

INSTITUTO POLITECNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO (ESCOM)

PROGRAMACIÓN ORIENTADA A OBJETOS

NOMBRE DEL ALUMNO:

* SANTOS MÉNDEZ ULISES JESÚS

PRÁCTICA:

* HILOS

NÚMERO DE PRÁCTICA: 4

OPCIÓN:

* MARCO DIGITAL

FECHA DE ENTREGA:

* 18/06/2021

GRUPO:

* 2CM11

**Hilos**

**Introducción**

Se desarrollo un Marco Digital con 8 imágenes diferentes sobre diversos monstruos mitológicos y de historias de libros, a continuación, se verá el uso de los hilos para saber si existe o no la imagen haciendo uso del try y catch para el manejo de excepciones.

**Marco Teórico**

**Hilos**

Un programa multihilo contiene dos o más partes que pueden ejecutarse de forma concurrente. Cada parte de ese programa se llama hilo (Thread) y cada hilo establece un camino de ejecución independiente.

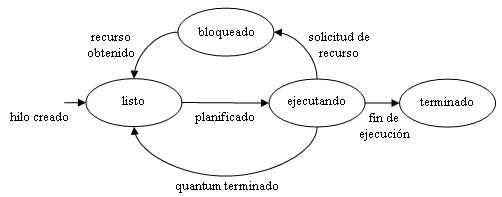
Es un tipo de multitarea distinta a la multitarea por procesos, la cual ejecuta varios programas a la vez. En Java, es un solo programa con varios hilos a la vez.

La programación multihilo permite escribir programas muy eficientes que utilizan al máximo la CPU, reduciendo al mínimo el tiempo que está libre. Esto es importante en los entornos interactivos y en red en los que se trabaja con Java.

Los hilos pueden estar en distintos estados. Un hilo puede estar ejecutándose. También puede estar preparado para ejecutarse tan pronto como disponga de tiempo de CPU. Un hilo que está ejecutándose para suspenderse, lo que equivale a detener temporalmente su actividad.

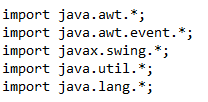
El sistema multihilo de Java está construido en torno a la clase **Thread,** sus métodos y su interfaz de apoyo **Runnable.** La clase Thread involucra la noción hilo de ejecución. Para crear un hilo, se debe transferir un objeto Runnable (objeto que instrumenta la interfaz Runnable al definir un método run ()) al constructor Thread para que defina su propio método run ().

El método run() del Thread o el objeto Runnnable especificados en el cuerpo del hilo empieza a ejecutarse cuando se llama al método **start()** del objeto Thread. El hilo funciona hasta que regresa el método run() o hasta que se llama al método **stop()** del objeto Thread.



**Desarrollo**

1. Se importaron las siguientes librerías de Java para el diseño del Marco Digital y para el uso del JFrame y del ComboBox, en ella están contenidas las clases y métodos que se utilizarán



1. Se declara la clase pública MarcoDig que implementa la interfaz Runnable y ActionListener



1. Se establecen las variables de instancia para general el ComboBox y las cadenas que se insertaran en el ComboBox, un panel, etiquetas y la variable que es el Hilo de control

Texto

Descripción generada automáticamente

1. En el constructor del programa ponemos los parámetros para inicializar las variables de instancia, tanto del panel así como de las imágenes y su inserción al arreglo de las mismas, recordando que las dos últimas instrucciones del constructor siempre van al último, una reserva memoria para el hilo y la otra llama al constructor con new.

Texto

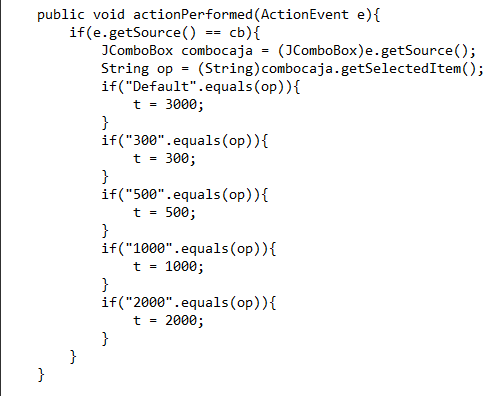
Descripción generada automáticamente

1. Se manda a llamar al método Arrimages que es en donde se agregaron las imágenes

Texto

Descripción generada automáticamente

1. Se tiene el método actionPerformed que será el encargado de extraer datos debido a la interacción del usuario con el ComboBox



1. Se establece el cuerpo del hilo con el método run en ella se hace uso del try y catch, se pidió que se hiciera uso del operador modulo.

Texto, Carta

Descripción generada automáticamente

Logotipo

Descripción generada automáticamente

1. Se tiene finalmente el main que es el que nos ayudará a poder tener la clase principal para así ejecutar nuestro programa

Texto

Descripción generada automáticamente

1. Se compila y se ejecuta el programa MarcoDig.java

Texto

Descripción generada automáticamente

Imagen que contiene Sitio web

Descripción generada automáticamente

Imagen que contiene foto, diferente, firmar, espejo

Descripción generada automáticamente

**Conclusión**

En conclusión, esta práctica me fue útil para entender mejor el concepto de hilo y como manipularlo para mejorar la eficiencia del programa y de la misma forma aplicable para el manejo de errores o excepciones que se puedan encontrar en el programa.