

Apellido y nombre: _____ Legajo: _____

RECUPERATORIO DE PRIMER EXAMEN PARCIAL

14/07/2023

Resolvé el siguiente ejercicio utilizando el lenguaje **Java** y el paradigma orientado a objetos. Lee el enunciado al menos dos veces antes de intentar confeccionar la solución.

ENUNCIADO

Una empresa de software que se dedica a crear componentes de interfaz gráfica de usuario (GUI) nos pide desarrollar la lógica general del sistema de menús que utilizará para las apps de consola.

A continuación, se muestra un posible sistema de menús para un micro bloc de notas por consola:

```
-- uNotePad --

1 - Nuevo
2 - Abrir
3 - Guardar
4 - Modo >
X - Salir

Ingrese la opción deseada:
```

Menú Principal

```
-- uNotePad/Modo --

1 - Claro
2 - Oscuro
3 - Otro >
X - Salir

Ingrese la opción deseada:
```

Menú "Modo"

```
-- uNotePad/Modo/Otro --

1 - Desde archivo...
2 - Desde URL...
X - Salir

Ingrese la opción deseada:
```

Menú "Otro"

Como se ve, cada menú tendrá un título y uno o más ítems de menú, por ahora, de dos tipos:

- Ítem de acción: Permite ejecutar una acción (por ejemplo, "Nuevo", "Abrir", "Guardar", "Imprimir", "Ayuda", etc)
- Ítem de acceso a submenú: Permite acceder a un menú secundario (por ejemplo, "Archivos recientes", "Toolbars", "Modo", etc) que poseerá como atributo.

Además, cada menú debe guardar sí o sí, de forma independiente, un ítem que permita salir de él.

Todos los tipos de ítems de menú deben poseer una descripción y un carácter asociado, para que el usuario pueda elegirlo. Los ítems que permiten salir de un menú siempre tendrán la descripción "Salir" y el carácter asociado será la 'X'.

Tanto los menús como sus ítems son mostrables y ejecutables. Cuando se ejecuta un menú, éste debe mostrar todos sus ítems y, luego, su ítem de salida. Por último, se debe mostrar una leyenda que invite al usuario a ingresar una opción*. Esto se repetirá hasta que el usuario seleccione que quiere salir. Si la entrada es inválida, debe ignorarse. En caso de coincidir con algún ítem, este deberá ejecutarse:

- Los ítems de acción por ahora solo mostrarán la leyenda "PENDIENTE".
- Los ítems de acceso a menú secundario simplemente ejecutarán tal menú.
- Los ítems de salida por ahora no hacen nada (su implementación no lleva código).

* Para leer un **char** con Java, utilizá el método **nextLine()** de la clase **Scanner**. Esto retorna un **String**. Para quedarse con el primer caracter, utilizá el método **charAt(0)** de la clase **String**.

A partir del modelo enunciado, se pide:

- 1) Desarrollar el método **agregarItem()** que permita agregar un ítem a un menú (que no sea de salida).
- 2) Desarrollar los métodos **ejecutar()** y **mostrar()** de cada clase, más los métodos derivados necesarios.
- 3) Desarrollar un test en el método **main** del proyecto que permita recrear el menú ejemplificado anteriormente (micro bloc de notas por consola).

FORMATO DE ENTREGA

Archivo **.zip** o **.rar** nombrado de la siguiente manera (reemplazando lo que está en verde):

INSPT-ProgII-p1Recu-2023-SistemaDeMenus-TUAPELLIDO-TUNOMBRE

El **.zip** o **.rar** debe contener el proyecto Java, el cual debe tener la misma nomenclatura.

No cumplir con alguna indicación del formato de entrega disminuye la nota del examen.

CRITERIOS DE EVALUACIÓN

Para considerar aprobado el examen, el mismo debe demostrar la correcta aplicación de los siguientes conceptos de la programación orientada a objetos y el lenguaje Java:

- Definición de clases y asignación adecuada de sus responsabilidades.
- Encapsulamiento, ocultando detalles de implementación y utilizando métodos getters y setters sólo cuando corresponda.
- Modularización reutilizable y mantenible, usando funciones con correcta parametrización, aplicando alta cohesión y bajo acoplamiento.
- Correcta implementación de los constructores.
- Validación de los datos que ingresan al sistema.
- Aplicación de herencia y polimorfismo, incluyendo interfaces.
- Aplicación de los principios S.O.L.I.D
- Algoritmos precisos, finitos y eficientes.
- Buenas prácticas de programación:
 - Uso de constantes en lugar de números mágicos.
 - Nombres de variables, métodos y clases descriptivos, que sigan la convención del lenguaje.
 - Uso adecuado de los ciclos. No alterar la naturaleza de los mismos utilizando, por ejemplo, instrucciones de control como **break** y **continue**.
 - Código ordenado e indentado correctamente.

% Correcto	0 a 20	25 a 45	50 a 55	60	65 a 70	75	80	85 a 90	95	100
Nota	1	2	¿4?	4	5	6	7	8	9	10

(Escala solo con fines orientativos)