UF 2: Multi-thread programming

Maribel Madueño

Thread in Java

❶ Thread Class

❷ Runnable Interface

❸ Object Class

❹ ThreadGroup Class

Thread Class

http://docs.oracle.com/javase/7/docs/api/java/lang/Thread.html

Thread constructors

- Thread()
- Thread(Runnable t)
- Thread(Runnable t, String name)
- Thread(String name)

Thread constructors

- Thread(ThreadGroup g, Runnable t)
- Thread(ThreadGroup group, String name)
- Thread(ThreadGroup g, Runnable t, String name)
- Thread(ThreadGroup g, Runnable t, String name, long stackSize)

Thread methods

- void setName(String name):
  Assigns a name to the thread
- String getName():
  Returns the thread name
- void start():
  The thread is set to the **ready** state.
- void run():
  Initiates the execution of the thread.

start() causes its call.

Do not invoke this method

Thread methods

- void join(): wait for this thread to die.
- void join(long t): Waits at most *t* ms for this thread to die.
- static void sleep(long t):
  The thread is set to the **sleep** state t ms
- static void yield():
  The thread is set to the **ready** state

https://docs.oracle.com/javase/tutorial/essential/concurrency/sleep.html
https://docs.oracle.com/javase/tutorial/essential/concurrency/join.html

Thread methods

- Boolean isAlive():

  It can be used to test if a thread has been started but not terminated.

- void setDaemon():

  Set the thread as a daemon. A daemon ends when its creator ends.

- Boolean isDaemon():

  It is used to test if a thread is a daemon

Thread methods

- void setPriority(int p):
  Set the execution priority of the thread
- int getPriority():
  Returns the thread priority
- static Thread currentThread():
  Returns the executed thread

Thread methods

- ThreadGroup getThreadGroup():
  Returns the thread assigned group

- String toString():
  Returns a string containing name, priority and group.

Runnable Interface

http://docs.oracle.com/javase/7/docs/api/java/lang/Runnable.html

Runnable methods

- void run():

When an object implementing interface Runnable is used to create a thread, starting the thread causes the object's run method to be called in that separately executing thread.

Object Class

http://docs.oracle.com/javase/7/docs/api/java/lang/Object.html

Object methods

- void wait()

  sets a thread in the **waiting** state
- void notify()

  sets a waiting thread in the **ready** state
- void notifyAll()

  sets all the waiting threads in the **ready** state

Can only be invoked in synchronized code

Object methods

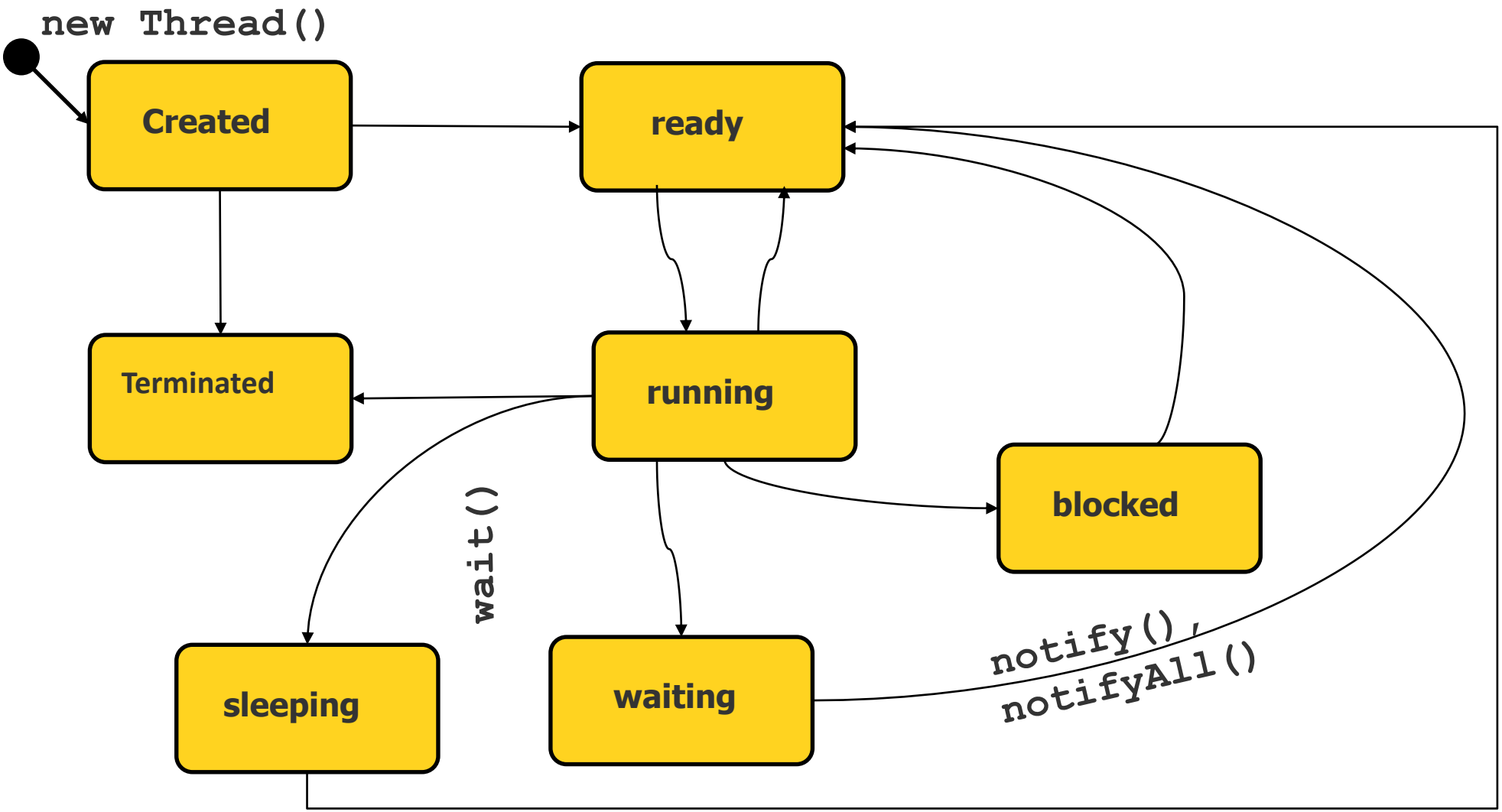
Synchronized method

```
synchronized public void  syncMethod(){
//...
}
```

Synchronized block code

```
synchronized (this)
{
//....
}
```

Object methods



ThreadGroup Class

http://docs.oracle.com/javase/7/docs/api/java/lang/Thread.html

ThreadGroup constructors

- ThreadGroup(String name)
- ThreadGroup(ThreadGroup parent, String name)

ThreadGroup assigment

```
ThreadGroup sgr = new ThreadGroup ("gName");
Thread fil = new Thread (sgr, "tName");
```

Additional threads created by a thread belong to the parent group.