



UNIVERSIDAD AUTÓNOMA “TOMÁS FRÍAS” INGENIERÍA DE SISTEMAS				
ESTUDIANTE:			C.I.:	
DOCENTE:	Ing. Ditmar Castro Angulo	MATERIA:	SIS-211 G2	PRÁCTICA N5 AUX
AUXILIAR:	Univ. Gabriel Alejandro Garvizu Salas	FECHA:	02/11/2024	

Ejercicio 1: Sistema de Notificaciones de Eventos en un Juego

- 1. Crea una interfaz funcional llamada “EventoNotificacion” con un método **notificar(String mensaje)**.
- 2. Crea una clase “Evento” que tenga un atributo **nombre** y un método **registrarNotificacion(EventoNotificacion notificacion)**. Este método almacenará la notificación.
- 3. Agrega un método **activar()** que llamará al método **notificar** de todas las notificaciones registradas, pasando un mensaje que incluya el nombre del evento.
- 4. Crea una clase principal para demostrar el uso de la interfaz funcional registrando y activando diferentes

Ejercicio 2: Sistema de Bonificaciones en un Juego

- 1. Crea una interfaz funcional llamada “CalculadoraBonificacion” con un método **calcular(int puntosBase)** que devuelva un “int”.
- 2. Crea una clase “Jugador” con atributos **nombre** y **puntos** y un método **aplicarBonificacion(CalculadoraBonificacion calculadora)** que aplique la bonificación y actualice los puntos del jugador.
- 3. Crea una clase principal donde se creen instancias de “Jugador” y se apliquen diferentes bonificaciones usando expresiones lambda.

Ejercicio 3: Sistema de Tareas en un Juego de Aventura

- 1. Crea una interfaz funcional llamada “Tarea” con un método **ejecutar()** que no devuelva nada.
- 2. Crea una clase “Aventura” con un método **agregarTarea(Tarea tarea)** que almacene la tarea en una lista y un método **iniciar()** que ejecute todas las tareas almacenadas.
- 3. Crea una clase principal donde se agreguen y ejecuten diferentes tareas usando expresiones lambda.

Ejercicio 4: Sistema de Filtros de Objetos en un Juego

- 1. Crea una interfaz funcional llamada “FiltroObjeto” con un método **filtrar(Objeto objeto)** que devuelva un booleano.
- 2. Crea una clase “Objeto” con atributos **nombre** y **peso**.
- 3. Crea una clase “Inventario” que tenga una lista de objetos y un método **filtrarObjetos(FiltroObjeto filtro)** que devuelva una lista de objetos que cumplen con el filtro.
- 4. Crea una clase principal donde se filtren objetos del inventario usando diferentes criterios con expresiones lambda.

Ejercicio 5

Una vez completados los 4 ejercicios anteriores, es hora de realizar el ejercicio que vale más puntaje, hasta se podría decir que este ejercicio vale 100% de la práctica, puesto que si la carpeta con los distintos archivos no es subida a su repo de GitHub no podrá calificarse. Tal vez ya te hayas percatado de que va, en caso de que no, ahí va.

Deberá de subirse los distintos archivos que se desarrollaron a lo largo de la práctica, a su repo en GitHub para su revisión, caso contrario no será válida la entrega por ningún otro medio.

Buena suerte y nos vemos en la siguiente práctica.