

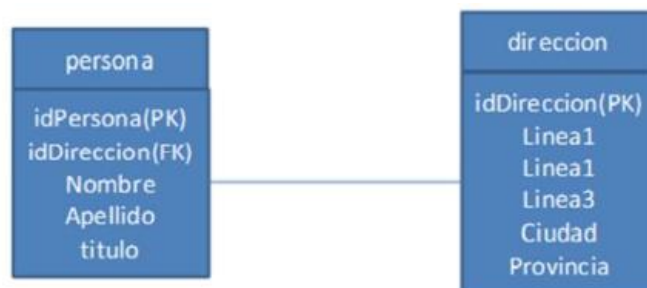
Ejercicio 01

Identifica los tipos de objeto, describiendo con pseudocódigo sus atributos y métodos, de las siguientes descripciones:

- Se desea registrar números de teléfono en una base de datos. Un número de teléfono viene dado por el código del país (dos dígitos), código de la región (tres dígitos) y el número (siete dígitos). Ejemplo:00-124-3566987
- Se desea registrar los datos de alumnos en la base de datos de un colegio. De cada alumno se necesita su DNI, nombre y apellidos, dirección (calle, código postal, población, provincia), fecha de nacimiento, teléfono y curso en el que se ha matriculado. De cada alumno se podrá conocer su edad.
- Arnau va a un banco para abrir una cuenta corriente. El encargado le explica que la información que podrá consultar por internet es su número de cuenta, el nombre del titular y el saldo de interés. También le comenta que podrá gestionar ingresos y hacer transferencias. Finalmente, Arnau abre dos cuentas: una de ahorro y otra de vivienda.

Ejercicio 02

Dada esta definición de tablas relacionales:



- Crea un tipo que sirva para declarar variables que almacenen objetos de tipo dirección.
- Partiendo del tipo anterior, crea un tipo que sirva para declarar variables que almacenen objetos del tipo persona.

Ejercicio 03

Dada esta definición de tablas relacionales:

<pre>CREATE TABLE PRODUCTOS (ID NUMBER (5) PRIMARY KEY, NOMBRE VARCHAR2(80), ID_CAT NUMBER (2) REFERENCES CATEGORIAS, ID_PROV NUMBER (3) REFERENCES PRODUCTOS);</pre>	<pre>CREATE TABLE CATEGORIAS (ID NUMBER (2) PRIMARY KEY, NOMBRE VARCHAR2(50));</pre>	<pre>CREATE TABLE PROVEEDORES (ID NUMBER (2) PRIMARY KEY, NOMBRE VARCHAR2(50), CIUDAD_SEDE VARCHAR2(50));</pre>
---	--	---

- a) Crea un tipo que sirva para declarar variables que almacenen objetos de tipo categoría.
- b) Crea un tipo que sirva para declarar variables que almacenen objetos de tipo proveedor.
- c) Utiliza los dos tipos anteriores para crear un tipo para productos.

Ejercicio 04

Crea un nuevo tipo que se llame NOTICIA_T que pueda almacenar:

- Un código de noticia
- La fecha de publicación de la noticia
- El número de días que tiene que estar publicada la noticia
- El texto de la noticia

Para este tipo, crea una función que retorne si ahora mismo esta noticia tiene que estar publicada o no.

Crea un bloque PL/SQL en el que se declare y se cree un objeto del tipo NOTICIA_T, con fecha de publicación igual a la fecha actual y número de días en los que la noticia tiene que estar publicada igual a 2. Muestra por consola si la noticia debe estar o no publicada.

Ejercicio 05

Crea un tipo que se llame `Alumno_T` que permita almacenar como atributos el DNI del alumno, la nota de la primera evaluación, de la segunda evaluación y de la tercera. Una vez definido el tipo asócialo dos funciones:

- `CalcMedia`, que tomará los valores de las tres notas, los promediará y retornará el resultado sin decimales.
- `CalcMediaPond`, que recibirá como parámetros el porcentaje que vale cada evaluación sobre la nota final. Por ejemplo: 25, 50, 25. Esta función debe retornar la nota media teniendo en cuenta esta ponderación.

Crea un bloque PL/SQL en el que se declare y se cree un objeto del tipo `ALUMNO_T` y calcula la nota media y la nota ponderada teniendo en cuenta que el peso de las evaluaciones es 25%, 50%, 25%

Ejercicio 06

Al tipo que has creado para almacenar productos del ejercicio 2, añádele dos atributos:

`price` y `tax`. Crea una función nueva que calcule el precio final de este producto teniendo en cuenta el precio base del producto (`price`) y la tasa asociada (`tax`). Tened en cuenta que `tax` se da en porcentaje: si es 15, se refiere a que es el 15%.

Crea un bloque PL/SQL en el que se declare y se cree un objeto del tipo `PRODUCTO_T` y calcula su precio final teniendo en cuenta que el precio base es de 5.25€ y la tasa asociada del 21%.

Ejercicio 07

Crea un tipo de objeto `rectangle_t` que tenga como atributos longitud, anchura y área. Define un constructor que tenga únicamente dos parámetros (longitud, anchura) y que calcule el área multiplicando los parámetros recibidos de longitud y altura. Define en un bloque PL/SQL dos objetos `rectangle_t`, usando el constructor por defecto y el constructor creado. Muestra por pantalla, el valor del área de cada uno de los rectángulos creados.

Ejercicio 08

- a) Implementa la función MAP en categoria_T para que el atributo que decida la ordenación sea el nombre de la categoría.
- b) Implementa la función MAP en proveedor_T para que el atributo que decida la ordenación sea el CIF.
- c) Implementa la función ORDER en NOTICIA_T. Se deben considerar menores los que tengan una fecha de publicación anterior y, mayores, los tengan una fecha de publicación más reciente.

Ejercicio 09

- a) Define el tipo de objeto CUSTOMER_T que disponga como atributos de:
 - Id: de tipo VARCHAR2
 - Name: de tipo VARCHAR2
 - Address: de tipo VARCHAR2
- b) Haz que el objeto CUSTOMER_T se pueda comparar por su name haciendo uso de MAP MEMBER METHOD. Demuestra que la implementación es correcta, mediante un bloque PL/SQL: define dos objetos CUSTOMER_T, compáralos y muestra por consola el resultado (cuál es el customer con name menor).
- c) Define el tipo de objeto CUSTOMER_T de forma que permita comparar este tipo de objetos por su name haciendo uso de ORDER MEMBER METHOD. Demuestra que la implementación es correcta según lo indicado en el apartado b)

Ejercicio 10

- a) Crea una tabla de objetos noticia_t que se llame noticias_obj. La clave primaria debe ser el código. Inserta 3 valores y comprueba que los valores se han insertado correctamente. Realiza un SELECT de la tabla noticias_obj para verificar que las noticias se han insertado correctamente.
- b) Crea una tabla que contenga un objeto noticia_t y un varchar2 que se llame sección. La clave primaria de la tabla debe ser el atributo código del objeto noticia. Inserta 3 valores y comprueba que se han insertado correctamente.

Ejercicio 11

Usa el tipo de objeto noticia_t y la tabla noticias_obj. Modifica la definición del tipo de objeto noticia_t para que disponga de un método estático que permita insertar una nueva noticia en la tabla noticias_obj. Usa un bloque PL/SQL para usar el método static definido. Realiza un SELECT de la tabla noticia_obj para verificar que la noticia se ha insertado correctamente.

Ejercicio 12

Crea un tipo de objeto person_t que permita la definición de subtipos y pueda almacenar:

- Idno: entero
- name character (30)
- surname character (30)
- Birth: fecha
- phone character

Y los siguientes métodos.

- get_datos: método que devuelve una string con el name, surname y phone
- get_fecha: método que permita la ordenación de personas por su birth (MAP).

Crea un subtipo de person_t denominado student_t que permita la definición de subtipos y pueda almacenar:

- college: carácter (30)
- averageScore: integer

Debe sobrescribir el método get_datos mostrando name, surname, phone, college y averageScore

Crea un subtipo de student_t denominado student_parcial_t que pueda almacenar:

- numHours: integer

Debe sobrescribir el método get_datos mostrando name, surname, phone, college y averageScore y numHours

En un bloque PL/SQL,

- a) Crea un objeto tipo student_t y otro student_parcial_t. Usa el método get_datos para mostrar en pantalla los datos de cada uno de ellos. Muestra por pantalla también cuál de ellos es más joven.
- b) Crea una tabla de objetos del tipo person_t (PK idno) con los siguientes registros

Personas

IDNO	NAME	SURNAME	BIRTH	PHONE
12	ISMAEL	BELTRAN	15/02/1995	22446688
27	BERTA	MATEO	23/09/1991	22446688
29	GONZALO	CASANOVA	06/05/1993	22446688

Estudiantes

IDNO	NAME	SURNAME	BIRTH	PHONE	COLLEGE	AVERAGESCORE
15	FRANCISCO	SUAREZ	15/02/1990	22446688	LA SALLE GRACIA	9.0
20	JOSE	PEREZ	23/09/1992	22446688	LA SALLE GRACIA	7.5

Estudiantes a tiempo parcial

IDNO	NAME	SURNAME	BIRTH	PHONE	COLLEGE	AVERAGESCORE	NUMHOURS
24	JAVIER	GARCIA	20/05/1990	22446688	LA SALLE GRACIA	8.0	317
25	ELENA	CASTELLANOS	23/10/1990	22446688	LA SALLE GRACIA	9.5	317

- c) Muestra por pantalla los datos (get_datos) de los estudiantes (sólo estudiantes) ordenados por fecha de nacimiento (birth)
- d) Muestra por pantalla los datos (get_datos) de los estudiantes a tiempo parcial ordenados por fecha de nacimiento (birth)

Ejercicio 13

Se desea disponer de una base de datos de los inmuebles que tiene a la venta una promotora. Un inmueble se caracteriza por su superficie, precio y ciudad donde se ubica. La promotora tiene a la venta los siguientes tipos de inmuebles:

- Oficina: que se caracteriza por número de plantas y si dispone o no de ascensor.
- Vivienda: que se caracteriza por número de habitaciones, número de baños, si tiene terraza o no, si tiene piscina o no, si tiene garaje o no y si tiene ascensor o no.

En el caso de viviendas, dispondrá de un método (getTipo) que devuelva su clasificación en función del número de habitaciones:

- 1hab-> Clasificación: estudio
- 2hab->Clasificación: apartamento
- >2hab->Clasificación: piso

Crea los tipos y subtipos necesarios, permitiendo almacenar la información citada. Los inmuebles se ordenarán por su precio (de menor a mayor) y dispondrán de una función (show) que permita devolver la descripción del inmueble (la descripción del inmueble será una cadena de caracteres que indicará si es una Oficina o una Vivienda y contendrá las características citadas según el tipo de inmueble).

Crea una tabla que disponga de 2 campos, un campo para contener inmuebles, y un campo que sirva para almacenar la fecha de alta del inmueble en la base de datos. Introduce en la tabla varios registros de tipo vivienda y oficina.

Realiza las siguientes consultas:

- Muestra las descripciones de cada uno de los inmuebles insertados en la tabla ordenados por precio (de menor a mayor).
- Muestra las descripciones de los inmuebles que fueron dados de alta hace más de un mes.
- Muestra las descripciones de los inmuebles cuyo precio se encuentre entre dos determinadas cantidades.
- Muestra las descripciones de los inmuebles que sean viviendas.

Ejercicio 14

Crea un nuevo tipo que se llame empleado (employee_t) que tenga como atributos el nombre del empleado (name) y quién es su jefe (manager). El atributo manager será una referencia al mismo tipo employee_t.

Una vez creado el tipo, crea una tabla (employees) capaz de almacenar objetos de ese tipo.

Después, inserta 2 filas en esta tabla. La primera fila será un empleado que no tendrá jefe.

La segunda fila será un empleado que tendrá como jefe al primero.

Ejercicio 15

Crea un nuevo tipo que se llame departamento (department_t) que tenga como atributos el nombre del departamento (name) y el empleado (manager) que dirige este departamento (una referencia a un empleado).

Crea una nueva tabla (departamentos) capaz de almacenar objetos del tipo department_t.

Inserta 2 filas en esta tabla, las dos con un manager.

Ejercicio 16

En una clínica veterinaria, se desea disponer de los datos asociados a las mascotas que son atendidas, así como de los veterinarios que las atienden.

Cada mascota tiene un veterinario asignado.

Un veterinario se identifica de forma única por su número de colegiado, almacenándose además su nombre y dirección. De cada mascota se quiere almacenar su identificador (único), fecha de nacimiento, raza y el veterinario asignado. Se desea disponer de 2 tablas en la base de datos para las mascotas y para los veterinarios.

- Construye los tipos de objetos necesarios, así como las tablas, haciendo uso de referencias a objetos.
- Inserta registros en las tablas
- Realiza una consulta de la fecha de nacimiento e identificador de las mascotas asignadas a un determinado veterinario

Ejercicio 17

Crea un tipo para almacenar números de teléfono cuyos atributos serán:

- Código del país VARCHAR2(2)
- Código de la región VARCHAR2(3)
- Número VARCHAR2(7)

Ejemplo:00-124-3566987

- a) Crea un tipo de datos colección que permita almacenar 5 números de teléfono
- b) Crea una tabla "agenda" que permita almacenar los números de teléfono asociados a un nombre de persona.
- c) Inserta valores en la tabla para distintas personas.
- d) Modifica uno de los números de teléfono asociados a una persona que exista en la agenda
- e) Muestra por consola los números de teléfono completos de una de las personas que hayas insertado en la tabla

Ejercicio 18

Una empresa de ingeniería dispone en su plantilla de una relación de jefes de proyecto. La empresa se encarga de redactar proyectos. Un proyecto viene definido por su nombre, por ejemplo "PROYECTO ACONDICIONAMIENTO CARRETERA CV-70 TRAMO PK 0+000 A PK1+500" y la fecha de finalización o de entrega.

Cada proyecto tiene asociado un conjunto de planos. Un plano viene definido por un título y los nombres de los ingenieros firmantes. Como máximo un plano puede estar firmado por 2 ingenieros.

Se desea disponer de una tabla de jefes de proyecto con su nombre y teléfono, y una tabla de proyectos con el nombre del proyecto, la fecha de finalización y el jefe de proyecto asignado.

Tener en cuenta que un proyecto está dirigido por un jefe de proyecto (un proyecto siempre está asignado a un jefe de proyecto) y un jefe de proyecto puede dirigir 0 o varios proyectos.

Construye los tipos de objetos necesarios, así como las tablas, haciendo uso de referencias a objetos. Inserta registros en las tablas de jefes de proyecto y de proyectos.

Sobre uno de los proyectos de la tabla de proyectos que hayas definido:

- a) Introduce los siguientes planos:

ID	TITULO	INGENIEROS FIRMANTES
01	SITUACION Y EMPLAZAMIENTO	Ingeniero A, Ingeniero B
02	SERVICIOS AFECTADOS	Ingeniero C, Ingeniero D
03	SEGURIDAD Y SALUD	Ingeniero E, Ingeniero F

- b) Actualiza la tabla para que en el plano 02, sólo este como ingeniero firmante 'Ingeniero C'

- c) Introduce un nuevo plano con los siguientes datos:

ID	TITULO	INGENIEROS FIRMANTES
04	REPLANTEO INICIAL	Ingeniero G, Ingeniero H

- d) Actualiza el título del plano con id=02 para que sea 'SERVICIOS AFECTADOS EN RED DE

TELEFONIA'

- e) Elimina el plano cuyo título es 'SEGURIDAD Y SALUD'

Ejercicio 19

Se desea construir un modelo para la gestión de empresas:

- a) Crea un tipo para las direcciones de empresas, contendrá el tipo de dirección y la dirección (número, calle y piso).
- b) Crea un tipo colección para almacenar hasta 3 direcciones (fiscal, postal y administrativa).
- c) Crea un supertipo empresa con los atributos CIF, nombre y direcciones.
- d) Crea un subtipo de empresa llamada sociedad anónima con los atributos número de accionistas, capital social y presupuesto
- e) Crea un subtipo de empresa llamada sociedad limitada que tenga una lista de socios, donde se especifique el nombre del socio y un porcentaje de posesión de la empresa
- f) Crea una tabla de empleados que tenga una referencia a Empresa
- g) Crea una tabla de empresas del tipo sociedades anónimas. Inserta una sociedad anónima con dos direcciones
- h) Crea una tabla de empresas de tipo sociedades limitadas. Inserta una sociedad limitada con dos direcciones y cuatro socios.
- i) Inserta un empleado para cada una de las empresas insertadas.
- j) Crea una consulta que muestre la empresa para la que trabajan los empleados.