
PRACTICA 1ª

1.1.- EJERCICIO 1

- Escribir el siguiente trozo de código en un editor de texto.

```
//Mi primer programa en Java
public class HolaMundo
{
    //Programa principal
    public static void main(String args[])
    {
        //Salida a pantalla
        System.out.println("¡¡¡Hola Mundo!!!");
    }
}
//Fin de mi primer programa en Java
```

- Guardar el fichero con extensión .java, el nombre del fichero será el mismo que el de la clase que contiene el programa principal (main).
- Para compilar el programa utilizaremos el compilador javac

```
javac <nombre completo del fichero>
```

Si no encuentra el fichero javac.exe hay que buscarlo y añadirlo al path del sistema

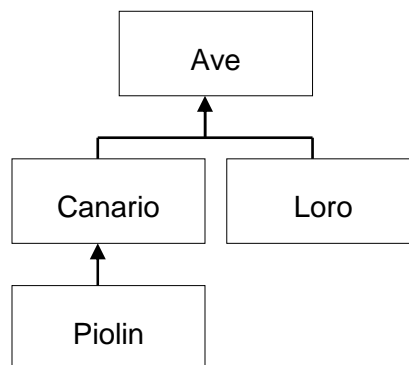
- A continuación hay que ejecutar el programa, para ello utilizaremos el intérprete java

```
java <nombre de la clase principal que contiene el main>
```

- Como resultado obtendremos por pantalla el mensaje

```
¡¡¡Hola Mundo!!!
```

1.2.- EJERCICIO 2



Implementar la siguiente jerarquía de clases

La clase Ave

- Tiene como atributos de instancia el sexo (M/H), la edad (entero) y como atributo de clase el número de aves creadas, que se incrementa cada vez que se crea un nuevo Ave
- Tiene como método el constructor que inicializa, con los parámetros que recibe, los dos atributos de la clase.
- Además existe un método de clase que indica el número de Aves creadas
- Además tiene un método quienSoy que saca los valores de los dos atributos por pantalla

La clase Canario:

- Tiene como atributos el sexo (M/H), la edad (entero) y el tamaño (real).
- La clase canario tiene dos constructores, uno inicializa el sexo y la edad según dos parámetros que le llegan y otro también inicializa el tamaño.
- Tiene un método altura que dependiendo del tamaño del canario saca un mensaje por pantalla ("Alto" si tamaño > 30, "Mediano" si tamaño [15..30] y "Bajo" si tamaño < 15)

La clase Loro:

- Tiene como atributos el sexo (M/H), la edad (entero), la región (N/S/E/O) y el color (String).
- Tiene un constructor que inicializa todos los atributos con parámetros que le llegan.
- Tiene un método deDondeEres que saca el mensaje "Norte", "Sur", "Este" u "Oeste" dependiendo de la región de donde sea el loro.

La clase Piolin:

- Tiene como atributos el sexo (M/H), la edad (entero) y el número de películas (entero).
- Su constructor inicializa todos los atributos con valores que le llegan como parámetros.
- En esta clase se encuentra el programa principal en el cual se crea una instancia de la clase Piolin y otro de la clase loro.
 - Llamar al método quienSoy de las dos instancias
 - Llamar al método altura con la instancia del Piolín
 - Invocar al método deDondeEres con la instancia del Loro
 - Modificar el tamaño del Piolín y volver a invocar al método altura
 - Modificar la región del Loro
 - Mostrar el número de Aves creadas.

1.3.- EJERCICIO 3

Modificar el ejercicio 2 para cumplir los siguientes requisitos:

- Debe haber ocultación de atributos en todos los casos que así lo requieran; en los casos necesarios se crearán los métodos set/get para acceder a los mismos
- Se creará una nueva clase llamada DatosPersonales, que tendrá como atributos el nombre del ave (String) y el nombre del dueño (String). Tendrá un constructor de clase en el que se inicialicen estos atributos y los métodos get/set para cada uno de los atributos, por separado
- La clase Ave tendrá un nuevo atributo llamado nombres de la clase DatosPersonales.
- La clase Ave tendrá un método abstracto que será cantar. Para un Piolín cantar será sacar el mensaje "Pio-pio soy un Piolín" y para un Loro cantar será sacar un mensaje diciendo "Piiio-piiiio loro bonito"
- El programa principal varía por completo.
 - Se creará un array de 3 Piolines y se inicializarán todos sus datos
 - Se asignará un nuevo nombre de animal y nuevo nombre de dueño al primero de los tres Piolines creados (recordad que no se puede acceder directamente a ninguno de los atributos ya que todos son privados)
 - Crear una instancia de la clase Loro
 - Hacer cantar al loro y a los tres Piolines
 - Sacar por pantalla el nombre del animal y del dueño del primer Piolín creado

1.4.- EJERCICIO 4

A partir del ejercicio anterior, se pide:

- Incluir cada una de las clases en un paquete. La relación de paquetes y clases es la siguiente:

La clase Ave en el paquete *aves*

La clase *Canario* en el paquete *aves.pajaros*

La clase *Loro* en el paquete *aves.tropicales*

La clase *Piolin* en el paquete *estrellastv*

La clase *MiPrograma* en el paquete *programas*

Sitúa cada una de las clases en el directorio que le corresponda en función del paquete al que pertenezca.

- En este momento el código que se ha obtenido no compila. Solúcnalo añadiendo:

- ✓ Todas las importaciones de clases (cláusula **import**) que sean necesarias.
- ✓ Todos los modificadores de acceso a las clases, a los atributos y a los métodos que sean necesarios. Intenta mantener (en la medida en que sea posible) el ocultamiento de la información, asignando el modificador de acceso más “restrictivo” que sea posible en cada caso.
- Ubicar todas las clases (junto con sus paquetes) dentro del directorio c:\Alumno
 - ✓ Configurar el classpath
 - ✓ Compilar y ejecutar
- Mover las clases Ave, Canario, Loro y Piolin (junto con sus correspondientes paquetes) al directorio c:\Alumno\Pajarracos
 - ✓ Configurar el classpath
 - ✓ Compilar y ejecutar
- Mover la clase MiPrograma (y su paquete) al directorio c:\Alumno\MiEjemplo
 - ✓ Configurar el classpath
 - ✓ Compilar y ejecutar
- Modificar el código para que la clase MiPrograma no se ubique dentro de ningún paquete (el paquete por defecto)
 - ✓ Configurar el classpath
 - ✓ Compilar y ejecutar