Übung 2 - Threads

Schreiben Sie eine Klasse Worker, mit dem einem Attribut protected String name und einem Attribut protected boolean shouldRun. Im Konstruktor wird der Name des Workers übernommen, sowie shouldRun auf true gesetzt. Die Klasse hat eine abstrakte Methode protected void work(). Erstellen Sie eine Methode protected void printStarted(), welche auf der Konsole den Namen des Workers, sowie „wurde gestartet“ ausgibt. Erstellen Sie eine weitere Methode protected void printStopped(), welche auf der Konsole den Namen des Workers, sowie „wurde gestoppt“ ausgibt. Schreiben Sie eine öffentliche Methode public void stopWorker(), welche shouldRun auf false setzt.

Erstellen Sie eine konkrete Implementierung des Workers mit dem Namen TimePrintWorker und implementieren Sie das Interface Runnable. In der Methode work() soll nun solange shouldRun auf true gesetzt ist, jede Sekunde die aktuelle Uhrzeit ausgegeben werden. Rufen Sie zu Beginn und Ende von work() jeweils die Methoden printStarted() und printStopped() auf. Rufen Sie die Methode work() in der Methode run() auf.

Die aktuelle Uhrzeit können Sie mit folgenden Codezeilen ausgeben:

Date d = new Date(); System.out.println(d.toString());

Erstellen Sie eine weitere konkrete Implementierung des Workers mit dem Name FileReadWorker und implementieren Sie das Interface Runnable. Übernehmen Sie zusätzlich im Konstruktor einen Wert für das neue Attribut private String path, welches den Pfad zu einer Textdatei abbildet. Erstellen Sie ein weiteres Attribut public ArrayList<String> lines, in welcher die eingelesen Textzeilen gespeichert werden sollen. Laden Sie in der Methode work() die angegebene Textdatei und speichern Sie die eingelesenen Textzeilen in der ArrayList. Überprüfen Sie auch nach jedem Lesevorgang, ob shouldRun noch immer true ist und brechen Sie gegebenenfalls ab. Für Tests können Sie den Lesevorgang auch verzögern, in dem Sie ein Thread.sleep() einbauen. Rufen Sie zu Beginn und Ende von work() jeweils die Methoden printStarted() und printStopped() auf. Rufen Sie die Methode work() in der Methode run() auf.

Schreiben Sie eine Demo-Anwendung, in welcher Sie zwei Threads erzeugen. Einer nimmt einen TimePrintWorker auf, einer einen FileReadWorker. Lassen Sie beide Threads gleichzeitig laufen und beenden Sie den TimeReadWorker-Thread, sobald die Textdatei eingelesen wurde (Tipp: Warten Sie, bis der FileReadWorker beendet ist und bitten Sie im Anschluss den TimePrintWorker, sich zu beenden.