Advanced Database

**Lab Sheet**

# OBJECT TYPES & Collection Types

OBJECTS

1. Create an Object type called O\_ADDRESS that has attributes address\_line1, address\_line2, address\_line3, address\_line4, city, country.
2. Create an Object type called O\_SUBJECT that has attributes subject\_code and subject\_name.

***Creating Collections VArrays and Nested Tables***

1. Create 2 varying arrays
   * One to hold up to 6 phone numbers
   * One to hold up to 16 College Class object ( a college class Object is made up of College Class Name (e.g. ADB, OS), Class Type (e.g. Lab, Lecture), Day, Time). For our purposes treat day and time as strings.
2. Create a nested table type called SUBJECT\_NTABLE that holds a collectionof subjects
3. Create a table STUDENT that contains a
   * student\_id INTEGER
   * full\_name VARCHAR2
   * phone\_nos Varying Array
   * classes Varying Array
   * subjects\_registered SUBJECT\_NTABLE
   * home\_address O\_ADDRESS
   * next\_of\_kin VARCHAR2
   * nok\_address O\_ADDRESS
4. Populate the STUDENT with some data ( 3 rows).
5. Query the phone number varray in the student table as follows

SELECT phone\_no FROM student

WHERE student\_id =1001;

1. Try using the same query except using TABLE keyword and note the output

SELECT p.\* FROM student s, TABLE(s.phone\_no) p

WHERE student\_id =1001;

SELECT p.COLUMN\_VALUE FROM student s, TABLE(s.phone\_no) p

WHERE student\_id =1001;

**A WORD ON HOW TO QUERY VARRAY AND NESTED TABLES**

For demonstration purposes, below is a table containing a VARRAY of objects. But I don't know how to find which divisions have an employees called John working in them. How could I look each value of the VARRAY: if the first value is not John, look at the next and so on until I get the value desired, or it realizes that the value doesn't exist?

CREATE OR REPLACE TYPE emp\_obj

AS OBJECT

(

last\_name VARCHAR (50),

first\_name VARCHAR (50),

birth\_dt DATE

);

/

CREATE OR REPLACE TYPE emp\_va

AS VARRAY (999)

OF emp\_obj;

/

CREATE TABLE division

(

division\_id NUMBER (9) PRIMARY KEY,

division\_name VARCHAR2 (100),

emps emp\_va

);

We know how to get all the items in my VARRAY with a query like:

SELECT e.first\_name

FROM division d

, TABLE (d.emps) e

WHERE d.division\_name = 'Sales';

You can use single elements from the VARRAY in the WHERE clause in

SELECT DISTINCT

d.division\_name

FROM division d

, TABLE (d.emps) e

**WHERE e.first\_name = 'John'**;

Or another way

SELECT division\_name

FROM division d

WHERE EXISTS

( SELECT 'X' -- Why 'X'? See note below.

FROM TABLE (d.emps)

WHERE first\_name = 'John'

);

**Why 'X'?** You could SELECT anything in the subquery: a literal (like I did) a column, SYSDATE, USER or a list of any kind(s) of values. The only crucial thing is whether or not the subquery returns any rows. SELECTing a literal is faster than SELECTing a real column or SYSDATE.

1. Now display the student id and fullname from students that attend **adb** class on **Friday** (change the values to suit your data)

10. What students are registered for subjects sdev or adb (change the values to suit your data and use the nested table column). Only one row should be displayed for each student registered.