Java threads: Thread states

## A thread in Java can find itself in any of the 6 following states:

* NEW : Thread that was just created, but hasn’t been started yet.
* RUNNABLE : Thread that is executing on the virtual machine.
* BLOCKED : Thread waiting for the liberation of a lock.
* WAITING : Thread waiting for another thread to execute some procedure.
* TIME\_WAITING : Thread waiting for another thread to execute some procedure or a timer to run out.
* TERMINATED : Thread terminated.

*T = new Thread()*

*T.wait()*

*T.start()*

*T.notify() ou*

*T.notifyAll()*

*synchronized*

obtaining the clock

*T.wait(n)*

end of thread

*T.notify()*

*ou T.notifyAll()*

Here is an example program that shows the 3 main states of a thread:

**public** **class** Test {

**public** **static** **void** main(String[] args) **throws** InterruptedException {

Thread aThread = **new** Thread(); // creating thread => state NEW

System.***out***.println(aThread.getState());

aThread.start(); // starting thread => state RUNNABLE

System.***out***.println(aThread.getState());

aThread.join(); // wait for the thread to terminate

System.***out***.println(aThread.getState()); // thread is terminated => state TERMINATED

}

}

Write some code that demonstrates the 3 other possible states of a thread.