Suggested Solutions

1. Create an Object type called MO_PERSON that has attributes **name**, and **email address**.

```
CREATE or REPLACE TYPE MO_PERSON AS OBJECT
(name  VARCHAR2(30),
 email_address VARCHAR2(30)
);
```

2. Create an Object type called MO_CONTACT that has attributes **MO_PERSON** , and **phone_number**.

```
CREATE or REPLACE TYPE MO_CONTACT AS OBJECT
(name MO_PERSON,
 phone_number  VARCHAR2(20)
);
```

3. Create an Object Table called myContacts based on MO_Contact that contains contacts with the Primary Key of email address and insert an object. Note, the person Object's name attribute is mandatory and the phone_number is optional.

```
CREATE TABLE mycontacts OF MO_CONTACT
(CONSTRAINT pk1_customer_names PRIMARY KEY(name.email_address),
CONSTRAINT name_mand_con CHECK(name.name IS NOT NULL));


INSERT INTO mycontacts VALUES
(MO_CONTACT(MO_PERSON('john smith','sj.smith@itnet.ie'), '40423432'));

INSERT INTO mycontacts VALUES
(MO_CONTACT(MO_PERSON('jim jones','jim.jones@itnet.ie'), null));

INSERT INTO mycontacts VALUES
(MO_CONTACT(MO_PERSON('mary hayes','mhayes@itnet.ie'), '40553432'));

INSERT INTO mycontacts VALUES
(MO_CONTACT(MO_PERSON('june rogers','j.rogers@itnet.ie'), null));

INSERT INTO mycontacts VALUES
(MO_CONTACT(MO_PERSON('marian keyes','marian,keyes@itnet.ie'), '40676732'));
Commit;

SELECT mc.name.email_address, mc.phone_number
FROM mycontacts mc
WHERE mc.name.name ='mary hayes';
```

4. Create a table called MO_CUSTOMER that contains a CUST_ID integer (Primary Key), mandatory columns, CONTACT_DETAILS (OF TYPE MO_contact AND IS MANDATORY), COMPANY NAME, CREDIT_LIMIT must be between €1600 and €10,000 and the constraint name to be given is CUSTOMER_CREDIT_CHECK . A MARKET column must be either national or international and the constraint must be named CUSTOMER_MARKET_CHECK.

```
CREATE TABLE CUSTOMER_TAB(
CUST_ID          INTEGER    PRIMARY KEY,
CONTACT          MO_CONTACT NOT NULL,
COMPANY_NAME     VARCHAR2(30) NOT NULL,
CREDIT_LIMIT     NUMBER,
MARKET           VARCHAR2(15),
CONSTRAINT CUSTOMER_CREDIT_CHECK
CHECK(CREDIT_LIMIT BETWEEN 1600 AND 10000),
CONSTRAINT CUSTOMER_MARKET_CHECK CHECK(MARKET
IN('NATIONAL','INTERNATIONAL'))
);
```

5. Populate the MO_CUSTOMER TABLE with records. Then undo your INSERTS

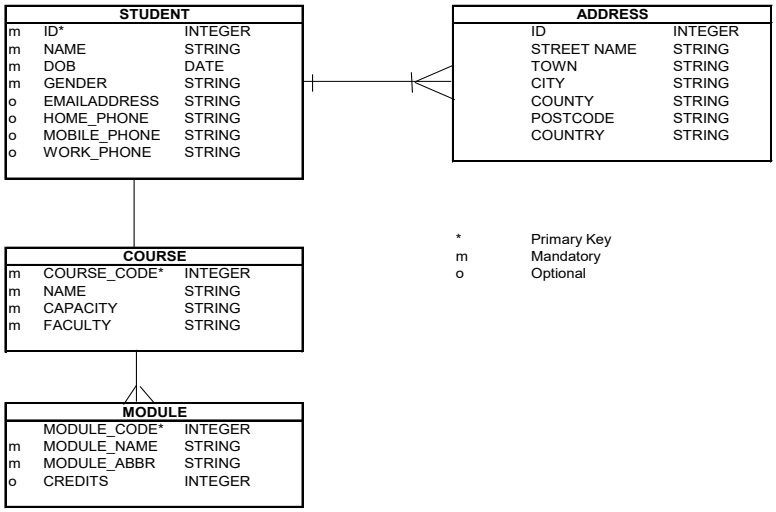
```
INSERT INTO CUSTOMER_TAB VALUES ( 1,
MO_CONTACT(MO_PERSON('johnSmith','sj.smith@itnet.ie'),
          '40423432'),
'Widgets LTD.',
 7000,
'NATIONAL');


ROLLBACK;
```

PART B Object Relation Databases

The database designer has been looking at Oracle's Object relational features and has decided to convert relational structure below with the following requirements

- Student Table with an appropriate VARRAY collection type for phone columns. Create appropriate Phone object type
- Addresses for student will be implemented as a nested table in the student table
- Student Table will reference a Course Object in an Object table for courses using REF.
- An object type Module with an appropriate object table will reference the course objects table using REF. Note how the 1:m relationship is now implemented



PART B

```
--PART B

-- Create phone type and varray.

CREATE OR REPLACE TYPE PHONE_TYPE_o AS OBJECT
(PHONE_CONTACT_TYPE VARCHAR2(20),
 PHONE_NUMBER   VARCHAR2(12));

CREATE OR REPLACE TYPE PHONE_LIST AS VARRAY(3) OF
PHONE_TYPE_o;


--create the address type that will be a nested
table in the students table
CREATE TYPE mo_address AS OBJECT
(ID        INTEGER,
STREET_NAME      VARCHAR2(30),
TOWN             VARCHAR2(20),
CITY             VARCHAR2(20),
COUNTY           VARCHAR2(30),
POSTCODE         VARCHAR2(10),
COUNTRY          VARCHAR2(20)
```

```
);


CREATE OR REPLACE TYPE ADDRESS_NTABLE AS TABLE OF
mo_address;


--create an object type for course and an object
table to course objects

CREATE OR REPLACE TYPE MO_COURSE AS OBJECT(
COURSE_CODE      INTEGER,
NAME             VARCHAR2(30),
CAPACITY         INTEGER,
FACULTY          VARCHAR2(30)
);

CREATE TABLE course_OBJTABLE of MO_COURSE (
COURSE_CODE      PRIMARY KEY,
NAME             NOT NULL,
CAPACITY         NOT NULL,
FACULTY          NOT NULL
);

--OBJ1 file
-- create a Module object type and that will have a
--REF to a course object
-- the module object table is scoped to ensure the
course object it points to is in the course
--object table course_OBJTABLE
CREATE TYPE MO_MODULE AS OBJECT(
MODULE_CODE      INTEGER,
MODULE_NAME      VARCHAR2(30),
MODULE_ABBR      VARCHAR2(30),
CREDITS          VARCHAR2(30),
COURSE           REF MO_COURSE
);



CREATE TABLE module_OBJTABLE of MO_COURSE (
COURSE           SCOPE IS  course_objtable,
MODULE_CODE     PRIMARY KEY,
MODULE_NAME      NOT NULL,
MODULE_ABBR      NOT NULL);



--Create a table student as follows

CREATE TABLE STUDENT_UNDERGRAD
(ID              INTEGER          PRIMARY KEY,
NAME             VARCHAR2(30)     NOT NULL,
DOB              DATE             NOT NULL,
GENDER           VARCHAR2(1)      NOT NULL,
EMAILADDRESS     VARCHAR2(15)     NOT NULL,
PHONE_NUMBERS    PHONE_LIST       NOT NULL,
ADDRESSES        ADDRESS_NTABLE,
COURSE REF MO_COURSE REFERENCES course_objtable
)NESTED TABLE ADDRESSES STORE AS addDetailsntable;
```

```
--INSERTS
--INSERTS
INSERT INTO course_OBJTABLE VALUES(
 MO_COURSE (1234,
            'BSc in ITMGT',
             60,
             'Computing'  )
    );



INSERT INTO module_OBJTABLE VALUES(
 MO_MODULE (1111,
           'Big Data Technologies',
           'BDT',
           5,
           (SELECT REF(c) FROM course_OBJTABLE c
            WHERE  c.course_code =1234)
         )
        );
```

```
INSERT INTO STUDENT_UNDERGRAD VALUES(
  1,
  'Sean McHugh',
  '23-oct-2000',
  'm',
  'smchugh@tud.ie',
  phone_list(phone_type_o('home','0871231123'),
             phone_type_o('college','0875444123')),
  ADDRESS_NTABLE(mo_address (1,
                             'Main St',
                             'Tallaght',
                             'Dublin',
                             'Dublin',
                             '24',
                             'Ireland'),
                 mo_address (2,
                             'Upper Road',
                             'Blanchardstown',
                              'Dublin',
                              'Dublin',
```

```
                                '15',
                                'Ireland')),
  (SELECT REF(c)
   FROM course_OBJTABLE c
   WHERE  c.course_code =1234)
  );



--Query 1
SELECT st.name,st.course.name,st.course.faculty
FROM STUDENT_UNDERGRAD st
WHERE st.name ='Sean McHugh';

--Query 2
SELECT st.name,p.phone_number, a.street_name
FROM STUDENT_UNDERGRAD st,
     TABLE(Phone_Numbers) p,
     TABLE(addresses) a
WHERE st.name ='Sean McHugh';
```