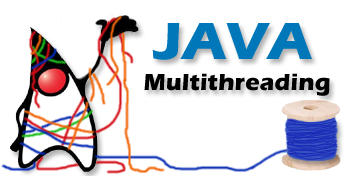
# Manejo de Hilos (Threads) en java.



Java es un lenguaje de programación multihilo que significa que podemos desarrollar un programa de subprocesos múltiples . Un programa de subprocesos múltiples contiene dos o más partes que pueden ejecutarse al mismo tiempo y cada parte puede manejar una tarea diferente al mismo tiempo, haciendo un uso óptimo de los recursos disponibles, especialmente cuando su computadora tiene varias CPU.

Por definición, la multitarea es cuando múltiples procesos comparten recursos de procesamiento comunes, como una CPU. Multi-threading amplía la idea de la multitarea a aplicaciones en las que puede subdividir operaciones específicas en una sola aplicación en hilos individuales. Cada uno de los hilos se puede ejecutar en paralelo. El sistema operativo divide el tiempo de procesamiento no solo entre diferentes aplicaciones, sino también entre cada subproceso dentro de una aplicación.

# Ciclo de vida.

Comienza el ciclo de vida y permanece acá hasta que el programa inicie el hilo

**New**

un hilo ejecutable ingresa al estado terminado cuando completa su tarea o termina de otra manera.

**Terminated (Dead)**

**Timed Waiting**

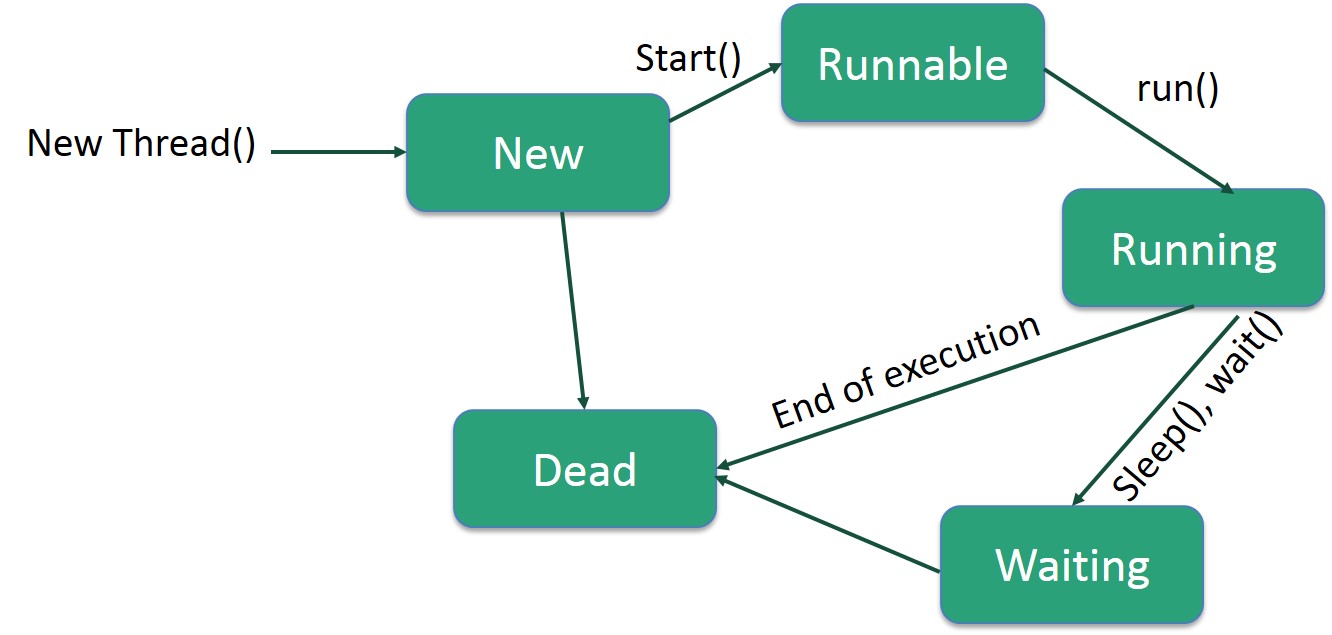
un hilo ingresar a este estado de espera durante un intervalo de tiempo específico. Un hilo en este estado pasa al ejecutarse cuando ese intervalo de tiempo expira o cuando ocurre el evento que está esperando.

El hilo espera mientras otro hilo esta realizando una tarea, pasa al estado ejecutable, cuando otro hilo espera para continuar la ejecución.

**Waiting**

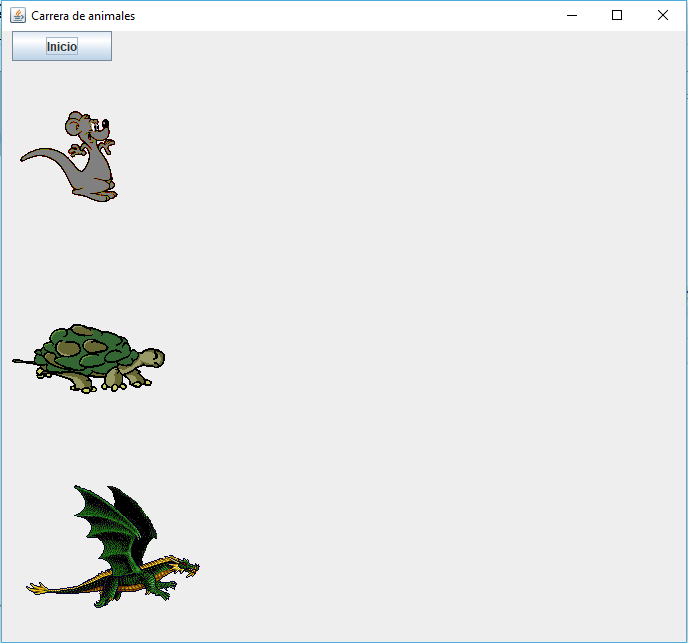
Iniciado el hilo, este se vuelve a ejecutar. Lo que significa que esta realizando su tarea.

**Runnable**



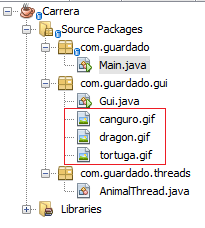
**Fuente: https://www.tutorialspoint.com/java/java\_multithreading.htm**

# Practica guiada



La practica consiste en simular una carrera de animales usando hilos e Interfaz Gráfica. La carrera comenzara cuando le demos click al botón Inicio.

1. Crearemos un proyecto el cual contendrá la siguiente estructura de paquetes:



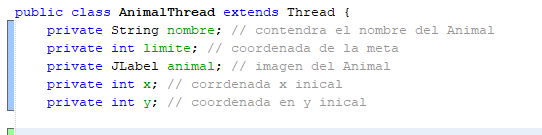
Imagenes que usaremos.

Clase que manejara nuestros hilos

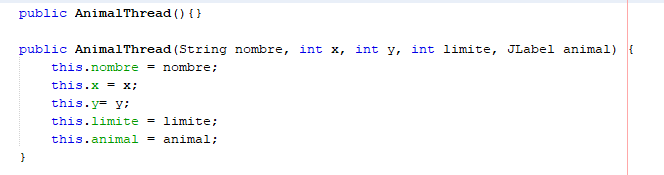
1. Creación de nuestros hilos (Clase AnimalThread.java).

Nos permitirá utilizar el método run()

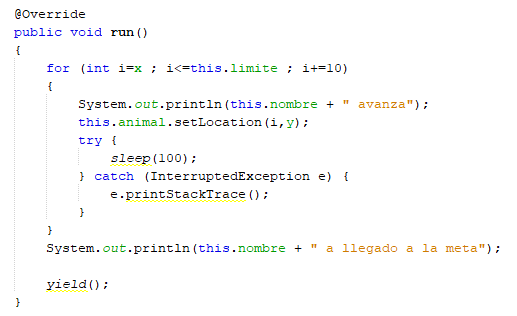
* Declaramos los atributos que vamos a utilizar.



* Crearemos dos constructores.



* Realizaremos la lógica para simular la carrera, esto se estará ejecutando gracias al método run(), el cual sobrescribimos de la siguiente forma.



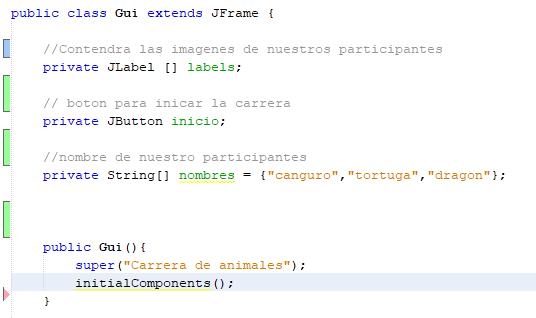
Simulamos el movimiento

Hace que el hilo se detenga por un cierto tiempo. El valor que recibe es en mili segundos

Permite ceder el paso a otro hilos que tengan la misma prioridad que el

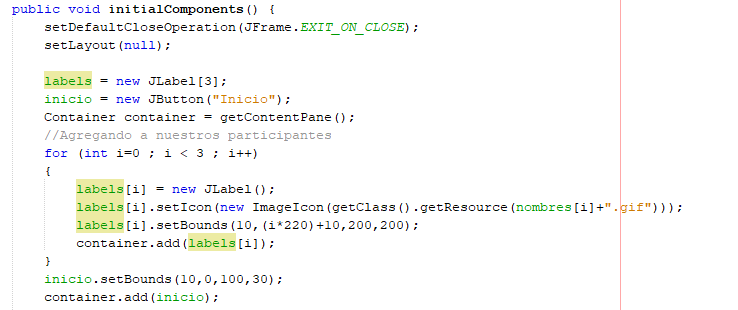
1. Diseño de interfaz Gráfica (Clase Gui.java) e implementación de la clase AnimalThread.java.

* Atributos y constructores



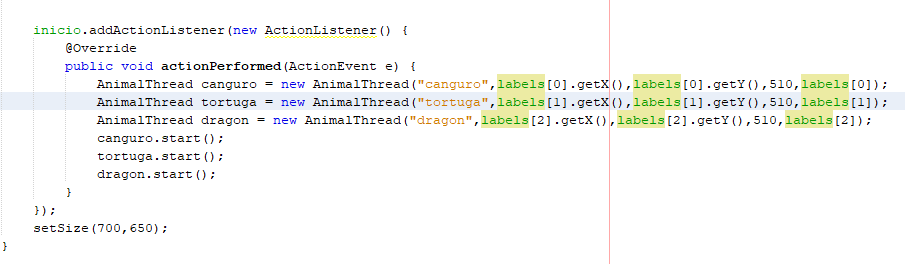
Titulo de la ventana

* Creación del método initialComponents.



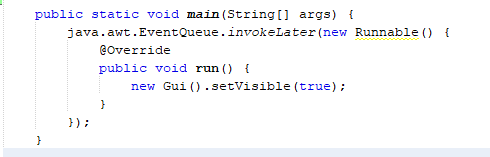
Creamos nuestros hilos.

Agregamos una Imagen al Label



Con el método .start(), ejecutamos nuestros hilos.

* Creación del método main dentro de nuestra clase Gui.



1. Levantando nuestra ventana en la clase Main.java.

