

ASIGNATURA:

PROGRAMACIÓN

UT4

PRÁCTICA 4

ALUMNO:

FERNANDO VALERO RUIZ

PRÁCTICA 4

Ejercicio 1: Genera el diagrama UML que represente las clases en los siguientes casos:

- a) Una clase Empleado que tenga los atributos *dni* y *nombre* de tipo String accesibles cualquier clase, un atributo *experiencia* de tipo entero accesible únicamente desde clases en el mismo paquete, y un atributo *sueldo* de tipo double y no accesible desde fuera de su propia clase. Como métodos se tiene *programar* al cual se le pasa el nombre del proyecto actual de tipo String y devolverá un booleano que indique si trabaja en desarrollo o no; otro método es *dirigir* al que se le indica también el nombre del proyecto actual y devuelve el número de empleados bajo su dirección. Los métodos solo pueden ser accedidos desde su propia clase.

- b) Una clase Gato que tenga como atributos *edad* (de tipo entero y sin modificador), *peso* (de tipo double y accesible solo desde clase Gato), *nombre* (con acceso desde cualquier clase), y *raza* (de tipo String y accesible desde cualquier clase dentro de su paquete y por cualquier subclase independientemente del paquete). Los métodos de esta clase son *correr* (el cual requerirá de una variable tipo double y no devolverá ningún valor) y *saltar* (sin parámetros de entrada y que devuelve un double). Ningún método de la clase puede ser accedido desde fuera de esta.

Respuesta:

Empleado
+ dni: String + nombre: String experiencia: int - sueldo: double
- programa (String): boolean - dirigir(String): int

Gato
- edad: int - peso: double + nombre: String # raza: String
- correr (double) - saltar(): double

Ejercicio 2: Escribe un programa que sirva para tener registradas las ventas de zapatos en una zapatería. Se requiere:

- Una clase ZapatoMain en un fichero .java que contendrá el método main.
- Una clase Zapato en otro fichero .java distinto, con los atributos *tipo*, *color*, *número* y *liquidación* de acceso público, y los atributos *cantidad* (pares de ese tipo almacenados) y *precio*, ambos de acceso privado.
- La clase Zapato tiene además todos sus métodos públicos: el método *almacenado* que devolverá la cantidad de pares disponibles de ese tipo y el método *precioVenta* que devolverá el precio de un par concreto.
- En la clase ZapatoMain se debe construir o instanciar un objeto *par1* con los siguientes atributos: tipo: "Náutico", color: "Azul", número: 42, liquidación: false, cantidad: 10 y precio: 40, y un objeto *par2* con los atributos: nombre: "De salón", color: "Rojo", número: 38, liquidación: true, cantidad: 3 y precio: 25.
- Finalmente, el programa debe imprimir los atributos de los objetos por pantalla:

```
run:
Tipo: Náutico   Color: Azul   Número: 42   En liquidación: false   Cantidad en almacén: 10   Precio: 40.0
Tipo: De salón  Color: Rojo   Número: 38   En liquidación: true    Cantidad en almacén: 3    Precio: 25.0
BUILD SUCCESSFUL (total time: 0 seconds)
```

Respuesta: Zapatos

Ejercicio 3: Se necesita crear un programa que solicite, almacene y presente información sobre los alumnos de un Instituto. Para ello:

- Crea tanto la clase AlumnoMain con el método *main*, como la clase Alumno cuyos atributos serán privados. Ambas clases pueden estar en el mismo fichero .java o en ficheros diferentes según prefieras.
- La información que interesa almacenar de cada estudiante es el nombre del alumno, su NIA (ej: "123"), el nombre del ciclo y su curso.
- El programa deberá solicitar al usuario la inserción de los datos de 2 alumnos, y al finalizar deberá mostrarlos por pantalla. Mira el ejemplo de salida:

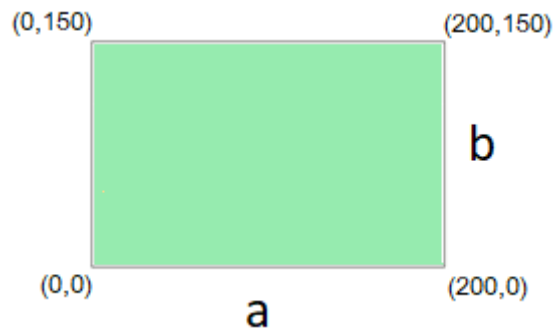
```
run:
Introduce los datos del alumno 1:
Nombre: Luis
NIA: 123
Ciclo: ASIR
Curso: 1
Introduce los datos del alumno 2:
Nombre: Paula
NIA: 456
Ciclo: DAM
Curso: 2

Alumno 1:      Nombre: Luis   NIA: 123      Ciclo: ASIR   Curso: 1
Alumno 2:      Nombre: Paula  NIA: 456      Ciclo: DAM    Curso: 2
BUILD SUCCESSFUL (total time: 27 seconds)
```

Respuesta: Instituto

Ejercicio 4: Se quiere calcular el área de un terreno a partir de sus coordenadas. Para ello:

- Crea una clase llamada Posicion con los atributos x , y (coordenadas) y una clase principal PosicionMain que contenga el método *main*.
- Crea un método que calcule el área del terreno de la siguiente figura, teniendo en cuenta que: $\text{Área} = a \cdot b$



- El resultado de la ejecución debe ser:

```
run:
Punto 1: (0, 0) m
Punto 2: (200, 0) m
Punto 3: (200, 150) m
Punto 4: (0, 150) m
Área: 30000.0 m2
BUILD SUCCESSFUL (total time: 0 seconds)
```

Respuesta: Terreno