**/\*10.4.1\*/**

**DECLARE TYPE price IS table of NUMBER INDEX BY VARCHAR (20);**

**/\*Declaring type of index by Table and type is price\*/**

**pricelist   price;     --declaring index by table**

**iname  varchar(20);**

**BEGIN**

**/\*Adding values to index by table\*/**

**pricelist('INV1'):=1200;**

**pricelist('INV2'):=1500;**

**pricelist('INV3'):=1200;**

**pricelist('INV4'):=1800;**

**iname:=pricelist.first;**

**while iname is not NULL**

**LOOP**

**dbms\_output.put\_line('Price of'|| iname ||'   '|| pricelist(iname));**

**iname:=Pricelist.NEXT(iname) ;/\* next value\*/**

**END LOOP;**

**END;**

**/\*10.4.2\*/**

**DECLARE TYPE price IS table of number INDEX BY VARCHAR (20);**

**/\*Declaring type of index by Table\*/**

**pricelist   price;     --declaring index by table**

**iname  varchar(20);**

**BEGIN**

**/\*Adding values to index by table\*/**

**pricelist('INV1'):=1200;**

**pricelist('INV2'):=1500;**

**pricelist('INV3'):=1200;**

**pricelist('INV4'):=1800;**

**pricelist('INV2'):=1900;/\* modifying an existing value\*/**

**pricelist.delete('INV4');   /\* deleting a value from Index By Table\*/**

**iname:=pricelist.first;**

**while iname is not NULL**

**LOOP**

**Dbms\_output.put\_line('Price of'|| iname ||    'is'  ||  TO\_CHAR(pricelist(iname)));**

**iname:=Pricelist.NEXT(iname); /\* next value\*/**

**END LOOP;**

**END;**

**/\*10.4.3\*/**

**DECLARE TYPE price IS table of number INDEX BY VARCHAR (20);**

**/\*Declaring type of index by Table\*/**

**pricelist   price;     --declaring index by table**

**iname  varchar(20);**

**BEGIN**

**/\*Adding values to index by table\*/**

**pricelist('INV1'):=1200;**

**pricelist('INV2'):=1500;**

**pricelist('INV3'):=1200;**

**pricelist('INV4'):=1800;**

**/\* deleting 2 keys and their corresponding values from Index By Table\*/**

**pricelist.delete('INV2',’INV3’));**

**iname:=pricelist.first;**

**while iname is not NULL**

**LOOP**

**Dbms\_output.put\_line('Price of'|| iname ||    'is'  ||  TO\_CHAR(pricelist(iname)));**

**iname:=Pricelist.NEXT(iname); /\* next value\*/**

**END LOOP;**

**END;**

**/\* 10.4.4\*/**

**DECLARE**

**TYPE price IS table of NUMBER INDEX BY binary\_integer;**

**/\*Declaring type of index by Table and type is price\*/**

**pricelist price;**

**iname INTEGER;**

**BEGIN**

**/\*Adding values to index by table\*/**

**pricelist(1):=1200;**

**pricelist(3):=1500;**

**/\* This shows index need not be consecutive\*/**

**pricelist(5):=1200;**

**pricelist(7):=1800;**

**pricelist(8):=1800;**

**iname:=pricelist.first;**

**while iname is NOT NULL LOOP**

**dbms\_output.put\_line('Price of '|| iname ||' '|| pricelist(iname));**

**iname:=Pricelist.NEXT(iname);/\* next value\*/**

**END LOOP;**

**END;**

**/\*10.6.1\*/**

**DECLARE**

**TYPE names IS TABLE OF VARCHAR2(20);**

**empNames names;**

**counter INTEGER;**

**BEGIN**

**empNames:=names('David','Camillus', 'Hina','Ajay');--creating nested Table**

**empNames.EXTEND;**

**empNAMES(empNAmes.Last):='Priya';--adding a value to the last element**

**Counter := empNames.count;-- returns size**

**FOR I IN 1..counter LOOP**

**Dbms\_output.put\_line(‘Name: ‘|| empNames(i)|);**

**END LOOP;**

**END;**

**/\*10.6.2\*/**

**DECLARE**

**TYPE salary IS TABLE OF INTEGER;**

**empSalary salary;**

**counter INTEGER;**

**BEGIN**

**empSalary:=salary(80000,90000,94000,96000,120000);**

**--creating salary nested table**

**Counter := empSalary.count;-- returns size**

**FOR I IN 1..counter LOOP**

**Dbms\_output.put\_line(‘salary:’||empSalary(i));**

**END LOOP;**

**END;**

**/\*10.6.3\*/**

**DECLARE**

**TYPE names IS TABLE OF VARCHAR2(20);**

**TYPE salary IS TABLE OF INTEGER;**

**empSalary salary;**

**empNames names;**

**counter INTEGER;**

**BEGIN**

**empNames:=names('David','Camillus', 'Hina','Ajay');--creating nested Table**

**empNames.EXTEND;**

**empNAMES(empNAmes.Last):='Priya';--adding a value to the last element**

**empSalary:=salary(80000,90000,94000,96000,120000);**

**--creating salary nested table**

**Counter := empNames.count;-- returns size**

**FOR I IN 1..counter LOOP**

**Dbms\_output.put\_line(‘Name: ‘|| empNames(i)||’;salary:’||empSalary(i));**

**END LOOP;**

**END;**

**/\*10.6.4\*/**

**DECLARE**

**TYPE names IS TABLE OF VARCHAR2(20);**

**TYPE salary IS TABLE OF INTEGER;**

**empSalary salary;**

**empNames names;**

**counter INTEGER;**

**BEGIN**

**empNames:=names('David','Camillus', 'Hina','Ajay');--creating nested Table**

**empNames.EXTEND;**

**empNAMES(empNAmes.Last):='Priya';--adding a value to the last element**

**empSalary:=salary(80000,90000,94000,96000,120000);**

**--creating salary nested table**

**Counter := empNames.count;-- returns size**

**FOR i IN 1 .. counter**

**LOOP**

**IF empSalary(i) = 94000 THEN**

**empSalary.DELETE(i);**

**empNames.DELETE(i);**

**END IF;**

**END LOOP;**

**FOR i IN 1 .. counter**

**LOOP**

**Dbms\_output.put\_line(' Name '||'   '||empNames(i)||'   '||'Salary '||'   '|| empSalary(i));**

**END LOOP;**

**END;**

**/\*10.6.5\*/**

**DECLARE**

**TYPE names IS TABLE OF VARCHAR2(20);**

**empNames names;**

**counter INTEGER;**

**BEGIN**

**empNames:=names('David','Camillus', 'Hina','Ajay', 'priya');**

**Counter := empNames.count;-- returns size**

**empNames.DELETE(2,4);**

**FOR i IN 1 .. counter**

**LOOP**

**IF empNames.EXISTS(i) = TRUE**

**THEN**

**Counter := empNames.count;-- returns size**

**Dbms\_output.put\_line(‘Name: ‘|| empNames(i));**

**ELSE**

**Dbms\_output.put\_line('No value found for record number '|| '  '||  i);**

**CONTINUE;**

**END IF;**

**END LOOP;**

**END;**

/**\*10.6.6\*/**

**DECLARE  
   TYPE names IS TABLE OF VARCHAR2(20);  
   inventory names;  
   copy1 names;  
   copy2 names;  
     
BEGIN  
 inventory := names('INV1', 'INV2', 'INV3', 'INV4');  
 copy1 := inventory;  
 copy2 := inventory;  
 IF copy1 = inventory**

**THEN  
    Dbms\_output.put\_line('Same Table');  
 ELSE  
    Dbms\_output.put\_line('Different Table');  
 END IF;  
 copy2.EXTEND;**

**IF copy2 = inventory**

**THEN  
    Dbms\_output.put\_line('Same Table');  
 ELSE  
    Dbms\_output.put\_line('Different Table');  
 END IF;  
END;**

**/\*10.8.1\*/**

**DECLARE**

**TYPE nameListType IS VARRAY(6) OF VARCHAR2(20) NOT NULL;**

**nameList nameListType;**

**BEGIN**

**nameList := nameListType('David','Camillus', 'Hina');**

**dbms\_output.put\_line('The number of elements in the Table:'||’ ‘|| nameList.COUNT);**

**END;**

**/\*10.8.2\*/**

**DECLARE  
    TYPE items IS VARRAY(4) OF VARCHAR(20);  
    inventoryItems items;  
BEGIN  
    inventoryItems := items('Egg', 'Bread','Biscuit');  
    dbms\_output.put\_line('Size of VARRAY = ' || inventoryItems.LIMIT);  
    dbms\_output.put\_line('Number of items in the  list = ' || inventoryItems.COUNT);  
     
    inventoryItems.EXTEND;  
    inventoryItems(inventoryItems.LAST) := 'Butter';  
    dbms\_output.put\_line('Size of VARRAY = ' || inventoryItems.LIMIT);  
    dbms\_output.put\_line('Number of items in the  list = ' || inventoryItems.COUNT);  
     
    FOR i in 1..inventoryItems.COUNT LOOP  
         dbms\_output.put\_line(inventoryItems(i));  
    END LOOP;  
--inventoryItems.EXTEND;**

**END;**

**/\* 10.8.3\*/**

**/\*Creating a 2D matrix \*/**

**DECLARE**

**TYPE c is VARRAY(3) of NUMBER(2);**

**TYPE matrix is VARRAY(3) of  c;**

**M matrix;**

**BEGIN**

**M := matrix( c(1,2,3), c(4,5,6),c(7,8,9) );**

**FOR x in 1..M.COUNT**

**LOOP**

**FOR y in 1..M(x).COUNT**

**LOOP**

**DBMS\_OUTPUT.PUT(M(x)(y));**

**DBMS\_OUTPUT.PUT('   ');**

**END LOOP;**

**DBMS\_OUTPUT.PUT\_LINE('   ');**

**END LOOP;**

**END;**

**/\* 10.9.1\*/**

**DECLARE**

**TYPE Courses IS TABLE of NUMBER;**

**TYPE Subject IS TABLE of Courses INDEX BY VARCHAR(20);**

**Marks Subject;**

**name VARCHAR(20) := :name;**

**BEGIN**

**Marks('David'):= Courses(93,94,95);**

**Marks('John'):= Courses(63,65,76);**

**Marks('Hina'):= Courses(92,91,90);**

**IF Marks.EXISTS(name)**

**THEN**

**dbms\_output.put\_line('Marks of '||' '|| name);**

**dbms\_output.put\_line('Number of Subjects Taken: ' || Marks(name).COUNT);**

**FOR i in 1..Marks(name).COUNT**

**LOOP**

**dbms\_output.put\_line('Subject Number '||'  '||i || ' Marks: ' || Marks(name)(i));**

**END LOOP;**

**ELSE**

**dbms\_output.put\_line('No records found for' ||’ ‘|| name);**

**END IF;**

**END;**

**/\*10.10.1\*/**

**DECLARE  
 TYPE studentRecord IS TABLE of NUMBER INDEX BY VARCHAR(30);  
 TYPE SubjectRecord IS TABLE of studentRecord INDEX BY VARCHAR(30);  
 Marks SubjectRecord;  
 name VARCHAR(30) := :name;  
 subject VARCHAR(30) := :subject;  
BEGIN  
 Marks('David')('Machine Learning'):=80;  
 Marks('David')('Linear Algebra'):=86;  
 Marks('David')('Data Structures'):=84;  
 Marks('John')('Machine Learning'):=95;  
 Marks('John')('Linear Algebra'):=96;  
 Marks('John')('Probability and Stats'):=89;  
 Marks('Hina')('Machine Learning'):=85;  
 Marks('Hina')('Linear Algebra'):=97;  
 Marks('Hina')('Computer Vision'):=92;  
 IF Marks.EXISTS(name) THEN**

**IF Marks(name).EXISTS(subject) THEN  
        dbms\_output.put\_line('Marks for '|| name || ' IN ' || subject || ': ' || TO\_CHAR(Marks(name)(subject)));  
    ELSE  
        dbms\_output.put\_line('No subject found');  
    END IF;  
 ELSE  
    dbms\_output.put\_line('No records found for ' || name);  
 END IF;  
END;**

**/\* 10.10.2\*/**

**DECLARE**

**TYPE price IS TABLE of NUMBER INDEX BY VARCHAR(30);**

**TYPE PriceRecord IS TABLE of price INDEX BY VARCHAR(30);**

**Pricelist    PriceRecord;**

**iname VARCHAR(30) := :iname;**

**color  VARCHAR(30) := :color;**

**BEGIN**

**Pricelist('INV1')('Red'):=1200;**

**Pricelist('INV1')('green'):=1100;**

**Pricelist('INV2') ('red'):=1500;**

**Pricelist('INV2') ('blue'):=1400;**

**Pricelist('INV2') ('yellow'):=1200;**

**Pricelist ('INV3')('red' ):=1200;**

**Pricelist ('INV3')('orange' ):=1400;**

**IF Pricelist.EXISTS(iname) THEN**

**IF Pricelist(iname).EXISTS(color) THEN**

**dbms\_output.put\_line('Price for '|| iname || ' FOR ' || color || ': ' || Pricelist(iname)(color));**

**ELSE**

**dbms\_output.put\_line('No Inventory  with that color  found');**

**END IF;**

**ELSE**

**Dbms\_output.put\_line('No records found for ' || iname);**

**END IF;**

**END;**

**/\*10.11.1\*/**

**DECLARE**

**CURSOR F1 IS SELECT \* FROM FLIGHT;**

**TYPE flightList IS TABLE OF F1%ROWTYPE;**

**flights\_Delhi flightList;**

**BEGIN**

**flights\_Delhi := flightList();**

**/\* nested table is initialized with a constructor\*/**

**FOR REC IN F1**

**LOOP**

**IF REC.SOURCE = 'Delhi' THEN**

**flights\_Delhi.EXTEND;**

**flights\_Delhi(flights\_Delhi.LAST) := REC;**

**END IF;**

**END LOOP;**

**Dbms\_output.put\_line('Details of flights starting from Delhi');**

**Dbms\_output.put\_line('Flight'||CHR(9)||'Source'||CHR(9)||'Destination');**

**FOR i in 1..flights\_Delhi.COUNT**

**LOOP**

**Dbms\_output.put\_line ( flights\_Delhi(i).FLIGHT\_NO || CHR(9) ||**

**flights\_Delhi(i).SOURCE || CHR(9) ||**

**flights\_Delhi(i).DESTIN);**

**END LOOP;**

**END;**

**/\*10.11.2\*/**

**DECLARE**

**CURSOR F1 IS SELECT \* FROM FLIGHT;**

**TYPE flightList IS TABLE OF F1%ROWTYPE INDEX BY binary\_integer;**

**flights\_Delhi flightList;**

**FLTNO binary\_integer;**

**BEGIN**

**FOR REC IN F1**

**LOOP**

**IF REC.SOURCE = 'Delhi'**

**THEN**

**flights\_Delhi(REC.FLIGHT\_NO) := REC;**

**END IF;**

**END LOOP;**

**Dbms\_output.put\_line('Flight'||CHR(9)||'Source'||CHR(9)||'Destination');**

**FLTNO := flights\_Delhi.FIRST;**

**WHILE FLTNO IS NOT NULL**

**LOOP**

**Dbms\_output.put\_line ( flights\_Delhi(FLTNO).FLIGHT\_NO ||CHR(9) ||**

**flights\_Delhi(FLTNO).SOURCE ||CHR(9) ||**

**flights\_Delhi(FLTNO).DESTIN);**

**FLTNO := flights\_Delhi.NEXT(FLTNO);**

**END LOOP;**

**END;**