

Ejercicio 1

AVES

Implementa la siguiente clase, añade todos los métodos necesarios para acceder públicamente a todos sus miembros:

```
public class Ave {  
  
    private String nombreVulgar;  
    private String nombreCientifico;  
    int numeroIndividuos;  
  
    public Ave(String nombreVulgar, String nombreCientifico, int numeroIndividuos) { } //Constructor  
  
    void incrementarNumeroIndividuos(int n){ } //Incrementa el valor de numeroIndividuos en n individuos  
    void decrementarNumeroIndividuos(int n){ } //Decrementa el valor de numeroIndividuos en n individuos  
  
    boolean peligroExtinción(){ } //Devuelve true si numeroIndividuos es menor a 100  
}
```

a) Crea una subclase de la clase Ave llamada AveVoladora que contenga además la siguiente propiedad:

```
private String tipoAve;
```

b) Crea un programa principal en una clase aparte que haga lo siguiente:

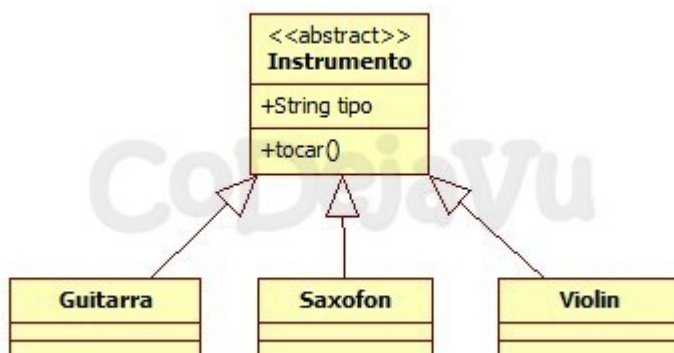
- ✓ Crear un ARRAY DE AveVoladoras (100 elementos), controla el nº de elementos reales que tienes en el array
- ✓ Añade AL array las siguientes aves:

```
"Halcón Mexicano", "Falco mexicanus", 85, "rapaz"  
"Águila Pescadora", "Pandion haliaetus", 105, "rapaz"  
"Tucán", "Ramphastos toco", 60, "trepadora"  
"Pavo Real", "Pavo cristatus", 92, "gallinácea"  
"Garza", "Mesophoyx intermedia", 98, "zancuda"  
"Flamenco", "Phoenicopterus", 56, "zancuda"
```

- ✓ Incrementa en 10 el número de individuos de las aves de la lista que estén en peligro de extinción.
- ✓ Emite un listado con todos los datos de las aves que no están en peligro de extinción.

Ejercicio 2

Con este esquema:



y este código:

```
/**
 * Clase Abstracta Instrumento
 */
abstract class Instrumento{

    public String tipo;

    public abstract void tocar();
}
```

Escribe las clases concretas, con sus constructores, y el método tocar por ej:

"Tocar La Guitarra entre 2 aguas"

"Tocar El violin. Re Mayor Op. 35, 1º movimiento –Tchaikovsky "

"Tocar el Saxofon **Born To Run** "

Además crea una clase testInstrumentos, para probar tu desarrollo

ejercicio3

Implementa las siguientes clases:

- clase Nota. Una nota contiene un identificador numérico y una línea de texto. Define constructor, setter y getter toString.
- clase NotaAlarma. Una nota que además contiene la hora en la que sonará la alarma. Define constructor, setter y getter toString. .
- clase BlocNotas que modela un bloc de notas en el que se pueden introducir notas, listar todas las notas, eliminar una nota mediante su posición en el bloc de notas o saber cuantas notas contiene el bloc de notas. Debes un array con un nº max de notas que se especifica al crear una instancia de bloc.
- clase Prueba que cree un bloc de Notas de ejemplo y pruebe las operaciones que soporta. .