

Módulo Profesional 03:

Programación II

Actividades Prácticas Evaluables UF5

CICLO FORMATIVO DE GRADO SUPERIOR EN

DESARROLLO DE APLICACIONES MULTIPLATAFORMA

MODALIDAD ONLINE



Ejercicios Prácticos

Objetivos

- Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos
- Identificar los cambios tecnológicos, organizativos, económicos y laborales en su actividad, analizando sus implicaciones en el ámbito de trabajo, para mantener el espíritu de innovación

Competencias asociadas:

- Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos

Metodología

- Preparación individual
- Para la realización del ejercicio se deberán visualizar todos los videotutoriales del curso.

Entrega

19 de noviembre del 2020.

Se deberá entregar un único proyecto en Java o C++ comprimido en **Zip** con la nomenclatura

MP03_PII_UF05_APE.zip

(el resto de requisitos de entrega se describen en la sección descripción de la actividad).

Dedicación estimada

12 horas

Documentos de referencia

Videotutoriales de la UF.

Libro de referencia de la asignatura

Resultados de aprendizaje

- Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.

Criterios de evaluación

- Se han identificado los bloques que componen la estructura de un programa informático.
- Se han utilizado entornos integrados de desarrollo.
- Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.
- Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.
- Se han introducido comentarios en el código.

Enunciados

1.- Registrar usuario

The image shows a Java Swing window titled "Registrar usuario". It has a standard Windows-style title bar with minimize, maximize, and close buttons. The window contains the following components:

- Nombre:** A text input field with the label "Nombre" to its left. The component is identified as `JTextField` in red text.
- Año nacimiento:** A text input field with the label "Año nacimiento" to its left. The component is identified as `JTextField` in red text.
- Ciudad:** A text input field with the label "Ciudad" to its left. The component is identified as `JTextField` in red text.
- Provincia:** A text input field with the label "Provincia" to its left. The component is identified as `JTextField` in red text.
- Registrar usuario:** A button with the text "Registrar usuario". The component is identified as `JButton` in red text.
- JLabel de Error:** A label at the bottom of the window, currently empty, identified as `JLabel de Error` in red text.

Al pulsar en el botón registrar usuario se comprobará si las cajas están rellenas correctamente.

Si hay alguna caja vacía en el **JLabel** error dirá **"Todas las cajas son obligatorias"**

Si el año de nacimiento no está comprendido entre 1900 y 2019 en el **JLabel** error dirá **"Año de nacimiento erróneo"**

Si la ciudad No tiene los valores Barcelona, Hospitalet o Leganes en el **JLabel** dirá "Ciudad no válida". Solo permite las ciudades Barcelona, Hospitalet y Leganes ninguna más.

En la provincia, si la ciudad es Leganes solo se permite la provincia Madrid, si la ciudad es Barcelona u Hospitalet solo permite la provincia Barcelona. No admite ninguna más. Si no se cumplen estas condiciones se mostrará en el **JLabel** “Provincia no válida”

En caso de que todos los valores de las cajas de texto sean correctos se mostrará en el **JLabel** “Registro correcto”

2.- Viajes



Ciudad	Precio
Roma	<input type="text"/>
Berlin	<input type="text"/>
Tokio	<input type="text"/>
Lisboa	<input type="text"/>

Indicar mejor precio

Mejor Precio

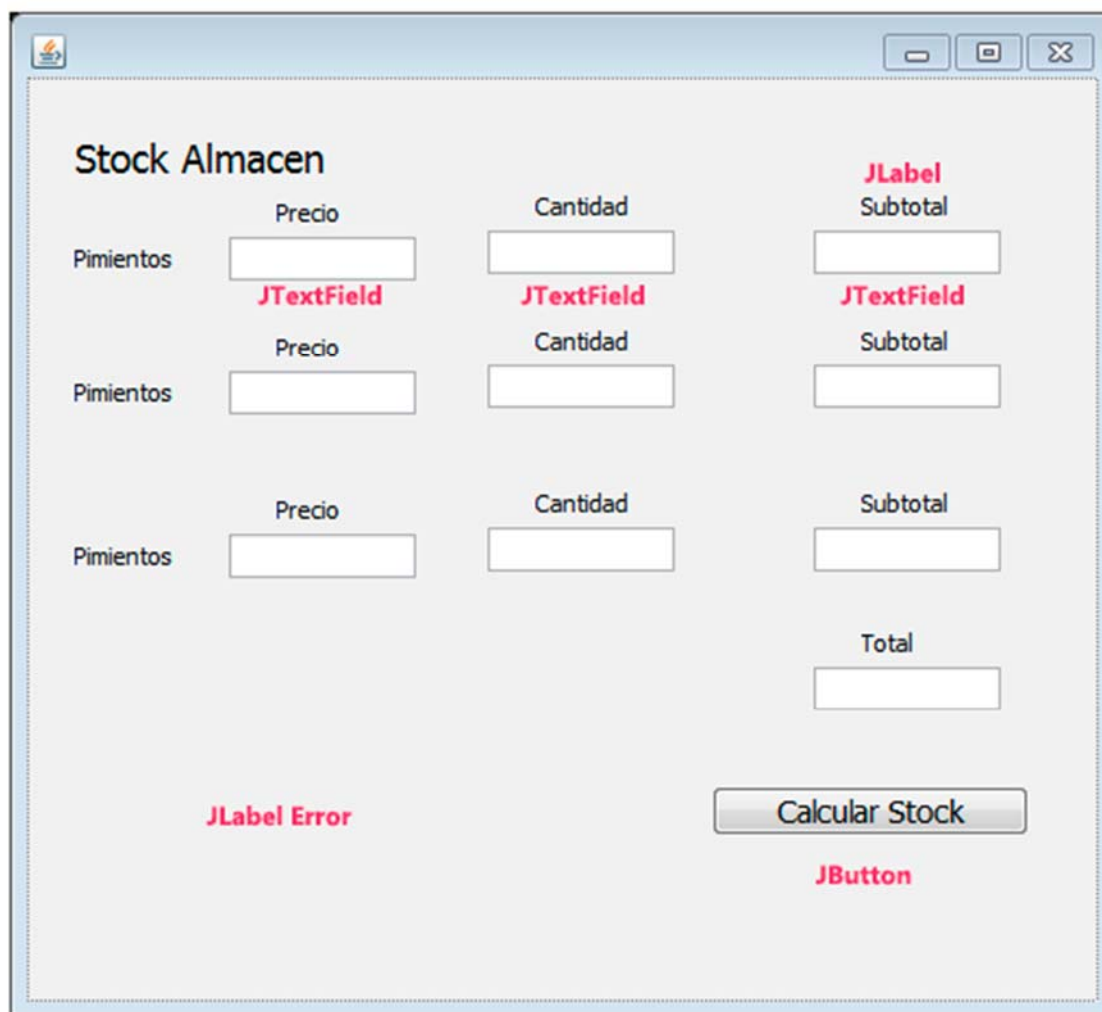
XXXXXXXXX

Existirán 4 cajas de texto, dentro de las cuales habrá un valor numérico. Existirá un botón que comprobará el estado de las cajas. Si al pulsar en el botón en cualquiera de las cajas existe algún valor NO numérico en el **JLabel** debajo de Mejor precio se mostrará el texto “**Formato no válido**”

Si al pulsar el botón todas las cajas tienen valor numérico se comprobará cual es la mayor de ellas y se escribirá en el **JLabel** que pone XXXXXXX el nombre de la ciudad más barata (con el menor precio)

Nota: No hay que comprobarlo, pero siempre en una de las cajas existirá un valor más pequeño que el resto, no se van a repetir nunca, es decir, dos cajas con el mismo valor.

3.- Stock Almacen



The screenshot shows a Java Swing window titled "Stock Almacen". The window contains three rows of input fields for calculating stock. Each row has a label "Pimientos" on the left, followed by two text boxes labeled "Precio" and "Cantidad" respectively. To the right of each row is a text box labeled "Subtotal". At the bottom right, there is a button labeled "Calcular Stock". At the bottom left, there is a label "JLabel Error". The text boxes are labeled "JTextField" and the button is labeled "JButton" in red text.

En todas las cajas de texto SÓLO se admitirán valores numéricos, si se pulsa el botón **Calcular Stock** y existe alguna caja de precio o cantidad vacía o con un valor no numérico en el **JLabel** Error se mostrará el texto **"Formato erróneo"**.

Cada línea tendrá el precio del producto y la cantidad del mismo. Al pulsar en el botón **Calcular stock** habrá que calcular el subtotal de cada uno de los productos, es decir multiplicando el precio por la cantidad. Una vez calculados los subtotales y ser mostrados

en las cajas de texto correspondiente, se realizará el cálculo del total que es la suma de todos los subtotales.