**/\* 6.6.1\*/**

**CREATE OR REPLACE PROCEDURE allowance(pcust\_code IN NUMERIC )  AS**

**mallowance      FLOAT;**

**msalary               FLOAT;**

**mcode                Number;**

**over\_salary        EXCEPTION;**

**BEGIN**

**SELECT cust\_code, salary INTO mcode,msalary**

**FROM INFORM**

**WHERE cust\_code = pcust\_code;**

**IF(msalary > 90000)**

**THEN**

**raise over\_salary;**

**ELSE**

**mallowance := msalary \* .10;**

**msalary := msalary + mallowance;**

**Dbms\_output.put\_line(mcode||'   '||msalary||'    '||'regular') ;**

**END IF;**

**EXCEPTION**

**WHEN over\_salary**

**THEN**

**Dbms\_output.put\_line(mcode||'   '||msalary||'  '||'over salary');**

**WHEN OTHERS**

**THEN**

**msalary:=msalary+500;**

**Dbms\_output.put\_line(mcode||'    '||msalary||'     '||'other criteria');**

**END allowance;**

**/\*6.7.1\*/**

**BEGIN**

**allowance(10);**

**END;**

**/\*6.7.2\*/**  
**DECLARE**

**CURSOR c1 is SELECT cust\_code, organization FROM   INFORM;**

**BEGIN**

**FOR e1 IN c1**

**LOOP**

**IF(e1.organization != 'Self')**

**THEN**

**allowance(e1.cust\_code);**

**END IF;**

**END LOOP;**

**END;**

**/\* 6.7.3\*/**

**CREATE OR REPLACE PROCEDURE bonus1 (pcust\_code IN NUMERIC ) AS**

**mbonus FLOAT;**

**msalary FLOAT;**

**BEGIN**

**SELECT salary INTO msalary FROM inform**

**WHERE cust\_code = pcust\_code;**

**mbonus := msalary \* .10;**

**insert into bt values(mbonus);**

**END bonus1;**

**/\*6.7.4\*/**

**CREATE OR REPLACE PROCEDURE Bonus2(pcust\_code IN NUMERIC )  AS**

**mbonus      FLOAT;**

**msalary     FLOAT;**

**BEGIN**

**/\* this is invalid statement because we are trying to assign     a value to a  parameter which is of mode IN \*/**

**pcust\_code := 10;**

**SELECT salary INTO msalary FROM inform**

**WHERE cust\_code = pcust\_code;**

**mbonus := msalary \* .10;**

**insert into bt values(mbonus);**

**END Bonus2;**

**/\*6.7.5\*/**

**CREATE OR REPLACE PROCEDURE bonus3(pcust\_code IN OUT NUMERIC )**

**AS**

**mbonus      FLOAT;**

**msalary     FLOAT;**

**mcode        Number;**

**BEGIN**

**/\* now this is valid because pcust\_code is of mode IN OUT \*/**

**pcust\_code := 10;**

**/\* reading pcust\_code \*/**

**SELECT cust\_code, salary INTO mcode, msalary FROM INFORM**

**WHERE cust\_code = pcust\_code;**

**mbonus := msalary \* .10;**

**Dbms\_output.put\_line(mcode||'     '||mbonus|);**

**END bonus3;**

**/\*6.7.6\*/**

**DECLARE**

**ccode    Number;**

**BEGIN**

**Bonus3(ccode);**

**END;**

**/\*6.7.7\*/**

**CREATE OR REPLACE PROCEDURE Bonus4(pcust\_code IN NUMERIC, mbonus OUT number ) AS**

**msalary FLOAT;**

**mcode number;**

**BEGIN**

**SELECT cust\_code, salary INTO mcode, msalary**

**FROM inform**

**WHERE cust\_code = pcust\_code;**

**mbonus := msalary \* .10;**

**Dbms\_output.put\_line('mcode'||'    '|| mcode||'   '|| 'mbonus'||'           '||mbonus);**

**END Bonus4;**

**/\*6.7.8\*/**

**DECLARE**

**gbonus number(6);**

**/\* gbonus corresponds to mbonus in the procedure bonus4()\*/**

**BEGIN**

**Bonus4(10, gbonus);**

**INSERT into bt values(gbonus);**

**END;**

**/\*6.9.1\*/**

**CREATE OR REPLACE PROCEDURE bonus6(pcust\_code IN NUMERIC, per\_bonus IN  FLOAT ) AS**

**msalary FLOAT;**

**mbonus FLOAT;**

**BEGIN**

**/\* reading pcust\_code, its valid \*/**

**SELECT salary**

**INTO msalary FROM INFORM**

**WHERE cust\_code = pcust\_code;**

**mbonus :=  msalary \* per\_bonus;**

**Dbms\_output.put\_line('Bonus'||'             '||mbonus);**

**END bonus6;**

**/\*6.9.2(a)\*/**

**BEGIN**

**bonus6 (10, 0.20);**

**END;**

**/\* 6.9.2(b)\*/**

**BEGIN**

**Bonus6 (per\_bonus=> 0.20, pcust\_code =>10);**

**END;**

**/\*6.9.2 ( c)\*/**

**BEGIN**

**Bonus6 ( pcust\_code =>10,0.20);**

**END;**

**/\*6.9.3\*/**

**CREATE OR REPLACE PROCEDURE bonus7(pcust\_code IN NUMERIC, per\_bonus  IN FLOAT DEFAULT  0.20 ) AS**

**msalary FLOAT;**

**mbonus  FLOAT;**

**BEGIN**

**/\* reading pcust\_code, its valid \*/**

**SELECT salary INTO msalary FROM inform**

**WHERE cust\_code = pcust\_code;**

**mbonus := msalary \* per\_bonus;**

**Dbms\_output.put\_line('Bonus'|| '             '||mbonus);**

**END bonus7;**

/**\*6.9.4\*/**

**BEGIN**

**Bonus7 (pcust\_code=>10);**

**END;**

**/\* 6.10.1\*/**

**Create  procedure faculty\_resign( teacher NUMBER )**

**As**

**BEGIN**

**DELETE FROM FACULTY**

**WHERE teacherno = teacher;**

**/\* if that teacherno doesn't exist, then raise error \*/**

**IF SQL%NOTFOUND**

**THEN**

**raise\_application\_error( -20005, 'Teacher no: '**

**||to\_char(teacher) || ' resigned already');**

**END IF;**

**END faculty\_resign;**

**/\*6.10.2\*/**

**BEGIN**

**faculty\_resign(8);**

**END;**

**/\*6.10.3\*/**

**Create Procedure u\_str\_pt(oldname in varchar2,**

**newname in varchar2)**

**As**

**/\* cursor declared for update of str\_pt in TRAIN table\*/**

**CURSOR c4 is SELECT str\_pt**

**FROM TRAIN for UPDATE of str\_pt;**

**BEGIN**

**For train\_rec in c4**

**LOOP**

**IF (train\_rec.str\_pt = oldname)**

**THEN**

**/\*here the current row is updated, without**

**searching the entire table \*/**

**update TRAIN**

**set str\_pt = newname**

**where current of c4;**

**END IF;**

**END LOOP;**

**END u\_str\_pt;**

/**\* 6.10.4\*/**

**BEGIN**

**U\_STR\_PT('Delhi', 'N.Delhi');**

**END;**

**/\*6.10.5\*/**

**CREATE PROCEDURE**

**REPNAME( license\_no in numeric, newname in varchar2) as**

**Begin**

**UPDATE  LICENSE**

**SET place\_issued = newname**

**WHERE lic\_no = license\_no;**

**End REPNAME**

**/\*6.10.5\*/**

**CREATE PROCEDURE**

**repname( license\_no in numeric, newname in varchar2) as**

**Begin**

**UPDATE  LICENSE**

**SET place\_issued = newname**

**WHERE lic\_no = license\_no;**

**End repname;**

/\*6.10.5(a)\*/

**Execute repname(13181, 'SRINAGAR’');**

**/\*6.10.6\*/**

**Create procedure Quantity( mqty number )**

**as**

**CURSOR c1 is SELECT cname**

**FROM Person p**

**where p.cust\_code in**

**( SELECT cust\_code from Transaction t**

**GROUP BY t.cust\_code**

**HAVING sum(qty) > mqty**

**);**

**BEGIN**

**FOR qty\_rec in c1**

**LOOP**

**DBMS\_OUTPUT.PUT\_LINE( qty\_rec.cname);**

**END LOOP;**

**END Quantity;**

**/\* 6.10.7\*/**

**BEGIN**

**Quantity(9);**

**END;**

**/\*6.10.8\*/**

**CREATE OR REPLACE PROCEDURE get\_fare1( P1 NUMBER)**

**AS**

**mcode Number;**

**mfare   Number;**

**BEGIN**

**SELECT pass\_code, fare INTO mcode,  mfare FROM PASSENG WHERE pass\_code = P1;**

**Dbms\_output.put\_line('mcode'||'    '|| mcode||'   '|| 'mfare'||'     '||mfare);**

**END get\_fare1;**

**/\*6.10.9\*/**

**BEGIN**

**get\_fare1(1115);**

**END;**

**/\*6.10.10\*/**

**CREATE  PROCEDURE DELETE\_CUST( TELNUMBER NUMBER)**

**AS**

**NO\_REC\_DELETED   NUMBER;**

**N1   Number;**

**N2   Number;**

**BEGIN**

**SELECT count(\*) into N1 from calls\_INFO;**

**DELETE FROM CALLS\_INFO**

**WHERE TEL\_NO = TELNUMBER;**

**SELECT  count(\*) into N2 FROM CALLS\_INFO;**

**NO\_REC\_DELETED := N1-N2;**

**Dbms\_output.put\_line('No of Records deleted' || NO\_REC\_DELETED);**

**END DELETE\_CUST;**

**/\* 6.10.11\*/**

**BEGIN**

**DELETE\_CUST (656445);**

**END;**