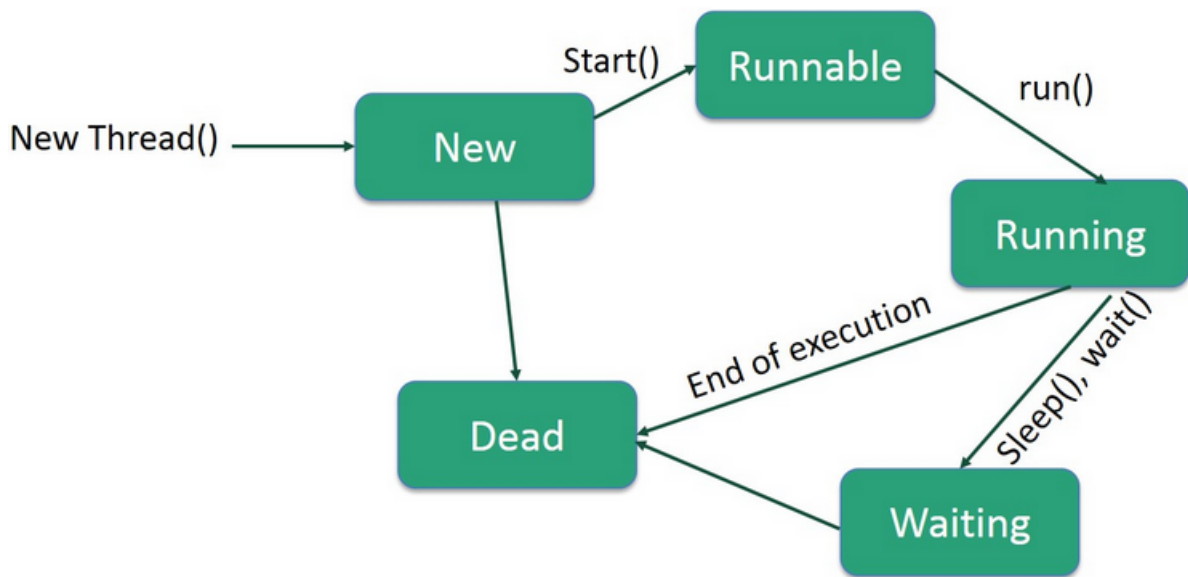




- D'abord il faut déclarer une classe pour lancer les nouvelles tâches (threads, représenter par une classe), cette tâche doit soit :
 - implémenter l'interface Runnable, dans ce cas on instancie la classe comme suit :
 - Classe nom = new Thread(new classe());
 - La classe Thread a un créateur qui prend une instance implémentant Runnable en argument.
 - hériter la classe Thread
- la nouvelle tâche (classe) doit définir la méthode run (public void run())
 - ceci est obligatoire si la tâche implémente Runnable
 - il ne l'est pas si la tâche hérite de la classe Thread, car ce dernier contient déjà la méthode run mais vide (la tâche créée ne fera rien)
- Thread.sleep() est une méthode statique permettant de mettre la tâche actuelle en veille.
 - cette méthode déclenche une exception (InterruptedException) il est obligatoire de l'intercepter.
- une méthode « synchronized » est une méthode qui ne peut pas être utilisée que par un thread à la fois (les threads doivent utiliser la même instance de la classe qui contient la méthode, sinon ça n'a aucune utilité !)
- dans le main on peut suspendre/repandre un thread nomThread.interrupt() et nomThread.continue(), il faut dans ce cas aussi intercepter l'exception InterruptedException
- on peut aussi attendre la fin d'un thread en utilisant la méthode nomThreadAattendre.join() dans le thread qui doit attendre (idem pour l'exception InterruptedException)
- threadname.stop() permet d'arrêter un thread depuis un autre