Object-relational databases

Exercises

EXERCISE 1

Create the object type ***Member*** with the following attributes:

```
idMember    CHAR(4),
fullname    VARCHAR2(100),
username    VARCHAR2(15),
phone       VARCHAR2(15),
email       VARCHAR2(100),
dateEntry   DATE
```

Create, the object type ***Teacher*** as an inherited type of *Member* with the following attributes:

```
title     VARCHAR2(100),
salary    NUMBER(8,2),
category  VARCHAR2(20)
```

Create, the object type ***Groups*** with the following attributes:

```
groupId       INTEGER,
code          CHAR(3),
name          VARCHAR2(50),
tutor         Teacher,
level         INTEGER,
morning       CHAR(1),
requirements  VARCHAR(100)
```

Create the object type ***Student*** as an inherited type of *Member* with the following attributes:

```
enrolid       INTEGER,
birthdate     DATE,
groupId       REF Groups,
gender        CHAR(1),
nationality   VARCHAR2(15)
```

EXERCISE 2

Create a constructor method for the object type ***Teacher*** with the following parameters:

- idMember,
- fullname,
- username,
- phone,
- email
- title
- salary

The method must:

- assign a default value of *Secondary School* for **category** and current date for **dateEntry**

EXERCISE 3

Create a "getTeacher" method for the ***Teacher*** object type that allows you to obtain the member data with the following format:
`"fullname (title) / category"`

For example, for a ***Teacher*** the method should show:

Juliette Smith (Bachelor in Computer Science) / Secondary School

EXERCISE 4

Create a VARRAY collection called ***TeacherList*** to store up to 15 ***Teacher*** objects. Later, create a ***TeacherList*** instance called ***TeacherList_1*** with the data of the following two teachers:

- The first teacher to insert in the list is:

  `idMember` T001,
  `fullname` Juliette Smith,
  `username` juli,
  `phone` 666123456,
  `email` juliette.smith@pau.cat
  `dateEntry` 01/04/2023
  `title` Bachelor in Computer Science
  `salary` 60000,
  `category` University Faculty

- The second teacher (using the constructor method done in the exercise 3) to insert in the list is:

  `idMember` T002
  `fullname` Nicole Brown
  `username` nico
  `phone` 666111222
  `email` nicole.brown@pau.cat
  `title` Data Science Master
  `salary` 50000

EXERCISE 5

Create a ***Groups_tab*** table of ***Groups*** objects. Insert the following rows into this table:

`groupId:` 10
`code`: S1W
`name`: Desenvolupament d'Aplicacions Web
`tutor`: object created in the exercise 4 for the second teacher
`level:` 1
`morning:` Y
`requirements:` Middle school, high school, university

`groupId`: 20
`code`: S2P

`name`: Desenvolupament d'Aplicacions Multiplataforma
`tutor`: object created in the exercise 4 in the first position of the list (it must be taken of the list)
`level`: 2
`morning`: N
`requirements`: S1P

EXERCISE 6

Create a ***`Students`*** table of ***`Student`*** objects. Insert two ***`Student`*** objects into this table. The first one with the data:

`idMember`	S001
`fullname`	Alice in Wonderland
`username`	ally
`phone`	665663311
`email`	`alice.wonder@pau.cat`
`dateEntry`	14/04/2023
`enrolid`	1
`birthdate`	01/01/1990
`groupId`	Reference to the group with `groupId` 10
`gender`	F
`nationality`	Wonderland

The second one with the data::

`idMember`	S002
`fullname`	George Jobs
`username`	geor
`phone`	665662222
`email`	`george.jobs@pau.cat`
`dateEntry`	14/04/2023
`enrolid`	2
`birthdate`	23/04/1989
`groupId`	Reference to the group with `groupId` 20
`gender`	M
`nationality`	English

EXERCISE 7

Obtain the group with `groupId` 20 from the ***`Groups_tab`*** table and assign it to a variable called ***`oneGroup`***. Later, modify the *`groupId`* attribute saved in the variable ***`oneGroup`*** and assign it the value 30, also modify the `code` attribute and assign it the value `S1P`, and its tutor must be the second tutor that had been previously created in the exercise 4. Finally, insert that group in the ***`Groups_tab`*** table.

Scoring criteria. Total 10 points.

EXERCISE 1: 1 Point
EXERCISE 2: 1 Point
EXERCISE 3: 1 Point

EXERCISE 4: 1 Point
EXERCISE 5: 2 Point
EXERCISE 6: 2 Point
EXERCISE 7: 2 Point

Resources needed to carry out the Task.

- Oracle Live SQL

Recommendations

It is recommended to do the exercises in the order in which they are indicated because to solve some of them it could be necessary to have done a previous one.

DELIVER

You must deliver a pdf file with:

- Screenshot of every exercise showing its execution on Oracle Live SQL.
- One script of every exercise. Download them from Oracle Live sql.