

### ACTIVIDAD DE ENSEÑANZA-APRENDIZAJE DIRIGIDA

<b>módulo:</b>	Programación (ciclo 1º DAW /DAM)
<b>Nombre de la tarea</b>	Tarea 3ª: Arrays de Objetos
<b>Tipo de tarea</b>	Creación de objetos en un array
<b>Duración estimada</b>	180 minutos
<b>Objetivos</b>	Manejar las clases en Java y las estructuras de almacenamiento.
<b>Material necesario</b>	PC con JAVA, NetBeans o ECLIPSE.
<b>Fecha de entrega</b>	Periodo marzo, 2º Trimestre. 21 al 27 de Marzo
<b>Forma de entrega</b>	En un archivo comprimido con el proyecto.
<b>Puntuación</b>	Cada Apartado 1 punto, excepto el f) y g) que valen 2 puntos.

## ARRAYS DE OBJETOS

### Tarea 03: ARRAYS DE OBJETOS.

#### Parte I: Creación e introducción de datos.

a) Crear una clase denominada Aspirante (number, name, secondname, levelStudies, email, phone).

El nivel de estudios que pueden tener son: Preschool, Elementary school, Middle school, High School, College, Graduate.

b) Crear los métodos constructores.

c) Crear los métodos set y get correspondientes.

d) Añade un método de toString que devuelve los datos de un aspirante separados por comas en un solo STRING.

d.1) Añade un método que imprima por pantalla los aspirantes y utiliza el método anterior creado.

e) Crea un método para insertar aspirantes e inserta a tres aspirantes desde la consola.

f) Crea tres métodos de ordenación:

I- por número.

II- Por nombre.

III- Por nivel de estudios (levelStudies).

Ordena el array y muéstralo por pantalla ordenado de cada forma.

g) Crea un nuevo método de impresión que muestre de los aspirantes los siguientes datos: secondName, name, levelStudies.

Añade un método de ordenación por apellido (secondName).

Muestra los aspirantes de forma Ascendente y descendente, ordenados por apellidos y solo muestre los tres campos anteriores.

#### PARTE II- INTRODUCCION A LA INTERFAZ GRÁFICA (OPCIONAL)

Cambia la forma de introducir los datos de los aspirantes para que sea por pantalla en lugar de por consola. Puedes utilizar la librería grafica de Swing, mediante la componente **swing.JOptionPane**.

Programación.

UT6-Estructuras de almacenamiento

Nota: Para utilizar un grupo constantes en Java se puede utilizar las enumeraciones o enum.

```
enum Color
{
    ROJO, VERDE, AZUL;
}

public class Test
{
    // El método
    public static void main(String[] args)
    {
        Color c1 = Color.ROJO;
        System.out.println(c1);
    }
}
```

Para comparar cadenas en Java con datos de tipo ENUM, inicialmente son incompatibles, pero existen diferentes formas:

Método **.name()** que devuelve una cadena.

Ejemplo: `c1.name().equalsIgnoreCase("cadena_a_comparar");`