Computer Science 22: Object Oriented Programming

Lecture #16: Concurrency II

In This Lecture

- Commonly used Thread methods
- Constructing Threads
  - Implementing Runnable Interface
  - Extending Thread Class
- Demo: Multiple Threads
- On Synchronization
- Demo: Thread Synchronization

Thread: Commonly Used Methods

Method	Description
getName()	Returns the name of the thread
getPriority()	Returns the priority of the thread
isAlive()	Determines whether the thread is running
join()	Pauses the thread until the terminates
run()	The entry point into the thread
sleep()	Suspends the thread; enables you to specify the period the thread is suspended
start()	Starts the thread

Constructing Threads

- Your class should either extend the **`Thread`** class or implement the **`Runnable`** interface.

Extending Thread Class

```
public class MyThread extends Thread {

    public MyThread(String threadName){
        super (threadName);
        this.start();  // you can also explicitly call start()
                       // in main()
    }

    public void run() {
       /* what will your thread do while it’s running? */
    }

}
```

Implementing Runnable Interface

```
public class MyThread implements Runnable {
    Thread t;
    public MyThread(String threadName){
        t = new Thread(this, threadName);
        t.start();  // you can also explicitly call start()
                    // in main()
    }

    public void run() {
        /* what will your thread do while it’s running? */
    }

}
```

Constructing Threads

- As a rule of thumb, you should implement the Runnable interface if the run() method is the only method of the Thread class you need to override. Otherwise, extend the Thread class itself and override the methods you want to override.

Demo

MULTIPLE THREADS

On Synchronization

- A major concern in multithreading when two or more threads share the same resource is that only one of them can access the resource at one time.
- You can use the keyword **`synchronized`** to limit the access to a resource to just one thread.

Demo

THREAD SYNCHRONIZATION