



## Cátedra Tecnología de Sistemas

Programación Intermedia

Código: 00824

### Tarea 2. Valor 4%

#### Temas de Estudio

1. Tema 1
2. Tema 2
3. Tema 3
4. Tema 4

#### Objetivo

Desarrollar una aplicación en Java que permita demostrar el conocimiento adquirido sobre manipulación de estructuras, manejo de archivos e interfaz gráfica de usuario.

#### Software de Desarrollo

Netbeans

#### Desarrollo

Como desarrollador en Java se le ha asignado la tarea de crear un programa para calcular los valores nutricionales de los alimentos que los pacientes de un centro de nutrición utilizan como referencia para incluir en sus dietas.

Para eso debe definir una clase llamada “Alimento” que, como mínimo, contenga los siguientes atributos:

- Código del alimento (String) este es el valor utilizado para identificar al alimento como único en la aplicación.
- Nombre del alimento (String)
- Cantidad de proteínas por cada 100 gramos (double)

- Cantidad de carbohidratos por cada 100 gramos (double)
- Cantidad de grasas por cada 100 gramos (double)
- Fecha de creación del alimento (date)

Además, se deben definir dos subclases que heredan de "Alimento":

- "AlimentoProcesado", esta debe agregar un atributo "Lista de ingredientes" (ArrayList<String>) que contenga la lista de ingredientes del alimento procesado.
- "AlimentoNatural", la cual debe agregar un atributo "Tipo de alimento" (String) que indique si se trata de una fruta, una verdura, un cereal, una carne, etc.
- El programa solo debe permitir crear objetos de tipo "AlimentoProcesado" y "AlimentoNatural", por lo que, no se puede crear objetos de tipo "Alimento".
- Cada alimento ingresado debe almacenarse en un archivo de texto con el nombre y extensión "Alimento.txt".
- La aplicación debe contar con una opción para modificar y eliminar alimentos, cuando elimina un alimento este se debe registrar en un archivo llamado "AlimentosEliminados.txt".
- El programa debe contener una opción donde se deben visualizar 2 reportes, 1) el usuario ingresa el código del alimento y este le muestra toda la información de este alimento; y 2) el usuario ingresa una fecha y debe mostrar la información de los productos creados en dicha fecha.
- La aplicación debe ser desarrollada con interfaz grafica de usuario (GUI por sus siglas en inglés).

#### **Validaciones importantes:**

- **No se pueden ingresar 2 productos con el mismo código.**
- **Cuando se solicita eliminar un producto debe verificarse que este ya exista el código registrado.**
- **Debe hacer uso de al menos**
- **Deben controlarse las excepciones.**

## Honestidad Académica



<https://audiovisuales.uned.ac.cr/play/player/23048>

### Nota Importante

Cada estudiante es responsable del contenido que entrega, si no es el archivo correcto, no podrá entregarlo posterior a la fecha establecida.

Si el contenido del archivo coincide con algún otro estudiante, o se comprueba que no es de su autoría, se aplicaría lo indicado en la plataforma en el documento [Lineamientos ante casos de plagio](#)

## Indicaciones Importantes

- Es obligatorio que incluya todo el directorio donde se encuentra el proyecto.
- El proyecto debe estar desarrollado en **Netbeans** que es la herramienta oficial del curso.
- El programa debe ser modular, utilizando de la mejor manera funciones definidas por usted.
- Los trabajos deben realizarse en forma individual. Dentro del código del programa debe de indicar la documentación que explique cómo fue realizado el programa.
- Si utiliza código de algún ejemplo del libro, o de otra fuente que no sea de su autoría, debe de indicarlo.
- Comprima todos los archivos en un solo archivo .zip o .rar.
- Nombre del archivo que envía: debe ser nombre y primer apellido del estudiante, y nombre de la tarea. **Ejemplo: JuanRojas-proyecto.**
- La entrega de la **tarea 1** en las fechas establecidas en la plataforma de aprendizaje en línea Moodle en el apartado que se indique.



Si no concluyó a tiempo la tarea, debe entregar lo que pudo hacer e incluir una carta explicando las razones por las cuales no finalizó.

### Rúbrica de Evaluación

Criterio	Cumple a satisfacción lo indicado en la evaluación	Cumple medianamente en lo indicado en la evaluación	Cumple en contenido y formato pero los aportes no son significantes	No cumple o no presenta lo solicitado
Presentación del código: Correcto uso de variables, nombres de métodos significativos, no presenta métodos vacíos.	5	3	2	0.1
Comentarios en código: No comete errores gramaticales u ortográficos dentro de los comentarios del código y son significativos para cada método y atributo	5	3	2	0.1
Creación correcta de la "Alimento"	10	7	4	0.1
Correcta creación de clase "AlimentoProcesado"	10	7	4	0.1
Correcta creación de clase "AlimentoNatural"	10	7	4	0.1
Correcta manipulación del archivo "Alimento.txt"	20	10	5	0.1
Correcta creación y registro del archivo "AlimentosEliminados.txt"	10	7	2	0.1
Uso correcto de una GUI (al menos 3 tipos de controles diferentes)	10	7	4	0.1
Reporte por código de alimento	10	10	5	0.1
Reporte por fecha de creación de alimento	10	10	5	0.1
<b>TOTAL</b>	<b>100</b>			