	1050	1000
2	P07865	1.22 floppies	5	piece	100	20	525	500
3	P07975	1.44 Drive	5	piece	10	3	1050	1000
4	P00001	1.44floppies	5	piece	100	20	1150	500
5	P07965	540 HDD	4	piece	10	3	8400	8000
6	P07885	CD Drive	3	piece	10	3	5250	5100
7	P07868	Keyboards	2	piece	10	1500	3150	3050
8	P03453	Monitors	6	piece	10	3	12000	11200
9	P06734	Mouse	5	piece	20	5	1050	500

	(No column name)
1	272

	(No column name)	(No column name)
1	3736.111111	3427.777777

	(No column name)	(No column name)
1	525	500

	min_price	(No column name)
1	525	12000

	(No column name)
1	4

Lab Assignment 3

Q1) Create the following tables:

1. Sales_master

Columnname	Datatype	Size	Attributes
Salesman_no	varchar2	6	Primary key/first letter must start with 's'
Sal_name	varchar2	20	Not null
Address	varchar2		Not null
City	varchar2	20	
State	varchar2	20	
Pincode	Number	6	
Sal_amt	Number	8,2	Not null, cannot be 0
Tgt_to_get	Number	6,2	Not null, cannot be 0
Ytd_sales	Number	6,2	Not null, cannot be 0
Remarks	Varchar2	30	

2. Sales_order

Columnname	Datatype	Size	Attributes
S_order_no	varchar2	6	Primary/first letter must be 0
S_order_date	Date	6	Primary key reference clientno of client_master table
Client_no	Varchar2	25	
Dely_add	Varchar2	6	
Salesman_no	Varchar2	6	Foreign key references salesman_no of salesman_master table
Dely_type	Char	1	Delivery part(p)/full(f),default f
Billed_yn	Char	1	
Dely_date	Date		Can not be lessthan s_order_date
Order_status	Varchar2	10	Values ('in process';'fulfilled';back order';'canceled

3. Sales_order_details

Column	Datatype	Size	Attributes
S_order_no	Varchar2	6	Primary key/foreign key references s_order_no of sales_order
Product_no	Varchar2	6	Primary key/foreign key references product_no of product_master
Qty_order	Number	8	
Qty_disp	Number	8	
Product_rate	Number	10,2	

Insert the following data into their respective tables using insert statement:

1. Data for sales_man master table:

Salesman_no	Salesman name	Address	City	Pin code	State	Salamt	Tgt_to_get	Ytd Sales	Rem
500001	Kiran	A/14 worli	Bombay	400002	Mah	3000	100	50	Goo
500002	Manish	65,nariman	Bombay	400001	Mah	3000	200	100	Goo
500003	Ravi	P-7 Bandra	Bombay	400032	Mah	3000	200	100	Goo
500004	Ashish	A/5 Juhu	Bombay	400044	Mah	3500	200	150	Goo

2. Data for salesorder table:

S_orderno	S_orderdate	Client no	Dely type	Bill yn	Salesman no	Delay date	Orderstatus
019001	12-jan-96	0001	F	N	50001	20-jan-96	Ip
019002	25-jan-96	0002	P	N	50002	27-jan-96	C
016865	18-feb-96	0003	F	Y	500003	20-feb-96	F
019003	03-apr-96	0001	F	Y	500001	07-apr-96	F
046866	20-may-96	0004	P	N	500002	22-may-96	C
010008	24-may-96	0005	F	N	500004	26-may-96	Ip

3. Data for sales_order_details table:

S_order no	Product no	Qty ordered	Qty disp	Product_rate
019001	P00001	4	4	525
019001	P07965	2	1	8400
019001	P07885	2	1	5250
019002	P00001	10	0	525
046865	P07868	3	3	3150
046865	P07885	10	10	5250
019003	P00001	4	4	1050
019003	P03453	2	2	1050
046866	P06734	1	1	12000
046866	P07965	1	0	8400
010008	P07975	1	0	1050
010008	P00001	10	5	525

Queries:

```sql
create table Sales_master(Salesman_no VARCHAR(6) NOT NULL PRIMARY KEY
CHECK(Salesman_no LIKE 'S%'), Sal_name VARCHAR(20) NOT NULL,

                          Address VARCHAR(30) NOT NULL,

City VARCHAR(20), Pincode NUMERIC(6), State VARCHAR(20),

Sal_amt NUMERIC(8,2) NOT NULL CHECK(Sal_amt > 0),

Tgt_to_get NUMERIC(6,2) NOT NULL CHECK(Tgt_to_get > 0),

Ytd_sales  NUMERIC(6,2)  NOT  NULL  CHECK(Ytd_sales  >  0),  Remarks
Varchar(30));
Insert     into     Sales_master     values('S00001','Kiran','A/14
worli','Bombay',400002,'Mah',3000,100,50,'Good');
Insert                        into                        Sales_master
values('S00002','Manish','65,nariman','Bombay',400001,'Mah',3000,200
,100,'Good');
Insert     into     Sales_master     values('S00003','Ravi','P-7
Bandra','Bombay',400032,'Mah',3000,200,100,'Good');
Insert     into     Sales_master     values('S00004','Ashish','A/5
Juhu','Bombay',400044,'Mah',3000,200,150,'Good');
SELECT * FROM Sales_master;

create table Sales_order(S_order_no VARCHAR(6) NOT NULL PRIMARY KEY
CHECK(S_order_no like 'O%'), S_order_date DATE,
                    Client_no        VARCHAR(6)        REFERENCES
client_master(client_no), Dely_add VARCHAR(6),
                    salesman_no        VARCHAR(6)        REFERENCES
Sales_master(Salesman_no),
```

```sql
                              Dely_type        CHAR(1)        DEFAULT        'f'
CHECK(Dely_type = 'f' OR Dely_type = 'p'),
                              Billed_yn CHAR(1), Dely_date DATE,
                              Order_status                    VARCHAR(10),
CHECK(Order_status        IN        ('in      process','fulfilled','back
order','canceled')),
                                      CONSTRAINT                Dely_date
CHECK(Dely_date >= S_order_date));
insert        into        Sales_order        values('O19001','1996-01-
12','0001','NULL','S00001','f','n','1996-01-20','in process');
insert        into        Sales_order        values('O19002','1996-01-
25','0002','NULL','S00002','p','n','1996-01-27','canceled');
insert        into        Sales_order        values('O16865','1996-02-
18','0003','NULL','S00003','f','y','1996-02-20','fulfilled');
insert        into        Sales_order        values('O19003','1996-04-
03','0001','NULL','S00001','f','y','1996-04-07','fulfilled');
insert        into        Sales_order        values('O46866','1996-05-
20','0004','NULL','S00002','p','n','1996-05-22','canceled');
insert        into        Sales_order        values('O10008','1996-05-
24','0005','NULL','S00004','f','n','1996-05-26','in process');
select * from Sales_order;

create  table  Sales_order_details(s_order_no  VARCHAR(6)  REFERENCES
Sales_order(S_order_no),
                                Product_no  VARCHAR(6)  REFERENCES
product_master(product_no),
                                Qty_order    NUMERIC(8),    Qty_disp
NUMERIC(8),        Product_rate        NUMERIC(10,2),        PRIMARY
KEY(S_order_no,Product_no));
INSERT into Sales_order_details values('O19001','P00001',4,4,525);
INSERT into Sales_order_details values('O19001','P07965',2,1,8400);
INSERT into Sales_order_details values('O19001','P07885',2,1,5250);
INSERT into Sales_order_details values('O19002','P00001',10,0,525);
INSERT into Sales_order_details values('O16865','P07868',3,3,3150);
INSERT into Sales_order_details values('O16865','P07885',10,10,5250);
INSERT into Sales_order_details values('O19003','P00001',4,4,1050);
INSERT into Sales_order_details values('O19003','P03453',2,2,1050);
INSERT into Sales_order_details values('O46866','P06734',1,1,12000);
INSERT into Sales_order_details values('O46866','P07965',1,0,8400);
INSERT into Sales_order_details values('O10008','P07975',1,0,1050);
INSERT into Sales_order_details values('O10008','P00001',10,5,525);
select * from Sales_order_details;
```

Output:

	Salesman_no	Sal_name	Address	City	Pincode	State	Sal_amt	Tgt_to_get	Ytd_sales	Remarks
1	S00001	Kiran	A/14 worli	Bombay	400002	Mah	3000.00	100.00	50.00	Good
2	S00002	Manish	65,nariman	Bombay	400001	Mah	3000.00	200.00	100.00	Good
3	S00003	Ravi	P-7 Bandra	Bombay	400032	Mah	3000.00	200.00	100.00	Good
4	S00004	Ashish	A/5 Juhu	Bombay	400044	Mah	3000.00	200.00	150.00	Good

	S_order_no	S_order_date	Client_no	Dely_add	salesman_no	Dely_type	Billed_yn	Dely_date	Order_status
1	O10008	1996-05-24	0005	NULL	S00004	f	n	1996-05-26	in process
2	O16865	1996-02-18	0003	NULL	S00003	f	y	1996-02-20	fulfilled
3	O19001	1996-01-12	0001	NULL	S00001	f	n	1996-01-20	in process
4	O19002	1996-01-25	0002	NULL	S00002	p	n	1996-01-27	canceled
5	O19003	1996-04-03	0001	NULL	S00001	f	y	1996-04-07	fulfilled
6	O46866	1996-05-20	0004	NULL	S00002	p	n	1996-05-22	canceled

	s_order_no	Product_no	Qty_order	Qty_disp	Product_rate
1	O10008	P00001	10	5	525.00
2	O10008	P07975	1	0	1050.00
3	O16865	P07868	3	3	3150.00
4	O16865	P07885	10	10	5250.00
5	O19001	P00001	4	4	525.00
6	O19001	P07885	2	1	5250.00
7	O19001	P07965	2	1	8400.00
8	O19002	P00001	10	0	525.00
9	O19003	P00001	4	4	1050.00
10	O19003	P03453	2	2	1050.00
11	O46866	P06734	1	1	12000.00
12	O46866	P07965	1	0	8400.00

Lab Assignment 4

Q1) Create the following tables:

1. **Challan Header**

Column name	data type	size	Attributes
Challan_no	varchar2	6	Primary key
s_order_no	varchar2	6	Foreign key references s_order_no of sales_order table
challan_date	date		not null
billed_yn	char	1	values ('Y','N'). Default 'N'

2. **Table Name: Challan_Details**

Challan No	S_order No	Challan Date	Billed
CH9001	019001	12-DEC-95	Y
CH865	046865	12-NOV-95	Y
CH3965	010008	12-OCT-95	Y

Q2) Insert the following values into the challan header and challan_details tables:

1. **Data for Challan Header table**

Challan No	S_order No	Challan Date	Billed
CH9001	019001	12-DEC-95	Y
CH865	046865	12-NOV-95	Y
CH3965	010008	12-OCT-95	Y

2. **Data for challan_details table**

Challan No	Product No	Qty Disp
CH9001	P00001	4
CH9001	P07965	1
CH9001	P07885	1
CH6865	P07868	3
CH6865	P03453	4
CH6865	P00001	10
CH3965	P00001	5
CH3965	P07975	2

Q3) Answer the following Questionries:

1. Make the primary key to client_no in client_master.
2. Add a new column phone_no in the client_master table.
3. Add the not null constraint in the product_master table with the columns description, profit percent, sell price and cost price.
4. Change the size of client_no field in the client_master table.

5. Select product_no, description where profit percent is between 20 and 30 both inclusive

Queries:

```sql
create table Challan_Header(Challan_no varchar(6) primary key,
s_order_no varchar(6) references Sales_order(S_order_no),
challan_date date not NULL, billed_yn char(1) default 'N'
check(billed_yn = 'Y' or billed_yn = 'N'));
insert into Challan_Header values('CH9001','O19001','1995-12-12','Y');
insert into Challan_Header values('CH6865','O46866','1995-11-12','Y');
insert into Challan_Header values('CH3965','O10008','1995-10-12','Y');
select * from Challan_Header;
create table Challan_Details(Challan_no VARCHAR(6) references
Challan_Header(Challan_no),
Product_no varchar(6), constraint FK_product_cd_prodmaster foreign
key(Product_no) references product_master(product_no),
Qty_disp NUMERIC(4,2) not NULL);
insert into Challan_Details values('CH9001','P00001',4);
insert into Challan_Details values('CH9001','P07965',1);
insert into Challan_Details values('CH9001','P07885',1);
insert into Challan_Details values('CH6865','P07868',3);
insert into Challan_Details values('CH6865','P03453',4);
insert into Challan_Details values('CH6865','P00001',10);
insert into Challan_Details values('CH3965','P00001',5);
insert into Challan_Details values('CH3965','P07975',2);
Select * from Challan_Details;

--Creating The Database
create database DBMS_Lab2;

-- To Changes the database context to the specified database or
database snapshot in SQL Server
use DBMS_Lab2;

--Creating The Tables And Inserting Values In It
--Creating Tables
CREATE TABLE client_master(client_no VARCHAR(6), name VARCHAR(20),
address_1 VARCHAR(30), address_2 VARCHAR(30),
city VARCHAR(15), pincode NUMERIC(6), state VARCHAR(15), bal_due
NUMERIC(10,2));
CREATE TABLE product_master(product_no VARCHAR(6), description
VARCHAR(20), profit_percent NUMERIC, unit_measure VARCHAR(10),
qty_on_hand NUMERIC, reorder_lv1 NUMERIC, sell_price NUMERIC,
cost_price NUMERIC);
--Inserting Values In The Table
insert into client_master values('0001', 'Ivan', 'n/a', 'n/a',
'Bombay', 400054, 'Maharashtra', 1500);
```

```sql
insert into client_master values('0002', 'Vandana', 'n/a', 'n/a', 'Madras', 780001, 'Tamilnadu', 0);
insert into client_master values('0003', 'Pramada', 'n/a', 'n/a', 'Bombay', 400057, 'Maharashtra', 5000);
insert into client_master values('0004', 'Basu', 'n/a', 'n/a', 'Bombay', 400056, 'Maharashtra', 0);
insert into client_master values('0005', 'Ravi', 'n/a', 'n/a', 'Delhi', 100001, 'n/a' , 2000);
insert into client_master values('0006', 'Rukmini', 'n/a', 'n/a', 'Bombay', 400050, 'Maharashtra', 0);
insert into product_master values('P00001', '1.44floppies', 5, 'piece', 100, 20, 525, 500);
insert into product_master values('P03453', 'Monitors', 6, 'piece', 10, 3, 12000, 11200);
insert into product_master values('P06734', 'Mouse', 5, 'piece', 20, 5, 1050, 500);
insert into product_master values('P07865', '1.22 floppies', 5, 'piece', 100, 20, 525, 500);
insert into product_master values('P07868', 'Keyboards', 2, 'piece', 10, 1500, 3150, 3050);
insert into product_master values('P07885', 'CD Drive', 2.5, 'piece', 10, 3, 5250, 5100);
insert into product_master values('P07965', '540 HDD', 4, 'piece', 10, 3, 8400, 8000);
insert into product_master values('P07975', '1.44 Drive', 5, 'piece', 10, 3, 1050, 1000);
insert into product_master VALUES('P08865', '1.22 Drive' , 5, 'piece', 2, 3, 1050, 1000);

--Displaying The Tables
select * from client_master;
select * from product_master;
--Query 1
--Step 1
ALTER TABLE client_master ALTER COLUMN client_no varchar(6) NOT NULL;
ALTER TABLE client_master ALTER COLUMN name VARCHAR(20) NOT NULL;
ALTER TABLE client_master ALTER COLUMN address_1 VARCHAR(30) NOT NULL;
ALTER TABLE client_master ALTER COLUMN address_2 VARCHAR(30) NOT NULL;
ALTER TABLE client_master ALTER COLUMN city VARCHAR(15) NOT NULL;
ALTER TABLE client_master ALTER COLUMN pincode NUMERIC(6) NOT NULL;
ALTER TABLE client_master ALTER COLUMN state VARCHAR(15) NOT NULL;
ALTER TABLE client_master ALTER COLUMN bal_due NUMERIC(10,2) NOT NULL;
--Step 2
alter table client_master add constraint client_master_pk primary key (client_no);
--Query 2
alter table client_master add phone_number numeric(10);
--Query 3
```

```sql
alter table product_master alter column description VARCHAR(20) NOT
NULL;
alter table product_master alter column profit_percent NUMERIC not
null;
alter table product_master alter column sell_price NUMERIC not null;
alter table product_master alter column cost_price NUMERIC not null;
--Query 4
ALTER TABLE dbo.client_master DROP CONSTRAINT client_master_pk;
ALTER TABLE client_master alter column client_no varchar(10) NOT NULL;
alter table client_master add constraint client_master_pk primary key
(client_no);
--Query 5
select    product_no,    description    from    product_master    where
profit_percent >= 20 and profit_percent <= 30;

--Dropping The Tables
IF OBJECT_ID ('dbo.client_master','U') IS NOT NULL
DROP TABLE client_master;
IF OBJECT_ID ('dbo.product_master','U') IS NOT NULL
DROP TABLE product_master;
SET ansi_warnings OFF
GO

--Dropping The Database
Drop DBMS_Lab2;
```

Output:

	Challan_no	s_order_no	challan_date	billed_yn
1	CH3965	O10008	1995-10-12	Y
2	CH6865	O46866	1995-11-12	Y
3	CH9001	O19001	1995-12-12	Y

	Challan_no	Product_no	Qty_disp
1	CH9001	P00001	4.00
2	CH9001	P07965	1.00
3	CH9001	P07885	1.00
4	CH6865	P07868	3.00
5	CH6865	P03453	4.00
6	CH6865	P00001	10.00
7	CH3965	P00001	5.00
8	CH3965	P07975	2.00

	client_no	name	address_1	address_2	city	pincode	state	bal_due	phone_number
1	0001	Ivan	n/a	n/a	Bombay	400054	Maharashtra	1500.00	NULL
2	0002	Vandana	n/a	n/a	Madras	780001	Tamilnadu	0.00	NULL
3	0003	Pramada	n/a	n/a	Bombay	400057	Maharashtra	5000.00	NULL
4	0004	Basu	n/a	n/a	Bombay	400056	Maharashtra	0.00	NULL
5	0005	Ravi	n/a	n/a	Delhi	100001	n/a	2000.00	NULL
6	0006	Rukmini	n/a	n/a	Bombay	400050	Maharashtra	0.00	NULL

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P00001	1.44floppies	5	piece	100	20	525	500
2	P03453	Monitors	6	piece	10	3	12000	11200
3	P06734	Mouse	5	piece	20	5	1050	500
4	P07865	1.22 floppies	5	piece	100	20	525	500
5	P07868	Keyboards	2	piece	10	1500	3150	3050
6	P07885	CD Drive	3	piece	10	3	5250	5100
7	P07965	540 HDD	4	piece	10	3	8400	8000
8	P07975	1.44 Drive	5	piece	10	3	1050	1000
9	P08865	1.22 Drive	5	piece	2	3	1050	1000

product_no	description

Lab Assignment 5

Q1) Answer the following Queries:

1. Find out the product which has been sold to 'Ivan Sayross.'
2. Find out the product and their quantities that will have do delivered.
3. Find the product_no and description of moving products.
4. Find out the names of clients who have purchased 'CD DRIVE'
5. List the product_no and s_order_no of customers haaving qty ordered less than 5 from the order details table for the product "1.44 floppies".
6. Find the products and their quantities for the orders placed by 'Vandan Saitwal 'and "Ivan Bayross".
7. Find the products and their quantities for the orders placed by client_no "C00001" and "C00002".
8. Find the order No, Client No and salesman No. where a client has been received by more than one salesman.
9. Display the s_order_date in the format "dd-mm-yy" e.g. "12- feb-96"
10. Find the date, 15 days after date.

Queries:

```
--Displaying The Tables for the review befre starting the assignment
select * from client_master;
select * from product_master;
select * from Sales_master;
select * from Sales_order;
select * from Sales_order_details;
select * from Challan_Header;
select * from Challan_Details;
--Queries
select  product_master.product_no,  product_master.description  from
Sales_order_details, product_master, client_master, Sales_order where
product_master.product_no  =  Sales_order_details.product_no   and
Sales_order.S_order_no   =   Sales_order_details.S_order_no    and
client_master.client_no     =      Sales_order.client_no       and
client_master.name = 'Ivan';
select product_master.product_no, Sales_order_details.Qty_order from
Sales_order_details,      product_master,      Sales_order      where
Sales_order.Order_status = 'fulfilled' AND product_master.product_no
=  Sales_order_details.product_no   and   Sales_order.S_order_no   =
Sales_order_details.S_order_no;
select  product_master.product_no,  product_master.description  from
product_master,     Sales_order_details,     Sales_order      where
Sales_order.Order_status     =     'in      process'      and
product_master.product_no=Sales_order_details.product_no        and
Sales_order.S_order_no = Sales_order_details.S_order_no;
select  client_master.name  from  client_master,  product_master,
Sales_order_details, Sales_order where product_master.product_no  =
Sales_order_details.product_no   and   Sales_order.S_order_no      =
Sales_order_details.s_order_no    and    client_master.client_no    =
```

```sql
Sales_order.client_no and product_master.description = 'CD Drive'
order by Sales_order.client_no;
select                        Sales_order_details.product_no,
Sales_order_details.S_order_no        from        product_master,
Sales_order_details      where      product_master.product_no      =
Sales_order_details.product_no and Sales_order_details.Qty_order < 5
and product_master.description = '1.44floppies';
select         client_master.name,        product_master.product_no,
product_master.description,      sum(Sales_order_details.Qty_order)
'Quantities'     from    product_master,    client_master,    Sales_order,
Sales_order_details       where        product_master.product_no       =
Sales_order_details.product_no    and    client_master.client_no     =
Sales_order.client_no      and    Sales_order_details.S_order_no      =
Sales_order.S_order_no    and    (client_master.name    =    'Ivan'    or
client_master.name    =    'Vandana')    group    by    client_master.name,
product_master.product_no, product_master.description;
Select        client_master.client_no,        product_master.product_no,
product_master.description,      sum(Sales_order_details.Qty_order)
'Quantities'     from    product_master,    client_master,    Sales_order,
Sales_order_details       where        product_master.product_no       =
Sales_order_details.product_no    and    client_master.client_no     =
Sales_order.client_no      and    Sales_order_details.S_order_no      =
Sales_order.S_order_no   and   (client_master.client_no   =   '0001'   or
client_master.client_no = '0002') group by client_master.client_no,
product_master.product_no, product_master.description;
select        client_master.client_no,        Sales_master.Salesman_no,
Sales_order.S_order_no from client_master, Sales_master, Sales_order
where     client_master.client_no    =    Sales_order.client_no    and
Sales_master.Salesman_no    =    Sales_order.Salesman_no    order    by
client_master.client_no;
select S_order_no, client_no, Salesman_no from Sales_order where 1 <
(select    count(*)    from    Sales_order    where    client_no     =
Sales_order.client_no and 1 = (select count(Sales_order.Salesman_no)
from Sales_order));
select format(Sales_order.S_order_date,'D','en-gb') as S_order_date
from Sales_order;
select    S_order_date    from    Sales_order    where    S_order_date    >=
dateadd(day,15,'1996-01-12');
```

Output:

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0001	Ivan			Bombay	400054	Maharashtra	1000.00
2	0002	Vandana			Madras	780001	Tamilnadu	0.00
3	0003	Pramada			Bombay	400057	Maharashtra	5000.00
4	0004	Basu			Bombay	400056	Maharashtra	0.00
5	0005	Ravi			Bombay	100001		2000.00
6	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P00001	1.44floppies	5	piece	100	20	1150	500
2	P03453	Monitors	6	piece	10	3	12000	11200
3	P06734	Mouse	5	piece	20	5	1050	500
4	P07865	1.22 floppies	5	piece	100	20	525	500
5	P07868	Keyboards	2	piece	10	1500	3150	3050
6	P07885	CD Drive	3	piece	10	3	5250	5100
7	P07965	540 HDD	4	piece	10	3	8400	8000
8	P07975	1.44 Drive	5	piece	10	3	1050	1000
9	P08865	1.22 Drive	5	piece	2	3	1050	1000

	Salesman_no	Sal_name	Address	City	Pincode	State	Sal_amt	Tgt_to_get	Ytd_sales	Remarks
1	S00001	Kiran	A/14 worli	Bombay	400002	Mah	3000.00	100.00	50.00	Good
2	S00002	Manish	65,nariman	Bombay	400001	Mah	3000.00	200.00	100.00	Good
3	S00003	Ravi	P-7 Bandra	Bombay	400032	Mah	3000.00	200.00	100.00	Good
4	S00004	Ashish	A/5 Juhu	Bombay	400044	Mah	3000.00	200.00	150.00	Good

	S_order_no	S_order_date	Client_no	Dely_add	salesman_no	Dely_type	Billed_yn	Dely_date	Order_status
1	O10008	1996-05-24	0005	NULL	S00004	f	n	1996-05-26	in process
2	O16865	1996-02-18	0003	NULL	S00003	f	y	1996-02-20	fulfilled
3	O19001	1996-01-12	0001	NULL	S00001	f	n	1996-01-20	in process
4	O19002	1996-01-25	0002	NULL	S00002	p	n	1996-01-27	canceled
5	O19003	1996-04-03	0001	NULL	S00001	f	y	1996-04-07	fulfilled
6	O46866	1996-05-20	0004	NULL	S00002	p	n	1996-05-22	canceled

	s_order_no	Product_no	Qty_order	Qty_disp	Product_rate
1	O10008	P00001	10	5	525.00
2	O10008	P07975	1	0	1050.00
3	O16865	P07868	3	3	3150.00
4	O16865	P07885	10	10	5250.00
5	O19001	P00001	4	4	525.00
6	O19001	P07885	2	1	5250.00
7	O19001	P07965	2	1	8400.00
8	O19002	P00001	10	0	525.00
9	O19003	P00001	4	4	1050.00
10	O19003	P03453	2	2	1050.00
11	O46866	P06734	1	1	12000.00
12	O46866	P07965	1	0	8400.00

	Challan_no	s_order_no	challan_date	billed_yn
1	CH3965	O10008	1995-10-12	Y
2	CH6865	O46866	1995-11-12	Y
3	CH9001	O19001	1995-12-12	Y

	Challan_no	Product_no	Qty_disp
1	CH9001	P00001	4.00
2	CH9001	P07965	1.00
3	CH9001	P07885	1.00
4	CH6865	P07868	3.00
5	CH6865	P03453	4.00
6	CH6865	P00001	10.00
7	CH3965	P00001	5.00
8	CH3965	P07975	2.00

	product_no	description
1	P00001	1.44floppies
2	P07885	CD Drive
3	P07965	540 HDD
4	P00001	1.44floppies
5	P03453	Monitors

	product_no	Qty_order
1	P07868	3
2	P07885	10
3	P00001	4
4	P03453	2

	product_no	description
1	P00001	1.44floppies
2	P07975	1.44 Drive
3	P00001	1.44floppies
4	P07885	CD Drive
5	P07965	540 HDD

	name
1	Ivan
2	Pramada

	product_no	S_order_no
1	P00001	O19001
2	P00001	O19003

	name	product_no	description	Quantities
1	Ivan	P00001	1.44floppies	8
2	Vandana	P00001	1.44floppies	10
3	Ivan	P03453	Monitors	2
4	Ivan	P07885	CD Drive	2
5	Ivan	P07965	540 HDD	2

	client_no	product_no	description	Quantities
1	0001	P00001	1.44floppies	8
2	0002	P00001	1.44floppies	10
3	0001	P03453	Monitors	2
4	0001	P07885	CD Drive	2
5	0001	P07965	540 HDD	2

	client_no	Salesman_no	S_order_no
1	0001	S00001	O19001
2	0001	S00001	O19003
3	0002	S00002	O19002
4	0003	S00003	O16865
5	0004	S00002	O46866
6	0005	S00004	O10008

S_order_no	client_no	Salesman_no

	S_order_date
1	24 May 1996
2	18 February 1996
3	12 January 1996
4	25 January 1996
5	03 April 1996
6	20 May 1996

	S_order_date
1	1996-05-24
2	1996-02-18
3	1996-04-03
4	1996-05-20

Lab Assignment 6

Q1) Answer the following Queries:

1. **Print the description and total quantity sold for each product.**
2. **Find the value of each product sold.**
3. **Calculate the average quantity sold for each client that has a maximum order value of 15000.**
4. **Find out the products which has been sold to Ivan.**
5. **Find the names of clients who have 'CD Drive'.**
6. **Find the products and their quantities for the orders placed by 'Vandana' and 'Ivan'.**
7. **Select product_no, total qty_ordered for each product.**
8. **Select product_no, product description and qty ordered for each product.**
9. **Display the order number and day on which clients placed their order.**
10. **Display the month and Date when the order must be delivered.**

Queries:

```
--Displaying the tables
select * from client_master;
select * from product_master;
select * from Sales_master;
select * from Sales_order;
select * from Sales_order_details;
select * from Challan_Header;
select * from Challan_Details;
--Executing the queries
select    description,    Sum(Qty_disp)    from    product_master,
Sales_order_details    where    product_master.product_no    =
Sales_order_details.product_no group by description;
select   Sales_order_details.Product_no,   product_master.description
from       Sales_order_details,       product_master       where
product_master.product_no =Sales_order_details.Product_no group by
Sales_order_details.Product_no,       product_master.description,
Sales_order_details.Product_rate;
select client_master.client_no, avg(Sales_order_details.Qty_disp) as
AvgSales from Sales_order_details, client_master, Sales_order where
client_master.client_no    =    Sales_order.Client_no    and
Sales_order.S_order_no = Sales_order_details.s_order_no group by
client_master.client_no   having   max(Sales_order_details.Qty_order
*Sales_order_details.Product_rate) > 1500;
select   client_master.client_no,   product_master.description   from
client_master, product_master, Sales_order_details, Sales_order where
client_master.client_no    =    Sales_order.Client_no    and
Sales_order.s_order_no   =   Sales_order_details.s_order_no   and
client_master.name  =  'Ivan'  group  by  client_master.client_no,
product_master.description;
select    client_master.name,    product_master.description    from
Sales_order_details, client_master, Sales_order, product_master where
```

```sql
client_master.client_no       =       sales_order.client_no       and
Sales_order.S_order_no    =    Sales_order_details.s_order_no    and
product_master.description = 'CD Drive' group by client_master.name,
product_master.description;
select        client_master.name,        product_master.description,
sum(Sales_order_details.Qty_order) as Units from Sales_order_details,
Sales_order,        client_master,        product_master        where
client_master.client_no       =       Sales_order.Client_no       and
Sales_order.S_order_no    =    Sales_order_details.s_order_no    and
product_master.product_no = Sales_order_details.Product_no group by
client_master.name,        product_master.description        having
(client_master.name = 'Ivan' or client_master.name = 'Vandana');
select product_master.product_no, sum(Sales_order_details.Qty_order)
as Units from Sales_order_details, client_master, product_master,
Sales_order        where        Sales_order.S_order_no        =
Sales_order_details.s_order_no    and    product_master.product_no    =
Sales_order_details.Product_no group by product_master.product_no;
select        product_master.product_no,        product_master.description,
sum(Sales_order_details.Qty_order) as Units from Sales_order_details,
client_master,        Sales_order,        product_master        where
Sales_order.S_order_no    =    Sales_order_details.s_order_no    and
product_master.product_no = Sales_order_details.Product_no group by
product_master.product_no, product_master.description;
select S_order_no, datename(weekday,S_order_date) from Sales_order;
select  S_order_no,  datepart(d,S_order_date)  as  'Delivery  Date',
datename(month,S_order_date) as 'Delivery Month' from Sales_order;
```

Output:

	client_no	name	address_1	address_2	city	pincode	state	bal_due
1	0001	Ivan			Bombay	400054	Maharashtra	1000.00
2	0002	Vandana			Madras	780001	Tamilnadu	0.00
3	0003	Pramada			Bombay	400057	Maharashtra	5000.00
4	0004	Basu			Bombay	400056	Maharashtra	0.00
5	0005	Ravi			Bombay	100001		2000.00
6	0006	Rukmini			Bombay	400050	Maharashtra	0.00

	product_no	description	profit_percent	unit_measure	qty_on_hand	reorder_lv1	sell_price	cost_price
1	P00001	1.44floppies	5	piece	100	20	1150	500
2	P03453	Monitors	6	piece	10	3	12000	11200
3	P06734	Mouse	5	piece	20	5	1050	500
4	P07865	1.22 floppies	5	piece	100	20	525	500
5	P07868	Keyboards	2	piece	10	1500	3150	3050
6	P07885	CD Drive	3	piece	10	3	5250	5100
7	P07965	540 HDD	4	piece	10	3	8400	8000
8	P07975	1.44 Drive	5	piece	10	3	1050	1000
9	P08865	1.22 Drive	5	piece	2	3	1050	1000

	Salesman_no	Sal_name	Address	City	Pincode	State	Sal_amt	Tgt_to_get	Ytd_sales	Remarks
1	S00001	Kiran	A/14 worli	Bombay	400002	Mah	3000.00	100.00	50.00	Good
2	S00002	Manish	65,nariman	Bombay	400001	Mah	3000.00	200.00	100.00	Good
3	S00003	Ravi	P-7 Bandra	Bombay	400032	Mah	3000.00	200.00	100.00	Good
4	S00004	Ashish	A/5 Juhu	Bombay	400044	Mah	3000.00	200.00	150.00	Good

	S_order_no	S_order_date	Client_no	Dely_add	salesman_no	Dely_type	Billed_yn	Dely_date	Order_status
1	O10008	1996-05-24	0005	NULL	S00004	f	n	1996-05-26	in process
2	O16865	1996-02-18	0003	NULL	S00003	f	y	1996-02-20	fulfilled
3	O19001	1996-01-12	0001	NULL	S00001	f	n	1996-01-20	in process
4	O19002	1996-01-25	0002	NULL	S00002	p	n	1996-01-27	canceled
5	O19003	1996-04-03	0001	NULL	S00001	f	y	1996-04-07	fulfilled
6	O46866	1996-05-20	0004	NULL	S00002	p	n	1996-05-22	canceled

	s_order_no	Product_no	Qty_order	Qty_disp	Product_rate
1	O10008	P00001	10	5	525.00
2	O10008	P07975	1	0	1050.00
3	O16865	P07868	3	3	3150.00
4	O16865	P07885	10	10	5250.00
5	O19001	P00001	4	4	525.00
6	O19001	P07885	2	1	5250.00
7	O19001	P07965	2	1	8400.00
8	O19002	P00001	10	0	525.00
9	O19003	P00001	4	4	1050.00
10	O19003	P03453	2	2	1050.00
11	O46866	P06734	1	1	12000.00
12	O46866	P07965	1	0	8400.00

	Challan_no	s_order_no	challan_date	billed_yn
1	CH3965	O10008	1995-10-12	Y
2	CH6865	O46866	1995-11-12	Y
3	CH9001	O19001	1995-12-12	Y

	Challan_no	Product_no	Qty_disp
1	CH9001	P00001	4.00
2	CH9001	P07965	1.00
3	CH9001	P07885	1.00
4	CH6865	P07868	3.00
5	CH6865	P03453	4.00
6	CH6865	P00001	10.00
7	CH3965	P00001	5.00
8	CH3965	P07975	2.00

	description	(No column name)
1	1.44 Drive	0
2	1.44floppies	13
3	540 HDD	1
4	CD Drive	11
5	Keyboards	3
6	Monitors	2
7	Mouse	1

	Product_no	description
1	P00001	1.44floppies
2	P00001	1.44floppies
3	P03453	Monitors
4	P06734	Mouse
5	P07868	Keyboards
6	P07885	CD Drive
7	P07965	540 HDD
8	P07975	1.44 Drive

	client_no	AvgSales
1	0001	2.400000
2	0002	0.000000
3	0003	6.500000
4	0004	0.500000
5	0005	2.500000

	client_no	description
1	0001	1.22 Drive
2	0001	1.22 floppies
3	0001	1.44 Drive
4	0001	1.44floppies
5	0001	540 HDD
6	0001	CD Drive
7	0001	Keyboards
8	0001	Monitors
9	0001	Mouse

	name	description
1	Basu	CD Drive
2	Ivan	CD Drive
3	Pra...	CD Drive
4	Ravi	CD Drive
5	Van...	CD Drive

	name	description	Units
1	Ivan	1.44floppies	8
2	Vandana	1.44floppies	10
3	Ivan	540 HDD	2
4	Ivan	CD Drive	2
5	Ivan	Monitors	2

	product_no	Units
1	P00001	168
2	P03453	12
3	P06734	6
4	P07868	18
5	P07885	72
6	P07965	18
7	P07975	6

	product_no	description	Units
1	P00001	1.44floppies	168
2	P03453	Monitors	12
3	P06734	Mouse	6
4	P07868	Keyboards	18
5	P07885	CD Drive	72
6	P07965	540 HDD	18
7	P07975	1.44 Drive	6

	S_order_no	(No column name)
1	O10008	Friday
2	O16865	Sunday
3	O19001	Friday
4	O19002	Thursday
5	O19003	Wednesday
6	O46866	Monday

	S_order_no	Delivery Date	Delivery Month
1	O10008	24	May
2	O16865	18	February
3	O19001	12	January
4	O19002	25	January
5	O19003	3	April
6	O46866	20	May

Lab Assignment 7

Q1) Answer the following Queries:

1. **Find the product_no and description of non- moving products.**
2. **Find the customer name, address, city and pincode for the client who has placed order no "019001".**
3. **Find the client names who have placed order before the month of may 96.**
4. **Find out if product "1.44 Drive" is ordered by only client and print the client_no name to whom it was sold.**
5. **Find the names of client who have placed orders worth Rs.10000 or more.**
6. **Select the orders placed by 'Rahul Desai".**
7. **Select all the clients and the salesman in the city of Bombay.**
8. **Select salesman name in "Bombay" who has atleast one client located at "Bombay".**
9. **Select the product_no, description, qty_on-hand,cost_price of non_moving items in the product_master table.**

Queries:

```sql
--Query 1
select product_no, description from product_master where product_no
in (select product_no from Sales_order_details where S_order_no in(
select S_order_no from Sales_order where Order_status = 'canceled'));
--Query 2
select client_no, name, address_1, address_2, city, pincode from
client_master where client_no in (select Client_no from Sales_order
where S_order_no = 'O19001');
--Query 3
select name from client_master where client_no in (select Client_no
from Sales_order where S_order_date < '1996-05-01');
--Query 4
select client_no, name from client_master where client_no in (select
client_no from Sales_order where S_order_no in (select S_order_no from
Sales_order_details where Product_no in (select Product_no from
product_master where description = '1.44 Drive')));
--Query 5
select name from client_master where client_no in (select client_no
from Sales_order where S_order_no in (select S_order_no from
Sales_order_details where Product_rate >= 10000.00));
--Query 6
select S_order_no, Client_no from Sales_order where Client_no in
(select client_no from client_master where name = 'Rahul Desai');
--Query 7 - Need some time to think till then I will execute Query 8,
9, and 10
--Query 8
select distinct name from client_master where city = 'Bombay' union
select distinct Sal_name from Sales_master where City = 'Bombay';
--Query 9
```

```sql
select  Sal_name  from  Sales_master  where  Salesman_no  in  (select
salesman_no from Sales_order where Client_no in (select client_no from
client_master where city = 'Bombay'));
--Query 10
select  product_no,  description,  qty_on_hand,  cost_price  from
product_master  where  product_no  not  in  (select  product_no  from
Sales_order_details);
```

Output:

	product_no	description
1	P00001	1.44floppies
2	P06734	Mouse
3	P07965	540 HDD

	client_no	name	address_1	address_2	city	pincode
1	0001	Ivan			Bombay	400054

	name
1	Ivan
2	Vandana
3	Pramada

	client_no	name
1	0005	Ravi

	name
1	Basu

S_order_no	Client_no

	name
1	Ashish
2	Basu
3	Ivan
4	Kiran
5	Manish
6	Pramada
7	Ravi
8	Rukmini

	Sal_name
1	Kiran
2	Manish
3	Ravi
4	Ashish

	product_no	description	qty_on_hand	cost_price
1	P07865	1.22 floppies	100	500
2	P08865	1.22 Drive	2	1000

	product_no	description	qty_on_hand	cost_price
1	P07865	1.22 floppies	100	500
2	P08865	1.22 Drive	2	1000

Lab Assignment 8

Q1) Answer the following Questions:

1. Create an index on the table client_master, field client_no.
2. Create an index on the sales_order, field s_order_no.
3. Create an composite index on the sales_order_details table for the columns s_order_no and product_no.
4. Create an composite index ch_index on challan_header table for the columns challan no and s_order_no.
5. Create an uniQuestion index on the table salesman_master, field salesman_no.
6. Drop index ch_index on table challan_header.
7. Create view on salesman_master whose sal_amt is less than 3500.
8. Create a view client_view on client_master and rename the columns as name, add1, add2, city, pcode, state respectively.
9. Select the client names from client_view who lives in city 'Bombay'.
10. Drop the view client_view.

Queries:

Creating the Database

```
CREATE DATABASE DBMS_Lab3;
use DBMS_Lab3;
```

Creating the Tables

```
CREATE TABLE client_master(client_no VARCHAR(6) PRIMARY KEY, name
VARCHAR(20), address_1 VARCHAR(30), address_2 VARCHAR(30), city
VARCHAR(15), pincode NUMERIC(6), state VARCHAR(15), bal_due
NUMERIC(10,2));
CREATE TABLE product_master(product_no VARCHAR(6) PRIMARY KEY,
description VARCHAR(20), profit_percent NUMERIC, unit_measure
VARCHAR(10), qty_on_hand NUMERIC, reorder_lv1 NUMERIC, sell_price
NUMERIC, cost_price NUMERIC);
insert into client_master values('0001', 'Ivan', '', '', 'Bombay',
400054, 'Maharashtra', 1500);
insert into client_master values('0002', 'Vandana', '', '', 'Madras',
780001, 'Tamilnadu', 0);
insert into client_master values('0003', 'Pramada', '', '', 'Bombay',
400057, 'Maharashtra', 5000);
insert into client_master values('0004', 'Basu', '', '', 'Bombay',
400056, 'Maharashtra', 0);
insert into client_master values('0005', 'Ravi', '', '', 'Delhi',
100001, ' ' , 2000);
insert into client_master values('0006', 'Rukmini', '', '', 'Bombay',
400050, 'Maharashtra', 0);
insert into product_master values('P00001', '1.44floppies', 5,
'piece', 100, 20, 525, 500);
insert into product_master values('P03453', 'Monitors', 6, 'piece',
10, 3, 12000, 11200);
```

```sql
insert into product_master values('P06734', 'Mouse', 5, 'piece', 20,
5, 1050, 500);
insert into product_master values('P07865', '1.22 floppies', 5,
'piece', 100, 20, 525, 500);
insert into product_master values('P07868', 'Keyboards', 2, 'piece',
10, 1500, 3150, 3050);
insert into product_master values('P07885', 'CD Drive', 2.5, 'piece',
10, 3, 5250, 5100);
insert into product_master values('P07965', '540 HDD', 4, 'piece',
10, 3, 8400, 8000);
insert into product_master values('P07975', '1.44 Drive', 5, 'piece',
10, 3, 1050, 1000);
insert into product_master VALUES('P08865', '1.22 Drive' , 5, 'piece',
2, 3, 1050, 1000);
create table Sales_master(Salesman_no VARCHAR(6) NOT NULL PRIMARY KEY
CHECK(Salesman_no LIKE 'S%'), Sal_name VARCHAR(20) NOT NULL,
Address VARCHAR(30) NOT NULL,
City VARCHAR(20), Pincode NUMERIC(6), State VARCHAR(20),
Sal_amt NUMERIC(8,2) NOT NULL CHECK(Sal_amt > 0),
Tgt_to_get NUMERIC(6,2) NOT NULL CHECK(Tgt_to_get > 0),
Ytd_sales NUMERIC(6,2) NOT NULL CHECK(Ytd_sales > 0), Remarks
Varchar(30));
Insert into Sales_master values('S00001','Kiran','A/14
worli','Bombay',400002,'Mah',3000,100,50,'Good');
Insert into Sales_master
values('S00002','Manish','65,nariman','Bombay',400001,'Mah',3000,200
,100,'Good');
Insert into Sales_master values('S00003','Ravi','P-7
Bandra','Bombay',400032,'Mah',3000,200,100,'Good');
Insert into Sales_master values('S00004','Ashish','A/5
Juhu','Bombay',400044,'Mah',3000,200,150,'Good');
create table Sales_order(S_order_no VARCHAR(6) NOT NULL PRIMARY KEY
CHECK(S_order_no like 'O%'), S_order_date DATE,
Client_no VARCHAR(6) REFERENCES client_master(client_no), Dely_add
VARCHAR(6),
salesman_no VARCHAR(6) REFERENCES Sales_master(Salesman_no),
Dely_type CHAR(1) DEFAULT 'f' CHECK(Dely_type = 'f' OR Dely_type =
'p'),
Billed_yn CHAR(1), Dely_date DATE,
Order_status VARCHAR(10), CHECK(Order_status IN ('in
process','fulfilled','back order','canceled')),
CONSTRAINT Dely_date CHECK(Dely_date >= S_order_date));
insert into Sales_order values('O19001','1996-01-
12','0001','NULL','S00001','f','n','1996-01-20','in process');
insert into Sales_order values('O19002','1996-01-
25','0002','NULL','S00002','p','n','1996-01-27','canceled');
insert into Sales_order values('O16865','1996-02-
18','0003','NULL','S00003','f','y','1996-02-20','fulfilled');
```

```sql
insert        into        Sales_order        values('O19003','1996-04-
03','0001','NULL','S00001','f','y','1996-04-07','fulfilled');
insert        into        Sales_order        values('O46866','1996-05-
20','0004','NULL','S00002','p','n','1996-05-22','canceled');
insert        into        Sales_order        values('O10008','1996-05-
24','0005','NULL','S00004','f','n','1996-05-26','in process');
create   table   Sales_order_details(s_order_no  VARCHAR(6)  REFERENCES
Sales_order(S_order_no),
Product_no VARCHAR(6) REFERENCES product_master(product_no),
Qty_order      NUMERIC(8),      Qty_disp      NUMERIC(8),      Product_rate
NUMERIC(10,2), PRIMARY KEY(S_order_no,Product_no));
INSERT into Sales_order_details values('O19001','P00001',4,4,525);
INSERT into Sales_order_details values('O19001','P07965',2,1,8400);
INSERT into Sales_order_details values('O19001','P07885',2,1,5250);
INSERT into Sales_order_details values('O19002','P00001',10,0,525);
INSERT into Sales_order_details values('O16865','P07868',3,3,3150);
INSERT into Sales_order_details values('O16865','P07885',10,10,5250);
INSERT into Sales_order_details values('O19003','P00001',4,4,1050);
INSERT into Sales_order_details values('O19003','P03453',2,2,1050);
INSERT into Sales_order_details values('O46866','P06734',1,1,12000);
INSERT into Sales_order_details values('O46866','P07965',1,0,8400);
INSERT into Sales_order_details values('O10008','P07975',1,0,1050);
INSERT into Sales_order_details values('O10008','P00001',10,5,525);
create   table   Challan_Header(Challan_no   varchar(6)   primary   key,
s_order_no varchar(6) references Sales_order(S_order_no),
challan_date   date   not   NULL,   billed_yn   char(1)   default   'N'
check(billed_yn = 'Y' or billed_yn = 'N'));
insert    into    Challan_Header    values('CH9001','O19001','1995-12-
12','Y');
insert    into    Challan_Header    values('CH6865','O46866','1995-11-
12','Y');
insert    into    Challan_Header    values('CH3965','O10008','1995-10-
12','Y');
create   table   Challan_Details(Challan_no   VARCHAR(6)   references
Challan_Header(Challan_no),
Product_no   varchar(6),   constraint   FK_product_cd_prodmaster   foreign
key(Product_no) references product_master(product_no),
Qty_disp NUMERIC(4,2) not NULL);
insert into Challan_Details values('CH9001','P00001',4);
insert into Challan_Details values('CH9001','P07965',1);
insert into Challan_Details values('CH9001','P07885',1);
insert into Challan_Details values('CH6865','P07868',3);
insert into Challan_Details values('CH6865','P03453',4);
insert into Challan_Details values('CH6865','P00001',10);
insert into Challan_Details values('CH3965','P00001',5);
insert into Challan_Details values('CH3965','P07975',2);
--Displaying Tables
select * from client_master;
select * from product_master;
```

```sql
select * from Sales_master;
select * from Sales_order;
select * from Sales_order_details;
select * from Challan_Header;
select * from Challan_Details;
```

Executing Queries

```sql
--Queries
--Query 1 to 6
--Query 1
drop index if exists [index_cm] on [client_master];
go
create index index_cm on client_master(client_no);
EXEC sp_helpindex 'dbo.client_master';
GO
--Query 2
drop index if exists [index_so] on [Sales_order];
go
create index index_so on Sales_order(S_order_no);
EXEC sp_helpindex 'dbo.Sales_order';
GO
--Query 3
drop index if exists [index_sod] on [Sales_order_details];
go
create index index_sod on Sales_order_details(S_order_no,Product_no);
EXEC sp_helpindex 'dbo.Sales_order_details';
GO
--Query 4
drop index if exists [ch_index] on [Challan_Header];
go
create index ch_index on Challan_Header(Challan_no,s_order_no);
EXEC sp_helpindex 'dbo.Challan_Header';
GO
--Query 5
drop index if exists [index_sm] on [Sales_master];
go
create unique index index_sm on Sales_master(Salesman_no);
EXEC sp_helpindex 'dbo.Sales_master';
GO
--Query 6
drop index if exists [ch_index] on [Challan_Header];
go
EXEC sp_helpindex 'dbo.Challan_Header';
GO
--Query 7
--Query 7 Part 1
--DROP view if exists dbo.sales_master_view;
--go
create view Sales_master_view as select * from Sales_master where
Sal_amt < 3500;
```

```
--Query 7 Part 2
select * from Sales_master_view;

--Query 8 to 10
--Query 8 Part 1
create view client_view as select name as 'Name', address_1 as
'Address_1', address_2 as'Address_2', city as 'City', pincode as
'Pincode', state as 'State' from client_master;
--Query 8 Part 2
select * from client_view;
--Query 9
select Name from client_view where City = 'Bombay';
--Query 10
DROP view if exists dbo.client_view;
Go
```

Output

Queries 1 to 6

	index_name	index_description	index_keys
1	index_cm	nonclustered located on PRIMARY	client_no
2	PK__client_m__BF218C7F95DA8C69	clustered, unique, primary key located on PRIMARY	client_no

	index_name	index_description	index_keys
1	index_so	nonclustered located on PRIMARY	S_order_no
2	PK__Sales_or__774A7E6C84BB7F6F	clustered, unique, primary key located on PRIMARY	S_order_no
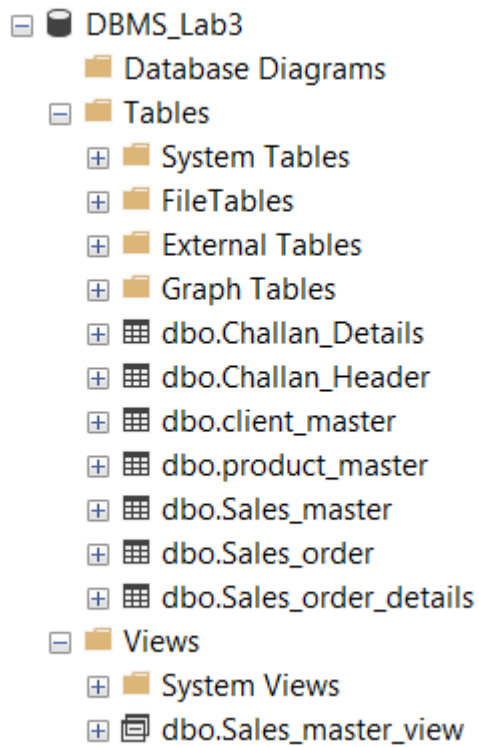
	index_name	index_description	index_keys
1	index_sod	nonclustered located on PRIMARY	s_order_no, Product_no
2	PK__Sales_or__317DCD3C2F88BF48	clustered, unique, primary key located on PRIMARY	s_order_no, Product_no

	index_name	index_description	index_keys
1	ch_index	nonclustered located on PRIMARY	Challan_no, s_order_no
2	PK__Challan___5F7213996E5F537C	clustered, unique, primary key located on PRIMARY	Challan_no

	index_name	index_description	index_keys
1	index_sm	nonclustered, unique located on PRIMARY	Salesman_no
2	PK__Sales_ma__7B0CE494925D55C8	clustered, unique, primary key located on PRIMARY	Salesman_no

	index_name	index_description	index_keys
1	PK__Challan___5F7213996E5F537C	clustered, unique, primary key located on PRIMARY	Challan_no

Query 7

- DBMS_Lab3
  - Database Diagrams
  - Tables
    - System Tables
    - FileTables
    - External Tables
    - Graph Tables
    - dbo.Challan_Details
    - dbo.Challan_Header
    - dbo.client_master
    - dbo.product_master
    - dbo.Sales_master
    - dbo.Sales_order
    - dbo.Sales_order_details
  - Views
    - System Views
    - dbo.Sales_master_view

	Salesman_no	Sal_name	Address	City	Pincode	State	Sal_amt	Tgt_to_get	Ytd_sales	Remarks
1	S00001	Kiran	A/14 worli	Bombay	400002	Mah	3000.00	100.00	50.00	Good
2	S00002	Manish	65,nariman	Bombay	400001	Mah	3000.00	200.00	100.00	Good
3	S00003	Ravi	P-7 Bandra	Bombay	400032	Mah	3000.00	200.00	100.00	Good
4	S00004	Ashish	A/5 Juhu	Bombay	400044	Mah	3000.00	200.00	150.00	Good

Query 8 to 10

- DBMS_Lab3
  - Database Diagrams
  - Tables
    - System Tables
    - FileTables
    - External Tables
    - Graph Tables
    - dbo.Challan_Details
    - dbo.Challan_Header
    - dbo.client_master
    - dbo.product_master
    - dbo.Sales_master
    - dbo.Sales_order
    - dbo.Sales_order_details
  - Views
    - System Views
    - dbo.client_view
    - dbo.Sales_master_view

	Name	Address_1	Address_2	City	Pincode	State
1	Ivan			Bombay	400054	Maharashtra
2	Vandana			Madras	780001	Tamilnadu
3	Pramada			Bombay	400057	Maharashtra
4	Basu			Bombay	400056	Maharashtra
5	Ravi			Delhi	100001	
6	Rukmini			Bombay	400050	Maharashtra

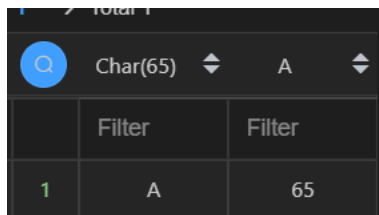|	Name
1	Ivan
2	Pramada
3	Basu
4	Rukmini

- DBMS_Lab3
  - Database Diagrams
  - Tables
    - System Tables
    - FileTables
    - External Tables
    - Graph Tables
    - dbo.Challan_Details
    - dbo.Challan_Header
    - dbo.client_master
    - dbo.product_master
    - dbo.Sales_master
    - dbo.Sales_order
    - dbo.Sales_order_details
  - Views
    - System Views
    - dbo.Sales_master_view

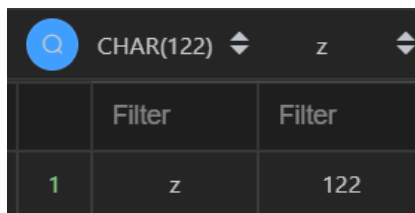# Lab Assignment 9

Q) Perform the following Functions:

1) SELECT Char(65), ASCII('A') AS A;

	Char(65) ⬍	A ⬍
	Filter	Filter
1	A	65
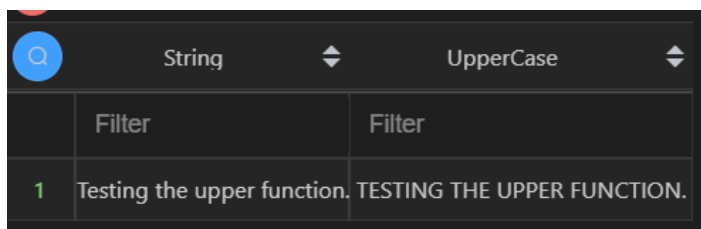
2) SELECT CHAR(122),ASCII('z') AS z;

	CHAR(122) ⬍	z ⬍
	Filter	Filter
1	z	122
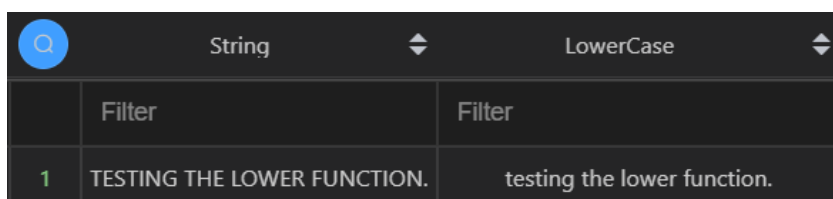
3) SELECT "Testing the upper function." AS String,UPPER("Testing the upper function.") AS UpperCase;

	String ⬍	UpperCase ⬍
	Filter	Filter
1	Testing the upper function.	TESTING THE UPPER FUNCTION.

4) SELECT "TESTING THE LOWER FUNCTION." AS String, LOWER("TESTING THE LOWER FUNCTION.") AS LowerCase;

	String ⬍	LowerCase ⬍
	Filter	Filter
1	TESTING THE LOWER FUNCTION.	testing the lower function.

5) SELECT "Testing the upper function." AS Initial,

  REPLACE(

    "Testing the upper function.",

    "upper",

    "replace"

) AS Final;

	Initial	Final
	Filter	Filter
1	Testing the upper function.	Testing the replace function.

6) SELECT CURRENT_DATE() AS DATE;

1 > Total 1

	DATE
	Filter
1	2021-10-21

7) SELECT EXTRACT(DAY FROM "2021-09-29");

1 > Total 1

	EXTRACT(DAY FROM ":
	Filter
1	29

8) SELECT SYSDATE();

1 > Total 1

	SYSDATE()
	Filter
1	2021-10-21 01:17:50

9) SELECT -32,ABS(-32) AS Absolute_Value;

-32	Absolute_Value
Filter	Filter
-32	32

10) SELECT 2.43,FLOOR(2.43) AS Floor_Value;

2.43	Floor_Value
Filter	Filter
2.43	2

11) SELECT "1,2,3",GREATEST(1, 2, 3) AS Greatest_Value;

	1,2,3	Greatest_Value
	Filter	Filter
1	1,2,3	3

12) SELECT "1,2,3",LEAST(1, 2, 3) AS Least_Value;

1 > Total 1

	1,2,3	Least_Value
	Filter	Filter
1	1,2,3	1

13) SELECT "This is a string",LENGTH("This is a string") AS Length;

	This is a string	Length
	Filter	Filter
1	This is a string	16

14) SELECT "25" AS Number,SQRT(25) AS Square_Root;

	Number	Square_Root
	Filter	Filter
1	25	5

15) SELECT "5^4" AS Equation,POWER(5, 4) AS Result;

	Equation	Result
	Filter	Filter
1	5^4	625

16) SELECT "4.56" AS Number,ROUND(4.56) AS RoundOff;

	Number	RoundOff
	Filter	Filter
1	4.56	5

17) SELECT "4.46" AS Number,ROUND(4.46) AS RoundOff;

	Number	RoundOff
	Filter	Filter
1	4.46	4

18) SELECT SIN(Pi() / 2);

	SIN(Pi() / 2)
	Filter
1	1

19) SELECT COS(Pi());

	COS(Pi())
	Filter
1	-1

20) SELECT TAN(Pi() / 4);

	TAN(Pi() / 4)
	Filter
1	0.9999999999999999

21) SELECT "-40" AS Number, SIGN(-40) AS Sign;

	Number	Sign
	Filter	Filter
1	-40	-1

22) SELECT "40" AS Number, SIGN(40) AS Sign;

	Number	Sign
	Filter	Filter
1	40	1

23) SELECT LN(10);

	LN(10)
	Filter
1	2.302585092994046

24) SELECT LOG(10, 10);

	LOG(10, 10)
	Filter
1	1

25) SELECT MOD(19, 7);

	MOD(19, 7)
	Filter
1	5

26) SELECT EXP(2);

	EXP(2)
	Filter
1	7.38905609893065

27) SELECT "tttesting trimttt" AS String,

TRIM('t' FROM "tttesting trimttt") AS Trimmed

	String	Trimmed
	Filter	Filter
1	tttesting trimttt	esting trim

Lab Assignment 10

Creating Tables:

CREATE Table Students(

  StudNr NUMERIC(5) PRIMARY KEY,

  Name VARCHAR(20),

  Semester NUMERIC

);

INSERT INTO Students VALUES (24002, 'Xenokrates', 18);

INSERT INTO Students VALUES (25403, 'Jonas', 12);

INSERT INTO Students VALUES (26120, 'Fichte', 10);

INSERT INTO Students VALUES (26830, 'Aristoxenos', 8);

INSERT INTO Students VALUES (27550, 'Schopenhauer', 6);

INSERT INTO Students VALUES (28106, 'Carnap', 3);

INSERT INTO Students VALUES (29120, 'Theophrastos', 2);

INSERT INTO Students VALUES (29555, 'Feuerbach', 2);

	StudNr decimal(5,0)	Name varchar(20)	Semester decimal(10,0)
	Filter	Filter	Filter
1	24002	Xenokrates	18
2	25403	Jonas	12
3	26120	Fichte	10
4	26830	Aristoxenos	8
5	27550	Schopenhauer	6
6	28106	Carnap	3
7	29120	Theophrastos	2
8	29555	Feuerbach	2

CREATE Table Attend

(StudNr NUMERIC(5),

LectureNr NUMERIC(4));

INSERT INTO Attend VALUES (26120, 5001);

INSERT INTO Attend VALUES (27550, 5001);

INSERT INTO Attend VALUES (27550, 4052);

INSERT INTO Attend VALUES (28106, 5041);

INSERT INTO Attend VALUES (28106, 5052);

INSERT INTO Attend VALUES (28106, 5216);

INSERT INTO Attend VALUES   (28106, 5259);

INSERT INTO Attend VALUES (29120, 5001);

INSERT INTO Attend VALUES (29120, 5041);

INSERT INTO Attend VALUES (29120, 5049);

INSERT INTO Attend VALUES (25403, 5022);

INSERT INTO Attend VALUES (29555, 5022);

INSERT INTO Attend VALUES (29555, 5001);

StudNr decimal(5,0)	LectureNr decimal(4,0)
Filter	Filter
1	26120	5001
2	26120	5001
3	27550	5001
4	27550	4052
5	28106	5041
6	28106	5052
7	28106	5216
8	28106	5259
9	29120	5001
10	29120	5041
11	29120	5049
12	25403	5022
13	29555	5022
14	29555	5001

CREATE Table Lectures(

   LectureNr NUMERIC(4) PRIMARY KEY,

   Title VARCHAR(20)

 );

INSERT INTO Lectures VALUES(5001, 'Grundzuge');

INSERT INTO Lectures VALUES(5041, 'Ethik');

INSERT INTO Lectures VALUES(5043, 'Erkenntnistheorie');

INSERT INTO Lectures VALUES(5049, 'Maeutik');

INSERT INTO Lectures VALUES(4052, 'Logik');

INSERT INTO Lectures VALUES(5052, 'Wissenschaftstheorie');

INSERT INTO Lectures VALUES(5216, 'Bioethik');

INSERT INTO Lectures VALUES(5259, 'Der Wiener Kreis');

INSERT INTO Lectures VALUES(5022, 'Glaube und Wissen');

INSERT INTO Lectures VALUES(4630, 'Die 3 Kritiken');

	* LectureNr decimal(4,0)	Title varchar(20)
1	4052	Logik
2	4630	Die 3 Kritiken
3	5001	Grundzuge
4	5022	Glaube und Wissen
5	5041	Ethik
6	5043	Erkenntnistheorie
7	5049	Maeutik
8	5052	Wissenschaftstheorie
9	5216	Bioethik
10	5259	Der Wiener Kreis

Queries:

All students with their name and number of lectures

SELECT s.name, Count(a.LectureNr) AS 'Number of lectures' FROM Attend a, Students s

WHERE a.StudNr = s.StudNr GROUP BY s.name;

Average semester of all students

SELECT AVG(Semester) As 'Average semester' FROM Students;

Average number of lectures per student

SELECT @total: = COUNT(LectureNr) FROM Attend;

SELECT s.name, Count(a.LectureNr) / @total AS 'Average Lectures'

FROM Attend a, Students s

WHERE a.StudNr = s.StudNr

GROUP BY s.name;

Students who attend lectures which also Fiche attends

SELECT s.name FROM Students s

WHERE StudNr IN (SELECT StudNr FROM Attend WHERE LectureNr IN

(SELECT LectureNr FROM Attend WHERE StudNr IN

(SELECT StudNr FROM Students WHERE name = 'Fichte')));

Q	name ⬍
	Filter
1	Fichte
2	Schopenhauer
3	Theophrastos
4	Feuerbach

Titles of lectures which are attended by students from the first 4 semesters

SELECT Title FROM Lectures WHERE LectureNr IN

(SELECT LectureNr FROM Attend WHERE StudNr IN

(SELECT StudNr FROM Students WHERE Semester <= 4));

Q	Title varchar(20) ⬍
	Filter
1	Ethik
2	Wissenschaftstheorie
3	Bioethik
4	Der Wiener Kreis
5	Grundzuge
6	Maeutik
7	Glaube und Wissen

Lab Assignment 11

Q1) Write a PL/SQL program to arrange the number of two variable in such a way that the small number will store in num_small variable and large number will store in num_large variable.

Code:

```
DECLARE

num_small NUMBER;

num_large NUMBER;

num_temp NUMBER;

BEGIN

num_small := :num_small;

num_large := :num_large;

IF num_small > num_large

THEN

num_temp := num_small;

num_small := num_large;

num_large := num_temp;

END IF;

DBMS_OUTPUT.PUT_LINE('num_small = '||num_small);

DBMS_OUTPUT.PUT_LINE('num_large = '||num_large);

END;
```

Output:

	Submit
Bind Variable	**Value**
:NUM_SMALL	10
:NUM_LARGE	2

Q2) Write a PL/SQL procedure to calculate the incentive on a target achieved and display the message either the record updated or not.

Code:

```
DECLARE

PROCEDURE test1 (

sal_achieve  NUMBER,

target_qty  NUMBER,

emp_id NUMBER

)

IS

incentive   NUMBER := 0;

updated  VARCHAR2(3) := 'No';

BEGIN

IF sal_achieve > (target_qty + 200) THEN

incentive := (sal_achieve - target_qty)/4;

UPDATE emp

SET salary = salary + incentive

WHERE employee_id = emp_id;

updated := 'Yes';

END IF;

DBMS_OUTPUT.PUT_LINE (

'Table updated?  ' || updated || ', ' ||
```

'incentive = ' || incentive || '.'

);

END test1;

BEGIN

test1(2300, 2000, 144);

test1(3600, 3000, 145);

END;

Output:

```
Table updated?  Yes, incentive = 75.
Table updated?  Yes, incentive = 150.


PL/SQL procedure successfully completed.
```

Q3) Write a PL/SQL program to check whether a number is even or odd.

Code:

DECLARE

num NUMBER;

BEGIN

num := :num;

IF (MOD(num,2) = 0)

THEN

DBMS_OUTPUT.PUT_LINE('even number');

ELSE

DBMS_OUTPUT.PUT_LINE('odd number');

END IF;

END;

Output:

Bind Variable	Value
:NUM	346541

Results   Explain   Describe   Saved SQL   History

odd number

Statement processed.

0.00 seconds

Q4) Write a PL/SQL procedure to calculate the incentive on a specific target otherwise a general incentive to be paid using IF-THEN-ELSE.

Code:

DECLARE

PROCEDURE test1 (

sal_achieve NUMBER,

target_qty NUMBER,

emp_id NUMBER

)

IS

incentive   NUMBER := 0;

BEGIN

IF sal_achieve > (target_qty + 200) THEN

incentive := (sal_achieve - target_qty)/4;

ELSE

incentive :=75;

END IF;

DBMS_OUTPUT.PUT_LINE ('incentive = ' || incentive);

UPDATE emp

SET salary = salary + incentive

WHERE employee_id = emp_id;

END test1;

BEGIN

test1(2300, 2000, 144);

test1(3600, 3000, 145);

END;

Output:

```
incentive = 75
incentive = 150
```

Q5) Write a PL/SQL program to check whether a date falls on weekend i.e. SATURDAY.

Code:

DECLARE

dat DATE;

d CHAR(3);

BEGIN

dat := DATE '2021-10-24';

d := TO_CHAR(dat, 'DY');

IF (d = 'SAT' OR d = 'SUN')

THEN

DBMS_OUTPUT.PUT_LINE('weekend');

ELSE

DBMS_OUTPUT.PUT_LINE('weekday');

END IF;

END;

Output:

Q6) Write a PL/SQL procedure to calculate incentive achieved according to the specific sale limit.

Code:

```
DECLARE

PROCEDURE test1 (sal_achieve  NUMBER)

IS

incentive  NUMBER := 0;

BEGIN

IF sal_achieve > 44000 THEN

incentive := 1800;

ELSIF sal_achieve > 32000 THEN

incentive := 800;

ELSE

incentive := 500;

END IF;

DBMS_OUTPUT.NEW_LINE;

DBMS_OUTPUT.PUT_LINE (

'Sale achieved : ' || sal_achieve || ', incentive : ' || incentive || '.'

);

END test1;

BEGIN

test1(45000);
```

test1(36000);

test1(28000);

END;

Output:

```
Sale achieved : 45000, incentive : 1800.

Sale achieved : 36000, incentive : 800.

Sale achieved : 28000, incentive : 500.

PL/SQL procedure successfully completed.
```

Q7) Write a PL/SQL program to count number of employees in department 50 and check whether this department have any vacancies or not. There are 45 vacancies in this department.

Code:

```
SET SERVEROUTPUT ON

DECLARE

tot_emp NUMBER;

BEGIN

SELECT Count(*)

INTO   tot_emp

FROM   employees e

join departments d

ON e.department_id = d.department_id

WHERE  e.department_id = 50;

dbms_output.Put_line ('The employees are in the department 50: '

||To_char(tot_emp));

IF tot_emp >= 45 THEN

dbms_output.Put_line ('There are no vacancies in the department 50.');

ELSE

dbms_output.Put_line ('There are some vacancies in department 50.');

END IF;

END;
```

Output:

```
The employees are in the department 50: 45
There are no vacancies in the department 50.

PL/SQL procedure successfully completed.
```

Lab Assignment 12

Questions:

19. Write a program in PL/SQL to update the salary of a specific employee by 8% if the salary exceeds the mid range of the salary against this job and update up to mid range if the salary is less than the mid range of the salary, and display a suitable message.

20. Write a program in PL/SQL using nested loop with EXIT WHEN statement.

21. Write a program in PL/SQL using loop with CONTINUE statement.

22. Write a program in PL/SQL using loop with CONTINUE WHEN statement.

23. Write a program in PL/SQL to print 1st n numbers.

Code:

Q23)

```
DECLARE

num NUMBER := :n;

itr NUMBER;

BEGIN

DBMS_OUTPUT.PUT_LINE('The first '||num||' numbers are: ');

for itr in 1..num loop

DBMS_OUTPUT.PUT(itr||' ');

END LOOP;

dbms_output.new_line;

END;
```

Q21)

```
DECLARE

num NUMBER := 0;

BEGIN

LOOP

DBMS_OUTPUT.PUT_LINE('Inside the loop:  n = ' || TO_CHAR(num));

num := num  + 1;

IF(num < 5)

THEN

CONTINUE;
```

```
END IF;

DBMS_OUTPUT.PUT_LINE('Inside the loop, after CONTINUE:  n = ' || TO_CHAR(num));

EXIT WHEN num = 7;

END LOOP;

DBMS_OUTPUT.PUT_LINE ('When out from the loop:  n = ' || TO_CHAR(num));

END;
```

Q20)

```
DECLARE

m NUMBER := 0;

BEGIN

LOOP

m := m + 1;

DBMS_OUTPUT.PUT_LINE('The value of the m is: '||m);

EXIT WHEN m > 4;

END LOOP;

END;
```

Q22)

```
DECLARE

n NUMBER := 0;

BEGIN

LOOP -- After CONTINUE statement, control resumes here

DBMS_OUTPUT.PUT_LINE ('The value inside the loop:  n = ' || TO_CHAR(n));

n := n + 1;

CONTINUE WHEN n < 4;

DBMS_OUTPUT.PUT_LINE

('The value inside loop, after CONTINUE:  n = ' || TO_CHAR(n));

EXIT WHEN n = 6;

END LOOP;

DBMS_OUTPUT.PUT_LINE ('The value after exit from the loop:  n = ' || TO_CHAR(n));

END;
```

Q19)

```
DECLARE

emp_min_salary NUMBER(6,0);

emp_max_salary NUMBER(6,0);

emp_mid_salary NUMBER(6,2);

tmp_salary    EMPLOYEES.SALARY%TYPE;

tmp_emp_id    EMPLOYEES.EMPLOYEE_ID%TYPE := 167;

tmp_emp_name        EMPLOYEES.FIRST_NAME%TYPE;

BEGIN

SELECT min_salary,

max_salary

INTO emp_min_salary,

emp_max_salary

FROM JOBS

WHERE JOB_ID = (SELECT JOB_ID

FROM EMPLOYEES

WHERE EMPLOYEE_ID = tmp_emp_id);

-- calculate mid-range

emp_mid_salary := (emp_min_salary + emp_max_salary) / 2;

-- get salary of the given employee

SELECT salary,first_name

INTO tmp_salary,tmp_emp_name

FROM employees

WHERE employee_id = tmp_emp_id;

-- update salary

IF tmp_salary < emp_mid_salary THEN

UPDATE employees

SET salary = emp_mid_salary

WHERE employee_id = tmp_emp_id;

ELSE
```

```
UPDATE employees

SET salary = salary + salary * 8 /100

WHERE employee_id = tmp_emp_id;

END IF;

--display message

IF tmp_salary > emp_mid_salary THEN

DBMS_OUTPUT.PUT_LINE('The      employee      '||tmp_emp_name||'      ID      '      ||
TO_CHAR(tmp_emp_id) ||

' works in salary ' || TO_CHAR(tmp_salary) ||

' which is higher than mid-range of salary ' || TO_CHAR(emp_mid_salary));

ELSIF tmp_salary < emp_mid_salary THEN

DBMS_OUTPUT.PUT_LINE('The      employee      '||tmp_emp_name||'      ID      '      ||
TO_CHAR(tmp_emp_id) ||

' works in salary ' || TO_CHAR(tmp_salary) ||

' which is lower than mid-range of salary ' || TO_CHAR(emp_mid_salary));

ELSE

DBMS_OUTPUT.PUT_LINE('The      employee      '||tmp_emp_name||'      ID      '      ||
TO_CHAR(tmp_emp_id) ||

' works in salary ' || TO_CHAR(tmp_salary) ||

' which is equal to the mid-range of salary ' || TO_CHAR(emp_mid_salary));

END IF;

END;
```
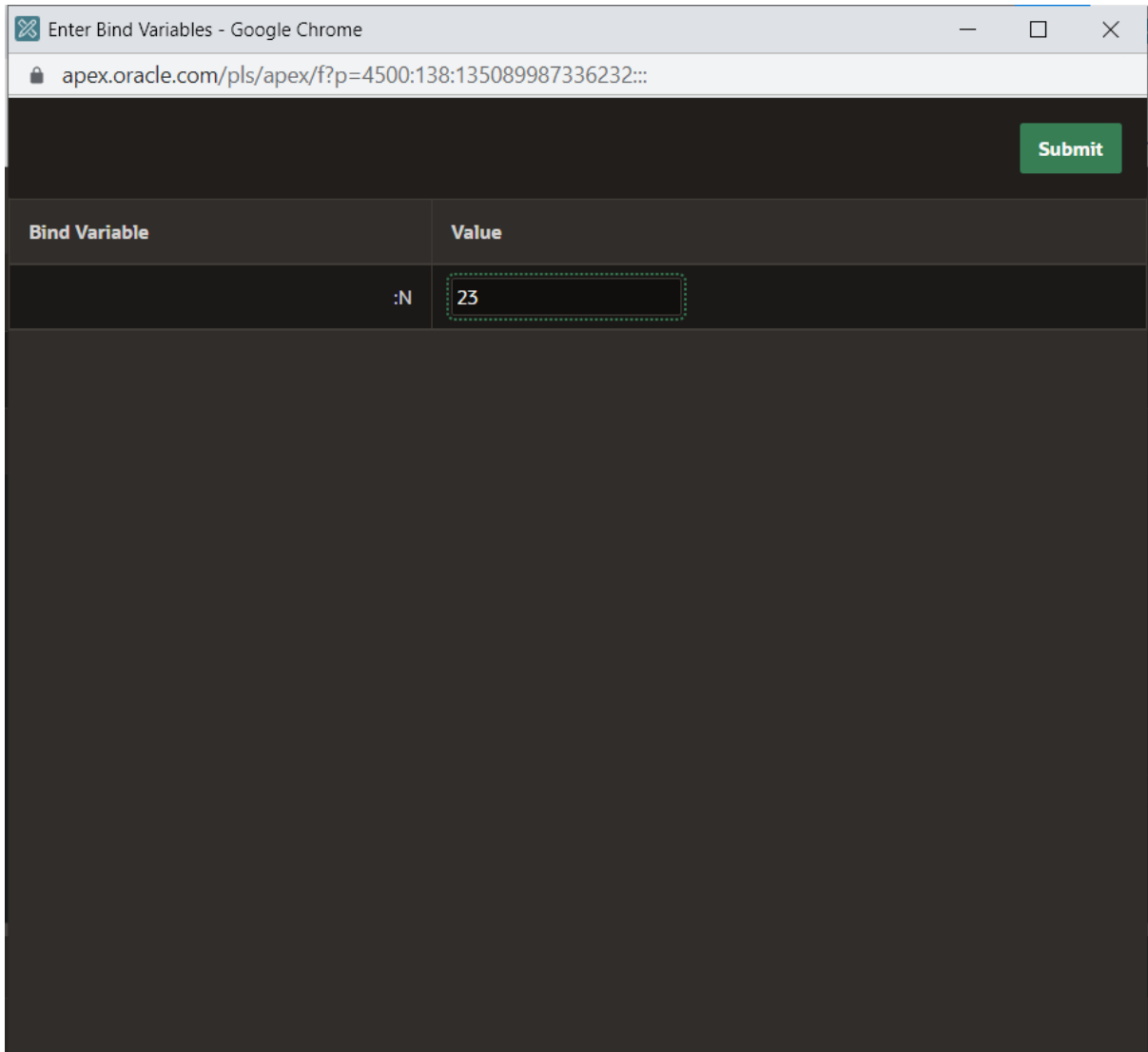
Output:

Q23)

Submit

Bind Variable	Value
:N	23

```
The first 23 numbers are:
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

Statement processed.


0.02 seconds
```

Q20)

```
Inside the loop:  n = 0
Inside the loop:  n = 1
Inside the loop:  n = 2
Inside the loop:  n = 3
Inside the loop:  n = 4
Inside the loop, after CONTINUE:  n = 5
Inside the loop:  n = 5
Inside the loop, after CONTINUE:  n = 6
Inside the loop:  n = 6
Inside the loop, after CONTINUE:  n = 7
When out from the loop:  n = 7

Statement processed.


0.01 seconds
```

Q21)

```
Inside the loop:  n = 0
Inside the loop:  n = 1
Inside the loop:  n = 2
Inside the loop:  n = 3
Inside the loop:  n = 4
Inside the loop, after CONTINUE:  n = 5
Inside the loop:  n = 5
Inside the loop, after CONTINUE:  n = 6
Inside the loop:  n = 6
Inside the loop, after CONTINUE:  n = 7
When out from the loop:  n = 7

Statement processed.


0.01 seconds
```

Q22)

```
The value inside the loop:   n = 0
The value inside the loop:   n = 1
The value inside the loop:   n = 2
The value inside the loop:   n = 3
The value inside loop, after CONTINUE:   n = 4
The value inside the loop:   n = 4
The value inside loop, after CONTINUE:   n = 5
The value inside the loop:   n = 5
The value inside loop, after CONTINUE:   n = 6
The value after exit from the loop:   n = 6


Statement processed.


0.01 seconds
```

Q19)

```
The employee Amit ID 167 works in salary 6200 which is lower than mid-range of salary 9004

PL/SQL procedure successfully completed.
```