**Thread Group**

A thread group represents a set of threads. In addition, a thread group can also include other thread groups. The thread groups form a tree in which every thread group except the initial thread group has a parent.

A thread is allowed to access information about its own thread group, but not to access information about its thread group's parent thread group or any other thread groups.

**Join Method**

If a join method is invoked on a thread that thread will wait until the other thread completes execution

**Demon Thread**

It is used to support non demon thread(main thread).

We can create demon thread by using setDeamon() method

**Need Of LockApI**

* Implementations of the Lock interface enable the use of such techniques by allowing a lock to be acquired and released in different scopes, and allowing multiple locks to be acquired and released in any order.
* Guarantee of order

**Lock Interface**

* Lock interface is used to as a thread synchronization mechanism similar to synchronized blocks. New Locking mechanism is more flexible and provides more options than a synchronized block

**Difference Between Synchronized Block And Lock ApI**

* **Guarantee of sequence** − Synchronized block does not provide any guarantee of sequence in which waiting thread will be given access. Lock interface handles it.
* **No timeout** − **Synchronized block has no option of time**out if lock is not granted. Lock interface provides such option.
* **Single method** − Synchronized block must be fully contained within a single method whereas a lock interface's methods lock() and unlock() can be called in different methods.