Java Threads

- Resources
  - Java Threads by Scott Oaks & Henry Wong (O' Reilly)
  - API docs
    - http://download.oracle.com/javase/6/docs/api/
      - java.lang.Thread, java.lang.Runnable
      - java.lang.Object, java.util.concurrent
  - Tutorials
    - http://download.oracle.com/javase/tutorial/essential/concurrency/index.html
    - http://download.oracle.com/javase/tutorial/essential/concurrency/procthread.html
  - Introduction to Java Threads
    - http://www.javaworld.com/javaworld/jw-04-1996/jw-04-threads.html
  - Thread safety
    - http://en.wikipedia.org/wiki/Thread-safety
    - http://www.javaworld.com/jw-08-1998/jw-08-techniques.html

Coverage

- Thread class
  - run, start methods
  - yield, join
  - sleep
- Synchronization
  - synchronized methods & objects
  - wait/notify/notifyAll
  - conditions

java.lang.Thread

- Two techniques to create threads in java
- 1) implementing the Runnable interface
  - The Runnable interface should be ***implemented*** by any class whose instances are intended to be executed by a thread. The class must define a method, called *run*, with no arguments.
  - invoke Thread constructor with an instance of this Runnable class
  - See pages 162 and 164 in text for an example
- 2) extending Thread
  - Define a subclass of java.lang.Thread
    - Define a *run* method
  - In another thread (e.g., the main), create an instance of the Thread subclass
    - Then, call *start* method of that instance

Example 1

- Create 2 threads from the Main, then start them
- Threads will be instances of different thread sub-classes

```java
class MyThreadA extends Thread {
   public void run() { // entry point for thread
        for (;;) {
                System.out.println("hello world1");
        }
   }
}

class MyThreadB extends Thread {
   public void run() { // entry point for thread
        for (;;) {
                System.out.println("hello world2");
        }
   }
}

public class Main1 {
   public static void main(String [] args) {
        MyThreadA t1 = new MyThreadA();
        MyThreadB t2 = new MyThreadB();
        t1.start();
        t2.start();
        // main terminates, but in Java the other threads keep running
        // and hence Java program continues running
   }
}
```