Thread LifeCycle :

\*

\* 1. New Thread which is Created

\*

\* 2. Runnable : Thread ready to run or about to run

\*

\* 3. Running : Calling the start(); method

\*

\* 4. Blocked/Paused : It stops Running for some time after star()

execution and eventually it will go into running

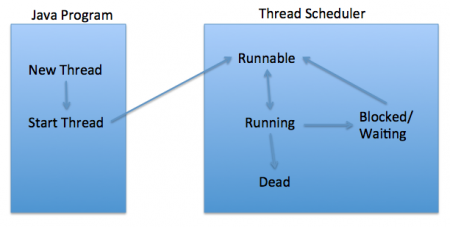
\*

\* 5. Dead/End : Thread executed the methods inside it

\*

Thread Life Cycle in Java

Below diagram shows different states of thread life cycle in java. We can create a thread in java and start it but how the thread states change from Runnable to Running to Blocked depends on the OS implementation of thread scheduler and java doesn’t have full control on that.

[](https://www.journaldev.com/wp-content/uploads/2012/12/Thread-Lifecycle-States.png)

New

When we create a new Thread object using new operator, thread state is New Thread. At this point, thread is not alive and it’s a state internal to Java programming.

Runnable

When we call start() function on Thread object, it’s state is changed to Runnable. The control is given to Thread scheduler to finish it’s execution. Whether to run this thread instantly or keep it in runnable thread pool before running, depends on the OS implementation of thread scheduler.

Running

When thread is executing, it’s state is changed to Running. Thread scheduler picks one of the thread from the runnable thread pool and change it’s state to Running. Then CPU starts executing this thread. A thread can change state to Runnable, Dead or Blocked from running state depends on time slicing, thread completion of run() method or waiting for some resources.

Blocked/Waiting

A thread can be waiting for other thread to finish using [thread join](https://www.journaldev.com/1024/java-thread-join-example) or it can be waiting for some resources to available. For example [producer consumer problem](https://www.journaldev.com/1034/java-blockingqueue-example) or [waiter notifier implementation](https://www.journaldev.com/1037/java-thread-wait-notify-and-notifyall-example) or IO resources, then it’s state is changed to Waiting. Once the thread wait state is over, it’s state is changed to Runnable and it’s moved back to runnable thread pool.

Dead

Once the thread finished executing, it’s state is changed to Dead and it’s considered to be not alive.

Above are the different states of thread. It’s good to know them and how thread changes it’s state. That’s all for thread life cycle in java.