

Multithreading

Davis Nygren, Austin Niccum, Garret Barnes, Daniel Lehman

Sources: docs.oracle.com

- **Threading/Parallelism Summary**

Parallelism is the concept of software being able to do more than one thing at a time, such as a visualizer playing audio and creating graphics simultaneously. Concurrency, a similar concept, is the concept of different units of software processing or accessing the same thing at essentially the same time.

The two types of units of execution are *processes* and *threads*. Processes are self-contained, while threads exist inside of a process and share the process's resources. Each Java application has one or more threads, starting with the main thread.

- **Parallelism (threading) in Java**

Java is designed to support parallelism. The simplest way is using multiple threads. In Java, a thread works by first defining a Runnable object, with Runnable being an interface with a single method known as "run", and then passing the Runnable object as a parameter to new Thread().start() to start it. The run acts as the threads "Main" method, in that the thread will run the code in the run method for the runnable object one constructs, then will die.

New threads can be created by implementing Runnable or extending Thread. Example code implementing runnable:

```
public class HelloRunnable implements Runnable {  
  
    public void run() {  
        System.out.println("Hello from a thread!");  
    }  
  
    public static void main(String args[]) {  
        (new Thread(new HelloRunnable())).start();  
    }  
}
```

Example code extending Thread:

```
public class HelloThread extends Thread {  
  
    public void run() {  
        System.out.println("Hello from a thread!");  
    }  
  
    public static void main(String args[]) {  
        (new HelloThread()).start();  
    }  
}
```

Multithreading

Davis Nygren, Austin Niccum, Garret Barnes, Daniel Lehman

New threads can also be created to be started later or immediately started inside of a class:

```
new Runnable() {  
    public void run() {  
        doStuff();  
    }  
}  
  
new Thread(new Runnable() {  
    public void run() {  
        doStuff();  
    }  
}).start();
```