

```
DROP TABLE IF EXISTS `customers`;
```

```
create TABLE customers (  
  customer_Id int(11) NOT NULL,  
  First_Name varchar(50) NOT NULL,  
  Last_Name varchar(50) NOT NULL,  
  phone varchar(50) NOT NULL,  
  creditLimit decimal(10,2) DEFAULT NULL,  
  PRIMARY KEY (`customer_Id`)  
);
```

```
insert into customers(customer_Id,First_Name,Last_Name,phone,creditLimit)  
values
```

```
(103,'Atelier','Schmitt','08-7896  
6578','21000.00'),(112,'Signal','King','7025551838','71800.00'),  
(114,'Ferguson','Peter','03-9520 4555','117300.00'),(119,'Labruno','Janine  
,','40.67.8555','118200.00'),  
(121,'Bergulfsen','Jonas ',','07-98  
9555','81700.00'),(124,'Nelson','Susan','4155551450','210500.00'),  
(125,'Piestrzeniewicz','Zbyszek ',','(26) 642-7555','0.00'),(128,'Keitel','Roland','+49 69  
66 90 2555','59700.00'),  
(129,'Murphy','Julie','6505555787','64600.00'),(131,'Lee','Kwai','2125557818','114900  
.00'),  
(141,'Freyre','Diego ',','(91) 555 94 44','227600.00'),(144,'Berglund','Christina  
,','0921-12 3555','53100.00'),  
(145,'Petersen','Jytte ',','31 12 3555','83400.00'),(146,'Saveley','Mary  
,','78.32.5555','123900.00'),  
(148,'Eric','Jacob','+65 221  
7555','103800.00'),(151,'Young','Jeff','2125557413','138500.00');
```

```
DROP TABLE IF EXISTS `orders`;
```

```
CREATE TABLE orders(  
  order_Id int(11) NOT NULL,  
  order_Date date NOT NULL,  
  shipped_Date date DEFAULT NULL,  
  Deliver varchar(15) NOT NULL,  
  customer_Id int(11) NOT NULL,  
  PRIMARY KEY (order_Id),  
  FOREIGN KEY (customer_Id) REFERENCES customers(customer_Id)  
);
```

```
insert into orders(order_Id,order_Date,shipped_Date,Deliver,customer_Id) values  
(10100,'2003-01-06','2003-01-13','Shipped',114),(10101,'2003-01-09','2003-01-18','Sh  
ipped',125),  
(10102,'2003-01-10','2003-01-18','Shipped',129),(10103,'2003-01-29','2003-02-07','Sh  
ipped',121),  
(10104,'2003-01-31','2003-02-09','Shipped',141),(10105,'2003-02-11','2003-02-21','Sh  
ipped',145);
```

1. Write a Query to add a column package_stat to the table orders.
2. Write a Query to change the package_stat column of orders table with 'not available' for all orders.
3. Write a Query to delete a row from customers table where credit_limit is 0.00

Write SELECT statements to achieve the following:

1. Write a Query to display the first_name with the occurrence of 'el' in the customers tables.
2. Write a Query to prepare a list with customer name ,customer_id ,order_id for the customers whose delivery status is shipped.
3. Write a Query to get the number of customers with the creditLimit greater than 50000.
4. Write a Query to display the customer_id, name (first name and last name), order_id and deliver for all customers.
5. Write a Query to customer name in order of creditLimit smallest to highest.
6. Write a stored procedure by name order_day. The procedure should show the customer_id and the day on which he had made the order.
7. Write a stored function by the name of cutomer_search. The stored function should return the maximum creditLimit made by any customer.

#####

1. Create DEPT table with the following structure:-

```
DEPTNO    INT(2)
DNAME     VARCHAR (14)
LOC       VARCHAR (13)
```

Insert the following rows in DEPT table:-

```
10, ACCOUNTING, NEW YORK
20, RESEARCH, DALLAS
30, SALES, CHICAGO
40, OPERATIONS, BOSTON
```

2. Create EMP table with the following structure:-
- ```
EMPNO INT(4)
```

|          |              |
|----------|--------------|
| ENAME    | VARCHAR (10) |
| JOB      | VARCHAR (9)  |
| HIREDATE | DATE         |
| SAL      | FLOAT(7,2)   |
| COMM     | FLOAT(7,2)   |
| DEPTNO   | INT(2)       |

Insert the following rows in EMP table:-

```

7369, SMITH, CLERK, 17-DEC-80, 800, null, 20
7499, ALLEN, SALESMAN, 20-FEB-81, 1600, 300, 30
7521, WARD, SALESMAN, 22-FEB-81, 1250, 500, 30
7566, JONES, MANAGER, 02-APR-81, 2975, null, 20
7654, MARTIN, SALESMAN, 28-SEP-81, 1250, 1400, null, 30
7698, BLAKE, MANAGER, 01-MAY-81, 2850, null, 30
7782,, CLARK, MANAGER, 09-JUN-81, 2450, null, 10
7788, SCOTT, ANALYST, 09-DEC-82, 3000, null, 20
7839, KING, PRESIDENT, 17-NOV-81, 5000, null, 10
7844, TURNER, SALESMAN, 08-SEP-81, 1500, 0, 30
7876, ADAMS, CLERK, 12-JAN-83, 1100, null, 20
7900, JAMES, CLERK, 03-DEC-81, 950, null, 30
7902, FORD, ANALYST, 03-DEC-81, 3000, null, 20
7934, MILLER, CLERK, 23-JAN-82, 1300, null, 10

```

Write SELECT statements to achieve the following:-

1. Display only the EMPNO and ENAME columns from EMP table.
2. Display all employees who are CLERKS and the MANAGERS.
3. Display the ENAME and JOB for all employees who belong to the same DEPTNO as employee 'KING'.
4. Find the names of all employees hired in the month of February (of any year).
5. Display the employees in descending order of DEPTNO.
6. Display the employee name and employee number of the employees with the headings as NUMBER and NAME.
7. Find the names of all employees who were hired on the last day of the month.
8. Find the name of the employee who is receiving the maximum salary.
9. Display the sum of SAL for all the employees belonging to DEPTNO 10.
10. Display the rows where JOB column ends with the letter 'T'.

11 Write a stored procedure to convert a temperature in Fahrenheit (F) to its equivalent in Celsius (C). The required formula is:-  $C = (F - 32) * 5/9$

Insert the temperature in Centigrade into TEMPP table. Calling program for the stored procedure need not be written.

12. Write a stored function by the name of Num\_cube. The stored function should return the cube of a number 'N'. The number 'N' should be passed to the stored function as a parameter. Calling program for the stored function need not be written.