

Nota: 4

# Bases de Datos

Tarea para BD07

## Tarea para BD07.

Enunciado.

1Juan Morillo

Desarrollo de Aplicaciones Multiplataforma (DAM)

1

Realiza los siguientes ejercicios. Antes de comenzar lee con detalle el enunciado de la tarea íntegramente y consulta con tu tutor las dudas que te puedan surgir.

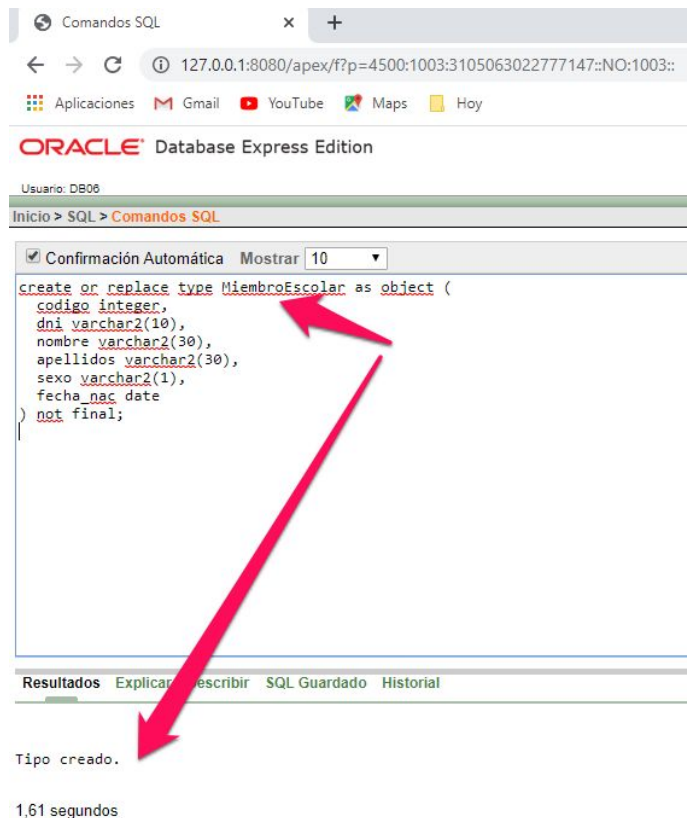
1. Crea el tipo de objetos "**MiembroEscolar**" con los siguientes atributos:

```
codigo INTEGER,  
dni VARCHAR2(10),  
nombre VARCHAR2(30),  
apellidos VARCHAR2(30),  
sexo VARCHAR2(1),  
fecha_nac DATE
```

### **Solucion:**

```
create or replace type MiembroEscolar as object (  
  codigo integer,  
  dni varchar2(10),  
  nombre varchar2(30),  
  apellidos varchar2(30),  
  sexo varchar2(1),  
  fecha_nac date  
) not final;
```

/

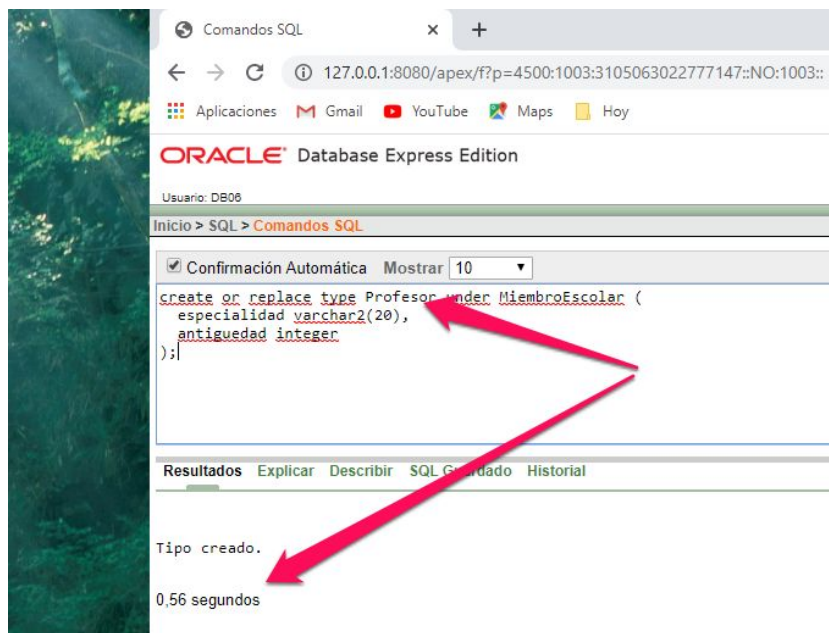


Crea, como tipo heredado de "MiembroEscolar", el tipo de objeto "Profesor" con los siguientes atributos:

```
especialidad VARCHAR2(20),  
antiguedad INTEGER
```

### Solucion:

```
create or replace type Profesor under MiembroEscolar (  
  
  especialidad varchar2(20),  
  antiguedad integer  
);
```

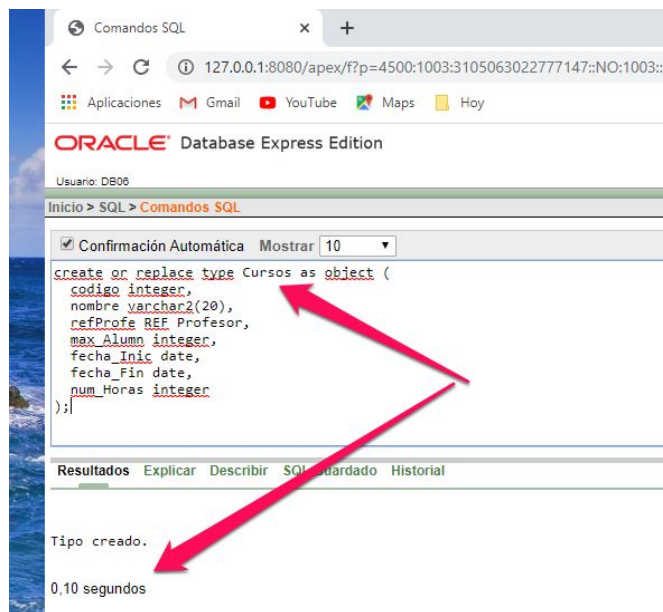


Crea el tipo de objeto "Cursos" con los siguientes atributos:

```
codigo INTEGER,  
nombre VARCHAR2(20),  
refProfe REF Profesor,  
max_Alumn INTEGER,  
fecha_Inic DATE,  
fecha_Fin DATE,  
num_Horas INTEGER
```

### Solución:

```
create or replace type Cursos as object (  
  codigo integer,  
  nombre varchar2(20),  
  refProfe ref Profesor,  
  max_Alumn integer,  
  fecha_Inic date,  
  fecha_Fin date,  
  num_Horas integer  
);
```

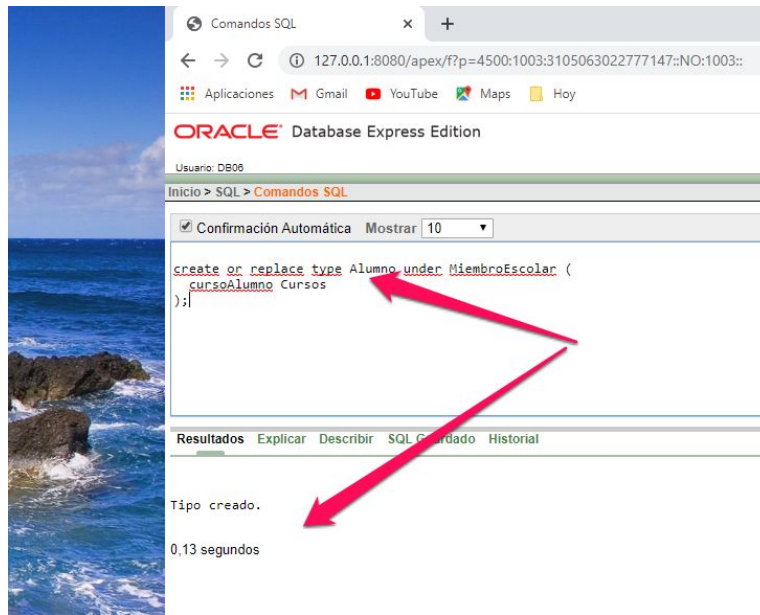


Crea, como tipo heredado de "MiembroEscolar", el tipo de objeto "Alumno" con los siguientes atributos:

cursoAlumno Cursos

### Solución:

```
create or replace type Alumno under MiembroEscolar (  
  cursoAlumno Cursos  
);
```



0,75

2. Crea un método constructor para el tipo de objetos "Profesor", en el que se indiquen como parámetros el código, nombre, primer apellido, segundo apellido y especialidad. Este método debe asignar al atributo "apellidos" los datos de primer apellido y segundo apellido que se han pasado como parámetros, uniéndolos con un espacio entre ellos.

### Solución:

create or replace type body Profesor as

constructor function Profesor (

codigo integer,

nombre varchar2,

primerApellido varchar2,

segundoApellido varchar2,

especialidad varchar2

return self as result

is

begin

self.codigo := codigo;

self.nombre := nombre;

self.apellidos:=concatprimerApellido, segundoApellido);

self.especialidad := especialidad;

return;

end;

end;

);

Falta antigüedad

Comandos SQL

127.0.0.1:8080/apex/f?p=4500:1003:3105063022777147::NO:1003::

Aplicaciones Gmail YouTube Maps Hoy

**ORACLE** Database Express Edition

Usuario: DB06

Inicio > SQL > Comandos SQL

☒ Confirmación Automática Mostrar 10

```
create or replace type body Profesor as
  constructor function Profesor (
    codigo integer,
    nombre varchar2,
    primerApellido varchar2,
    segundoApellido varchar2,
    especialidad varchar2
  ) return self as result
is
begin
  self.codigo := codigo;
  self.nombre := nombre;
  self.apellidos:=concat(primerApellido, segundoApellido);
  self.especialidad := especialidad;
  return;
end;
end;
```

Resultados Explicar Describir SQL Guardado Historial

Tipo creado.

0,31 segundos

3. Crea un método "getNombreCompleto" para el tipo de objetos "Profesor" que permita obtener su nombre completo con el formato "apellidos nombre".

**Solución:**

```
create or replace type body Profesor as
  constructor function Profesor (
    codigo integer,
    nombre varchar2,
    primerApellido varchar2,
    segundoApellido varchar2,
    especialidad varchar2
  ) return self as result
  is
  begin
    self.codigo := codigo;
    self.nombre := nombre;
    self.apellidos:=concatprimerApellido, segundoApellido);
    self.especialidad := especialidad;
    return;
  end;
end;
);

function getNombreCompleto return varchar2
  is
  begin
    return (apellidos | ' ' | nombre);
  end getNombreCompleto;
end;
```

error

1



127.0.0.1:8080/apex/f?p=4500:1003:83558484592872::NO:1003::

Aplicaciones

Gmail

YouTube

Maps

Hoy

ORACLE® Database Express Edition

Usuario: DB06

Inicio > SQL > Comandos SQL

☒ Confirmación Automática

Mostrar 10

```
create or replace type body Profesor as
  constructor function Profesor (
    codigo integer,
    nombre varchar2,
    primerApellido varchar2,
    segundoApellido varchar2,
    especialidad varchar2
  ) return self as result
is
begin
  self.codigo := codigo;
  self.nombre := nombre;
  self.apellidos:=concat(primerApellido, segundoApellido);
  self.especialidad := especialidad;
  return;
end;
end;
);

function getNombreCompleto return varchar2
is
begin
  return (apellidos | ' ' | nombre);
end getNombreCompleto;
end;
```

Resultados

Explicar

Describir

SQL Guardado

Historial

Tipo creado.

0,02 segundos

4. Crea un tabla "Profesorado" de objetos "Profesor". Inserta en dicha tabla dos objetos "Profesor". El primero de ellos con los datos:

codigo: 2

9Juan Morillo  
Desarrollo de Aplicaciones Multiplataforma (DAM)

```
dni: 51083099F
nombre: MARIA LUISA
apellidos: FABRE BERDUN
sexo: F
fecha_nac: 31/03/1975
especialidad: TECNOLOGIA
antiguedad: 4
```

El segundo objeto "Profesor" debes crearlo usando el método constructor que has realizado anteriormente. Debes usar los siguientes datos:

```
codigo: 3
nombre: JAVIER
apellidos: JIMENEZ HERNANDO
especialidad: LENGUA

antiguedad: 1
```

### Solución:

```
create table Profesorado of profesor;
```

```
declare
```

```
profesor1 Profesor;
```

```
profesor2 Profesor;
```

```
begin
```

```
insert into Profesorado values (2,'51083099F','MARIA LUISA','FABRE
BERDUN','F','31/03/1975','TECNOLOGIA',4);
```

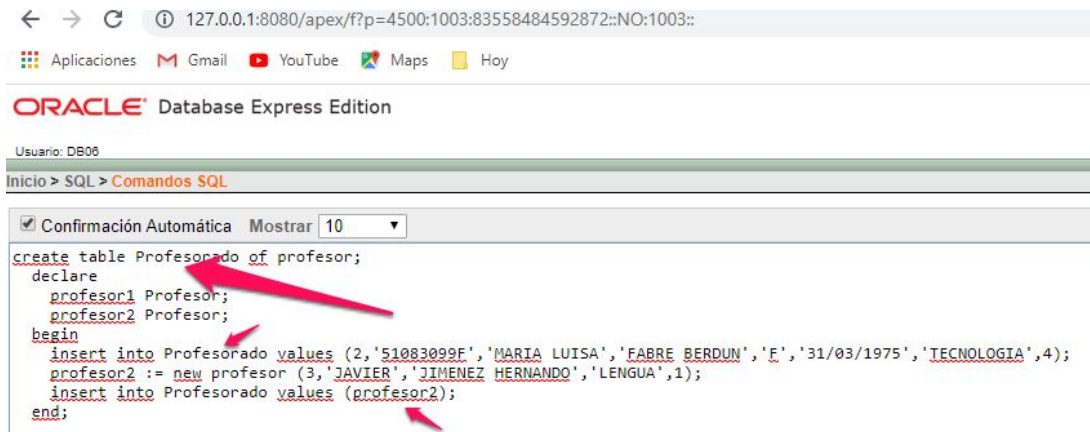
```
profesor2 := new profesor (3,'JAVIER','JIMENEZ HERNANDO','LENGUA',1);
```

```
insert into Profesorado values (profesor2);
```

```
end;
```

ES una tabla de profesores, debes insertar  
objetos PROFESOR usando los constructores

0,5



```
← → ↻ 127.0.0.1:8080/apex/f?p=4500:1003:83558484592872::NO:1003::
Aplicaciones Gmail YouTube Maps Hoy

ORACLE Database Express Edition

Usuario: DB06
Inicio > SQL > Comandos SQL

☒ Confirmación Automática Mostrar 10 ▼

create table Profesorado of profesor;
declare
  profesor1 Profesor;
  profesor2 Profesor;
begin
  insert into Profesorado values (2,'51083099F','MARIA LUISA','FABRE BERDUN','F','31/03/1975','TECNOLOGIA',4);
  profesor2 := new profesor (3,'JAVIER','JIMENEZ HERNANDO','LENGUA',1);
  insert into Profesorado values (profesor2);
end;
```

0

5. Crea una colección **VARRAY** llamada "**ListaCursos**" en la que se puedan almacenar hasta 10 objetos "**Cursos**". Guarda en una instancia "**listaCursos1**" de dicha lista, los dos cursos siguientes:

El primer curso que debes almacenar en dicha lista debe tener los siguientes datos:

```
codigo: 1
nombre: Curso 1
refProfe: Referencia al profesor cuyo codigo es 3.
max_Alumn: 20
fecha_Inic: 1/6/2011
fecha_Fin: 30/6/2011
num_Horas: 30
```

El segundo curso que debes almacenar en dicha lista debe tener los siguientes datos:

```
codigo: 2
nombre: Curso 2
refProfe: Referencia al profesor cuyo DNI es 51083099F.
max_Alumn: 20
fecha_Inic: 1/6/2011
fecha_Fin: 30/6/2011
num_Horas: 30
```

**Solución:**

```
create or replace type ListaCursos is varray(10) of Cursos;
```

```
declare
```

```
  C1 Cursos;
```

```
  C2 Cursos;
```

```
  listaCursos1 ListaCursos := ListaCursos('c1','c2','c3','c4',  
    'c5','c6','c7','c8','c9','c10');
```

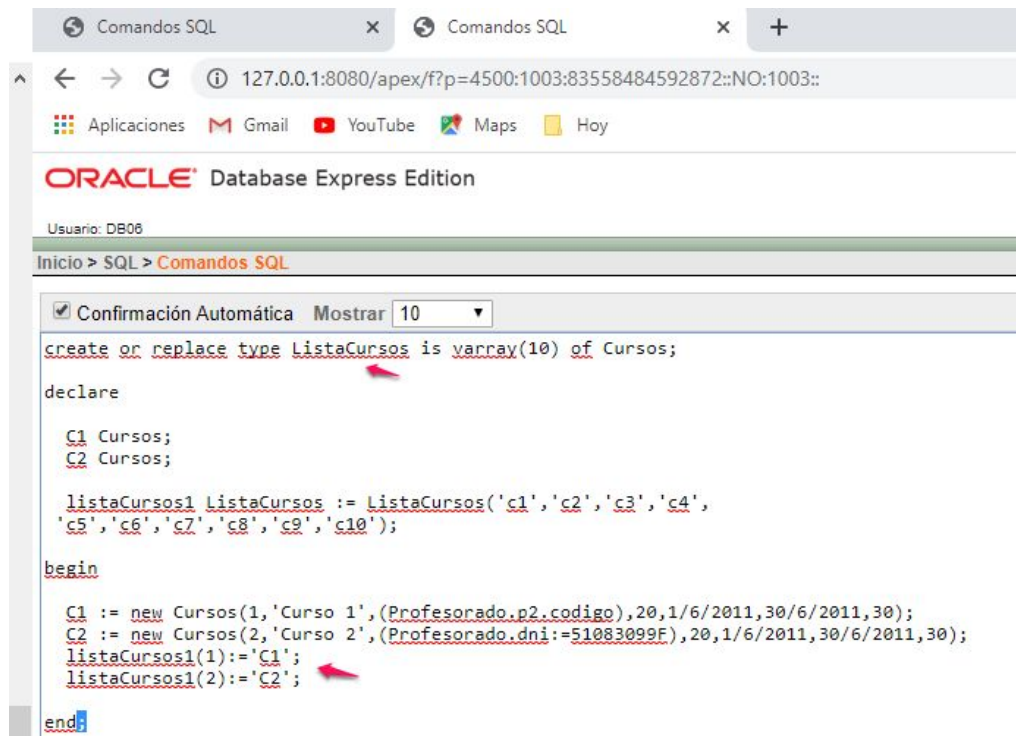
```
begin
```

```
  C1 := new Cursos(1,'Curso 1',(Profesorado.p2.codigo),20,1/6/2011,30/6/2011,30),  
  C2 := new Cursos(2,'Curso 2',(Profesorado.dni:=51083099F),20,1/6/2011,30/6/2011,30);  
  listaCursos1(1):='C1';  
  listaCursos1(2):='C2';
```

```
end;
```

Sobra

Hay que hacer select para recuperar los datos



```
Comandos SQL x Comandos SQL x +  
127.0.0.1:8080/apex/f?p=4500:1003:83558484592872::NO:1003::  
Aplicaciones Gmail YouTube Maps Hoy  
ORACLE Database Express Edition  
Usuario: DB06  
Inicio > SQL > Comandos SQL  
Confirmación Automática Mostrar 10  
create or replace type ListaCursos is varray(10) of Cursos;  
declare  
  C1 Cursos;  
  C2 Cursos;  
  listaCursos1 ListaCursos := ListaCursos('c1','c2','c3','c4',  
    'c5','c6','c7','c8','c9','c10');  
begin  
  C1 := new Cursos(1,'Curso 1',(Profesorado.p2.codigo),20,1/6/2011,30/6/2011,30);  
  C2 := new Cursos(2,'Curso 2',(Profesorado.dni:=51083099F),20,1/6/2011,30/6/2011,30);  
  listaCursos1(1):='C1';  
  listaCursos1(2):='C2';  
end;
```

0

6. Crea una tabla "Alumnado" de objetos "Alumno". Inserta en dicha tabla las siguientes filas:

```
codigo: 100
dni: 76401092Z
nombre: MANUEL
apellidos: SUAREZ IBAÑEZ
sexo: M
fecha_nac: 30/6/1990
cursoAlumno: objeto creado anteriormente para el primer curso
codigo: 102
dni: 6915588V
nombre: MILAGROSA
apellidos: DIAZ PEREZ
sexo: F
fecha_nac: 28/10/1984
cursoAlumno: objeto que se encuentre en la segunda posición de
"listaCursos1"
(debe tomarse de la lista)
```

### Solución:

Error compilación

```
create table Alumnado of Alumno;
```

```
declare
```

```
  a1 Alumno;
```

```
  a2 Alumno;
```

```
begin
```

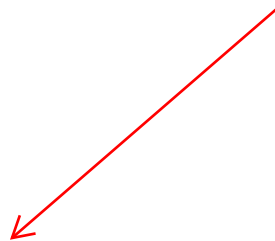
```
  a1 := new Alumno(100,'76401092Z','MANUEL','SUAREZ IBAÑEZ','M',30/6/1990,Curso1);
```

```
  a2 := new Alumno(102,'6915588V','MILAGROSA','DIAZ PEREZ',28/10/1984,xxx);
```

```
  insert into Alumnado values a1
```

```
  insert into Alumnado values (a2);
```

```
end;
```



```
create table Alumnado of Alumno;

declare
  a1 Alumno;
  a2 Alumno;

begin
  a1 := new Alumno(100,'76401092Z','MANUEL','SUAREZ IBÁÑEZ','M',30/6/1990,C1);
  a2 := new Alumno(102,'6915588V','MILAGROSA','DÍAZ PEREZ',28/10/1984,xxx);
  insert into Alumnado values (a1);
  insert into Alumnado values (a2);
end;
```

7. Obtener, de la tabla "Alumnado", el alumno que tiene el código 100, asignándoselo a una variable "unAlumno". Modifica el código del alumno guardado en esa variable "unAlumno" asignando el valor 101, y su curso debe ser el segundo que se había creado anteriormente. Inserta ese alumno en la tabla "Alumnado".
8. Crea un método MAP "ordenarCursos" para el tipo "Cursos". Este método debe retornar el nombre completo del profesor al que hace referencia cada curso. Para obtener el nombre debes utilizar el método getNombreCompleto que se ha creado anteriormente.

Realiza una consulta de la tabla "Alumnado" ordenada por "cursoAlumno" para comprobar el funcionamiento del método MAP.

LISTADO DE OBJETOS CREADO EN EL EJERCICIO:

Oracle Database Express Edition

Usuario: DB06

Inicio > Explorador de Objetos

Tipos

- ALUMNO
- CURSOS
- LISTACURSOS
- MIEMBROESCOLAR
- PROFESOR

Detalles de Objeto Sinónimos Permisos SQL

Borrar

Propietario	DB06
Nombre de Tipo	PROFESOR
Oid de Tipo	6CFAAC30C57C4D1CB4890AB73030116F
Objeto/Recopilación	OBJECT
Número de Atributos	8
Número de Métodos	0
Predefinido	NO
Completo	YES
Final	YES
Instanciable	YES
Propietario de Supertipo	DB06
Nombre de Supertipo	MIEMBROESCOLAR
Atributos Locales	2
Métodos Locales	0

Recuentos de Línea de Unidad x +

127.0.0.1:8080/apex/f?p=4500:1054:83558484592872::NO::

Aplicaciones Gmail YouTube Maps Hoy

**ORACLE** Database Express Edition

Usuario: DB06

Inicio > Utilidades > Informes de Objeto > PL/SQL > Recuentos de Línea de Unidad

Nombre del Objeto  Mostrar 15 Ir

Tipo De Objeto	Nombre De Objeto PL/SQL	Líneas ▼
TYPE BODY	PROFESOR	25
TYPE	LISTACURSOS	18
TRIGGER	INTEGRIDAD_AGENTES	13
TYPE	CURSOS	9
TYPE	MIEMBROESCOLAR	9
TYPE	PROFESOR	4
TYPE	ALUMNO	3
		81
		1 - 7

