CONSIDER FOLLOWING TABLES.

CREATE FOLLOWING TABLES WITH APPROPRIATE CONSTRAINTS AS PER THE GIVEN SPECIFICATIONS.

1. **SALESMAN**

SNUM	SNAME	CITY	COMMISSION
1001	PIYUSH	LONDON	12
1002	NIRAJ	SURAT	13
1003	MITI	LONDON	11
1004	RAJESH	BARODA	15
1005	ANAND	NEW DELHI	10
1006	RAM	PATAN	10
1007	LAXMAN	BOMBAY	9

DESCRIPTION OF ATTRIBUTES

1. SNUM : A UNIQUE NUMBER ASSIGN TO EACH SALESMAN.
2. SNAME : THE NAME OF SALESMAN.
3. CITY : THE LOCATION OF SALESMAN.
4. COMMISSION : THE SALESMAN COMMISSION ON ORDER.

2. **CUSTOMER**

CNUM	CNAME	CITY	RATING	SNUM
2001	HARDIK	LONDON	100	1001
2002	GITA	ROME	200	1003
2003	LAXIT	SURAT	200	1002
2004	GOVIND	BOMBAY	300	1002
2005	CHANDU	LONDON	100	1001
2006	CHAMPAK	SURAT	300	1007
2007	PRATIK	ROME	100	1004

DESCRIPTION OF ATTRIBUTES
1. CNUM : A UNIQUE NUMBER ASSIGN TO EACH CUSTOMER. CNAME :
2. THE NAME OF CUSTOMER.
3. CITY : THE LOCATION OF CUSTOMER.
4. RATING : A LEVEL OF PREFERENCE INDICATOR GIVEN TO THIS CUSTOMER.
5. SNUM : A SALESMAN NUMBER ASSIGN TO THIS CUSTOMER.

3. **ORDERS**

ONUM	AMOUNT	ODATE	CNUM	SNUM
3001	18.69	10/03/99	2007	1007

3002	767.19	10/03/99	2001	1001
3003	1900.10	10/03/99	2007	1004
3004	5160.45	10/03/99	2003	1002
3005	1098.25	10/04/99	2008	1007
3006	1713.12	10/04/99	2002	1003
3007	75.75	10/05/99	2004	1002
3008	4723.00	10/05/99	2006	1001
3009	1309.95	10/05/99	2004	1002
3010	9898.87	10/06/99	2006	1001

DESCRIPTION OF ATTRIBUTES

1. ONUM : A UNIQUE NUMBER ASSIGN TO EACH ORDER.
2. AMOUNT : AMOUNT OF ORDER IN RS.
3. ODATE : THE DATE OF ORDER.
4. CNUM : THE NUMBER OF CUSTOMER MAKING THE ORDER.
5. SNUM : THE NUMBER OF SALESMAN CREDITED WITH THE SALE.

OBJECTIVE	LEARN WRITING STORED FUNCTION. CALLING A FUNCTION FROM STORED PROCEDURE
NEW TERMS	CREATE FUNCTION CALL

NO.	PROBLEM STATEMENT	TEACHER'S SIGN WITH DATE
1.	WRITE A FUNCTION P_CNUM THAT RETURNS THE NAME OF CUSTOMER WHOSE CUSTOMER NO IS 2001. WRITE A PROCEDURE PR_PRINT THAT WILL DISPLAY THE NAME OF CUSTOMER CALLING THE ABOVE FUNCTION. **WRITE APPROPRIATE COMMNET THAT WILL EXPLAIN EACH STATEMENT EXPLICITLY.** EXAMPLE: HOW TO WRITE COMMENTS **/* THE FOLLOWING STATEMENT WILL DROP THE PROCEDURE P_CNUM IF IT IS ALREADY CREATED */** DROP PROCEDURE IF EXISTS P_CNUM; **/* THE FOLLOWING STATEMENT WILL CHANGE THE DELIMITER TO // */**	

```
DELIMITER  //
```

/* THE FOLLOWING STATEMENT WILL CREATED PROCEDURE NAMED P_CNUM */

```
CREATE PROCEDURE P_CNUM()
```

/* MARK THE BEGINNING OF THE BLOCK */

```
BEGIN
```

/* THE FOLLOWING STATEMENT WILL DECLARE A VARIABLE V_CNAME OF TYPE INTEGER */

```
    DECLARE V_CNAME VARCHAR(20);
    DECLARE V_CNUM INT;
```

/* THE FOLLOWING STATEMENT WILL FETCH CNAME AND CNUM OF CUSTOMER NUMBER 2001 FROM CUSTOMER TABLE AND ASSIGN ITS VALUE TO PROCEDURE VARIABLES V_CNAME AND V_CNUM RESPECTIVELY */

```
    SELECT CNAME, CNUM INTO V_CNAME, V_CNUM FROM
    CUSTOMER WHERE CNUM =2001;
```

/* THE FOLLOWING STATEMENT WILL DISPLAY THE VALUE OF VARIABLES V_CNAME AND V_CNUM AS C_NAME AND C_NUM RESPECTIVELY */

```
    SELECT V_CNAME AS C_NAME, V_CNUM AS C_NUM;
```

/* MARK THE ENDING OF THE BLOCK */

```
END //
```

/* THE FOLLOWING STATEMENT WILL CHANGE THE DELIMITER TO ; */

```
DELIMITER ;
```

/* THE FOLLOWING STATEMENT WILL CALL THE PROCEDURE P_CNUM */

	CALL P_CNUM; /* OUTPUT*/		
2.	WRITE A FUNCTION P_ORDER THAT RETURNS THE CUSTOMER NUMBER HAVING HIGHEST TOTAL ORDER AMOUNT. WRITE A PROCEDURE PR_PRINT THAT WILL DISPLAY THE CUSTOMER NUMBER BY CALLING THE ABOVE FUNCTION IN THE FOLLOWING FORMAT. CUSTOMER NUMBER XXX HAS PLACED THE ORDER WITH HIGHEST AMOUNT. **WRITE APPROPRIATE COMMNET THAT WILL EXPLAIN EACH STATEMENT EXPLICITLY.**	3	
3.	WRITE A FUNCTION P_INCREMENT THAT WILL ACCEPT CUSTOMER NUMBER AND INCREASE IT'S RATING BY 150. THE FUNCTION WILL RETURN THE INCREASED RATTING. WRITE A PROCEDURE PR_UPDATE THAT UPDATE THE INCREASED RATTING INTO CUSTOMER TABLE. **WRITE APPROPRIATE COMMNET THAT WILL EXPLAIN EACH STATEMENT EXPLICITLY.**		

4.	WRITE A FUNCTION P_GRADE THAT WILL ACCEPT CUSTOMER NUMBER AND RETURNS ITS GRADE.
	GRADE WILL BE DECIDED ACCORDING TO FOLLOWING RULES.
	    1. IF RATING IS 100 THEN GRADE WILL BE 'POOR'.
	    2. IF RATING IS 200 THEN GRADE WILL BE 'GOOD'.
	    3. IF RATING IS 300 THEN GRADE WILL BE 'EXCELLENT'.
	USE SIMPLE CASE STRUCTURE
	WRITE A PROCEDURE PR_PRINT THAT WILL DISPLAY THE CUSTOMER GRADE BY CALLING THE ABOVE FUNCTION.		

5.	REWRITE THE PROBLEM STATEMENT 4 WITHOUT USING CASE STRUCTURE.		