

# ***Programación Modular y Orientación a Objetos - Laboratorio 1***

## Requisitos previos

Se asume que el alumnado está familiarizado con el entorno de programación Eclipse, que se ha utilizado a lo largo del primer cuatrimestre en la asignatura Programación Básica para desarrollar programas en Ada.

## Objetivos

Este laboratorio presenta la perspectiva Java, que es la que se utilizará en la asignatura PMOO. Además, se trabajarán los conceptos básicos de la programación en Java en el entorno Eclipse, mediante la creación de proyectos, paquetes y clases sencillas. Más adelante, en laboratorios posteriores, se incrementará la complejidad de los mismos.

Al finalizar este laboratorio, el alumnado deberá ser capaz de:

- Entender el funcionamiento del entorno Eclipse y desenvolverse en el uso de la perspectiva Java.
- Crear un espacio de trabajo (*workspace*) en el que almacenar sus desarrollos.
- Crear proyectos, paquetes y clases muy sencillas, con atributos simples y métodos básicos.
- Familiarizarse con la sintaxis del lenguaje Java.

## Motivación

A lo largo del curso se va a utilizar la perspectiva Java del entorno Eclipse para escribir código. Por lo tanto, conviene conocer su funcionamiento.

En la resolución de los problemas que se planteen se utilizará la orientación a objetos, por lo que es fundamental dominar los conceptos básicos presentados y que se irán complementando en sucesivos laboratorios.

Por otra parte, conviene conocer cómo gestionar los *workspaces* de cara a almacenar y recuperar el trabajo de cada alumno o grupo de alumnos.



## **Tarea 1: el entorno Eclipse y la perspectiva Java**

En esta sencilla actividad solamente se pide arrancar el entorno Eclipse sobre un *workspace* que no sea el propuesto por defecto, sino que sea particular (por ejemplo, contenido en una carpeta de un dispositivo de almacenamiento USB externo) y con la perspectiva Java habilitada.



## **Tarea 2: los proyectos, los paquetes y las clases**

Una vez preparado el entorno de desarrollo, es momento de comenzar a trabajar con el primer proyecto Java de la asignatura. En esta actividad se pide definir una clase (Persona) con dos atributos (nombreCompleto y edad), y la constructora correspondiente.



## **Tarea 3: la programación de métodos en Java**

En este punto lo que se pide es añadir más funcionalidad a la clase Persona creada en la tarea anterior. En particular:

- Tres nuevos atributos: `idPersona` (entero), `grupoSanguíneo` (String), y `nacionalidad` (String).
- Una nueva constructora que reciba como parámetros los valores con los que inicializar los cinco atributos.
- Un método `tieneMismoID()`, que recibe como parámetro un objeto de tipo persona (llámese `pPersona`) y devuelve un booleano indicando si la persona actual (*this*) tiene el mismo atributo `idPersona` que `pPersona`.
- Un método `puedeConducir()`, que determine si la Persona tiene la edad mínima permitida para obtener el carné de conducir. Por simplificar, se supondrá que esto ocurre a los 14 años cuando la nacionalidad es “Etiópe”, a los 16 años cuando la nacionalidad es “Australiana” o “Estadounidense”, a los 17 años cuando es “Británica”, y a los 18 años para cualquier otra nacionalidad.
- Un método `inicialNombre()`, que devuelve la inicial del nombre de la Persona, y otro método `inicialApellido()`, que devuelve la inicial del Apellido. Se supondrá que el atributo `nombreCompleto` no contendrá apellidos compuestos, aunque puede tener nombres compuestos.

**Ejemplos:**

Nombre Completo	Inicial nombre	Inicial apellido
Juan Martín	J	M
Juan Pedro Martín	J	M
Juan Pedro Martín García	J	G

- Un método *puedeDonarleSangre()*, que recibe como parámetro un objeto de tipo persona (llámese pPersona), y devuelve un booleano que indica si la persona actual (*this*) puede donar sangre a pPersona. La tabla siguiente puede resultar de ayuda:

		Donante							
		O+	O-	A+	A-	B+	B-	AB+	AB-
Receptor	O+	X	X						
	O-		X						
	A+	X	X	X	X				
	A-		X		X				
	B+	X	X			X	X		
	B-		X				X		
	AB+	X	X	X	X	X	X	X	X
	AB-		X		X		X		X