

## PRÁCTICA 2 de JAVA: Swing y AWT de forma manual.

22/09/2021

<b>Fecha entrega</b>	Lunes, 27 de septiembre.	<b>Fecha defensa</b>	Miércoles, 29 de septiembre.
<b>Tipo</b>	Individual	<b>Asunto e-mail</b>	JAVA02
<b>Formato fichero</b>	ApellidoNombreJAVA02.zip (compartirlo por Google Drive)		

Desarrolla una aplicación gráfica en Java (JFragment) con NetBeans, implementando los gráficos desde el código Java y sin usar las herramienta que proporciona el IDE Netbeans.

Requisitos:

- 1.- Crea una clase libre con al menos 4 atributos con los tipos String, float, GregorianCalendar y otro tipo primitivo de Java.
- 2.- Desarrolla una lista enlazada de datos para almacenar objetos de la clase anterior.
  - 2.1.- Tendrá un apuntador inicial al primero objeto de la lista.
  - 2.2.- Cada elemento tendrá un apuntador al siguiente.
  - 2.3.- Aportará los métodos necesarios para su manejo y serán la única forma de manejar la lista desde otras clases.
  - 2.4.- Pueden añadirse las mejoras que creas oportunas.
- 3.- Desarrolla un interfaz gráfico para visualizar los elementos (objetos) almacenados. Deberán cargarse 10 elementos al inicio de forma automática).
  - 3.1.- Estará basado en la clase "Doble" de la carpeta adjunta "pq1AWT\_Swing"
  - 3.2.- Aparecerán JLabels descriptivos y TextFields para visualizar los atributos del empleado.
  - 3.3.- Tendrá dos JButtons para moverse sobre la estructura de datos: anterior y siguiente.
  - 3.4.- Los botones se desactivarán al principio (anterior) y al final (siguiente).
  - 3.5.- Después del ultimo elemento se permitirá avanzar a un nuevo registro con todos los campos en blanco donde se podrá insertar un nuevo elemento.
  - 3.6.- Habrá dos JButtons para el anterior apartado (alta o cancelar). Solo se podrá usar en ese caso. Pueden ser nuevos o renombrar los botones anterior y siguiente.
  - 3.7.- No deben producirse errores (desactivar y activar botones).
- 4.- Aparecerán distintivos elementos gráficos que deberá estar en contenedores intermedios (JPanels) con diferentes layouts.
- 5.- Debe estar basado en capas según el Modelo Vista Controlador (MVC). Las clases deben ser independientes y la comunicaciones entre ellas ha de realizarse a través de métodos.
- 6.- Debe funcionar en los equipos del taller del Centro.
- 7.- Control de errores en los nuevos campos a insertar. Se hará con una clase exclusiva.