



PROGRAMMING PROYECTION PROGRAMACION

INTRODUCCIÓN

Nuestro proyecto consiste en una aplicación de consola en Java diseñada para gestionar tareas de forma práctica, clara y completamente local. El programa permite al usuario crear tareas, marcarlas como completadas, listarlas y administrar un flujo básico de organización personal, todo a través de un menú interactivo en consola.

A diferencia de versiones anteriores, este nuevo código se enfoca en una estructura más limpia, modular y entendible. Se eliminaron dependencias externas como Google Drive o archivos JSON y ahora el enfoque principal está en el manejo directo de tareas usando clases simples y un flujo de menú más intuitivo.

El objetivo fue crear un sistema funcional, sencillo de usar y fácil de mantener.



```
"container">
  <div class="row">
    <div class="col-md-6 col-lg-8"> <!-- B
      <nav id="nav" role="navigation">
        <ul>
          <li><a href="index.html">Home</a></li>
          <li><a href="home-events.html">Home E
          <li><a href="multi-col-menu.html">Mul
          <li class="has-children"> <a href="#">
            <ul>
              <li><a href="tall-button-head
              <li><a href="image-logo.html">
              <li class="active"><a href="ta
            </ul>
          </li>
          <li class="has-children"> <a href="#">
            <ul>
              <li><a href="variable-width-sl
            </ul>
          </li>
        </ul>
      </nav>
    </div>
  </div>
</div>
```

OBJETIVO GENERAL

Desarrollar un programa en Java que permita al usuario gestionar tareas mediante un menú interactivo, con una estructura organizada y código limpio.

OBJETIVOS ESPECÍFICOS

- Implementar una clase que represente una tarea con sus atributos esenciales.
- Desarrollar un sistema de almacenamiento en memoria para las tareas.
- Crear un menú funcional en consola para manejar la aplicación.
- Organizar el código en clases separadas para mejorar claridad y mantenimiento.
- Simplificar el flujo de ejecución para una experiencia más directa.



OBJETIVOS DEL PROYECTO

```
17 string sInput;
18 int iLength, iN;
19 double dblTemp;
20 bool again = true;
21
22 while (again) {
23     iN = -1;
24     again = false;
25     getline(cin, sInput);
26     system("cls");
27     stringstream(sInput) >> dblTemp;
28     iLength = sInput.length();
29     if (iLength < 4) {
30         again = true;
31         continue;
32     } else if (sInput[iLength - 3] != '.') {
33         again = true;
34         continue;
35     } while (++iN < iLength) {
36         if (!isdigit(sInput[iN])) {
37             again = true;
38             break;
39         }
40     }
41 }
```



FUNCIONAMIENTO DEL PROGRAMA

Al iniciar, el programa muestra un menú principal donde el usuario puede:

1. Crear una tarea nueva.
2. Mostrar la lista de tareas disponibles.
3. Marcar una tarea como completada.
4. Eliminar una tarea.
5. Salir del programa.

Cada opción ejecuta un método desde la clase principal que interactúa con la lista de tareas almacenada en memoria.

El funcionamiento se basa en el ciclo:
mostrar menú → leer opción → ejecutar acción
→ volver al menú

Las tareas se guardan temporalmente mientras la aplicación está en ejecución.

ESTRUCTURA DEL PROYECTO

TASKPROGRAMACION.JAVA

ESTE ARCHIVO CONCENTRA TODA LA FUNCIONALIDAD DEL PROYECTO:
INCLUYE:

- LA ESTRUCTURA DE UNA TAREA:
- SE DEFINEN ATRIBUTOS BÁSICOS COMO NOMBRE, DESCRIPCIÓN Y ESTADO.
- LA LISTA DE TAREAS:
- SE MANEJA DIRECTAMENTE DENTRO DEL MISMO ARCHIVO MEDIANTE ESTRUCTURAS COMO ARRAYLIST.
- EL MENÚ DEL USUARIO:
- EL PROGRAMA MUESTRA OPCIONES PARA:
 - CREAR TAREAS
 - LISTAR TAREAS
 - COMPLETAR TAREAS
 - ELIMINAR TAREAS
 - SALIR
- TODA LA LÓGICA DEL PROGRAMA:
- DESDE LAS FUNCIONES PARA MANIPULAR LAS TAREAS HASTA EL CICLO PRINCIPAL QUE EJECUTA EL MENÚ.





RETOS ENFRENTADOS

Uno de los principales retos fue organizar correctamente la separación de clases para lograr un código modular. Inicialmente, parte del código estaba repetido o mezclado con la lógica del menú.

Otro desafío fue implementar un sistema estable de eliminación y búsqueda de tareas mediante índices, evitando errores al seleccionar elementos inexistentes. También fue necesario mejorar el funcionamiento del menú para garantizar que el usuario interactuara sin generar fallos por entradas inválidas.

La estructura actual permitió simplificar el proyecto, hacerlo más entendible y evitar dependencias innecesarias.



CONCLUSIONES Y MEJORAS FUTURAS

Conclusiones

- Aprendimos a estructurar un programa modular utilizando Java.
- Se reforzó el uso de clases y métodos organizados para tareas específicas.
- El proyecto final es simple, funcional y fácil de entender.
- El equipo trabajó de forma colaborativa distribuyendo responsabilidades.

Mejoras Futuras

- Agregar almacenamiento permanente usando archivos o una base de datos.
- Crear una interfaz gráfica en JavaFX.
- Implementar categorías para organizar las tareas.
- Añadir búsqueda avanzada y filtros.
- Exportar el listado de tareas a PDF o Excel.



MUCHAS GRACIAS
THANK YOU