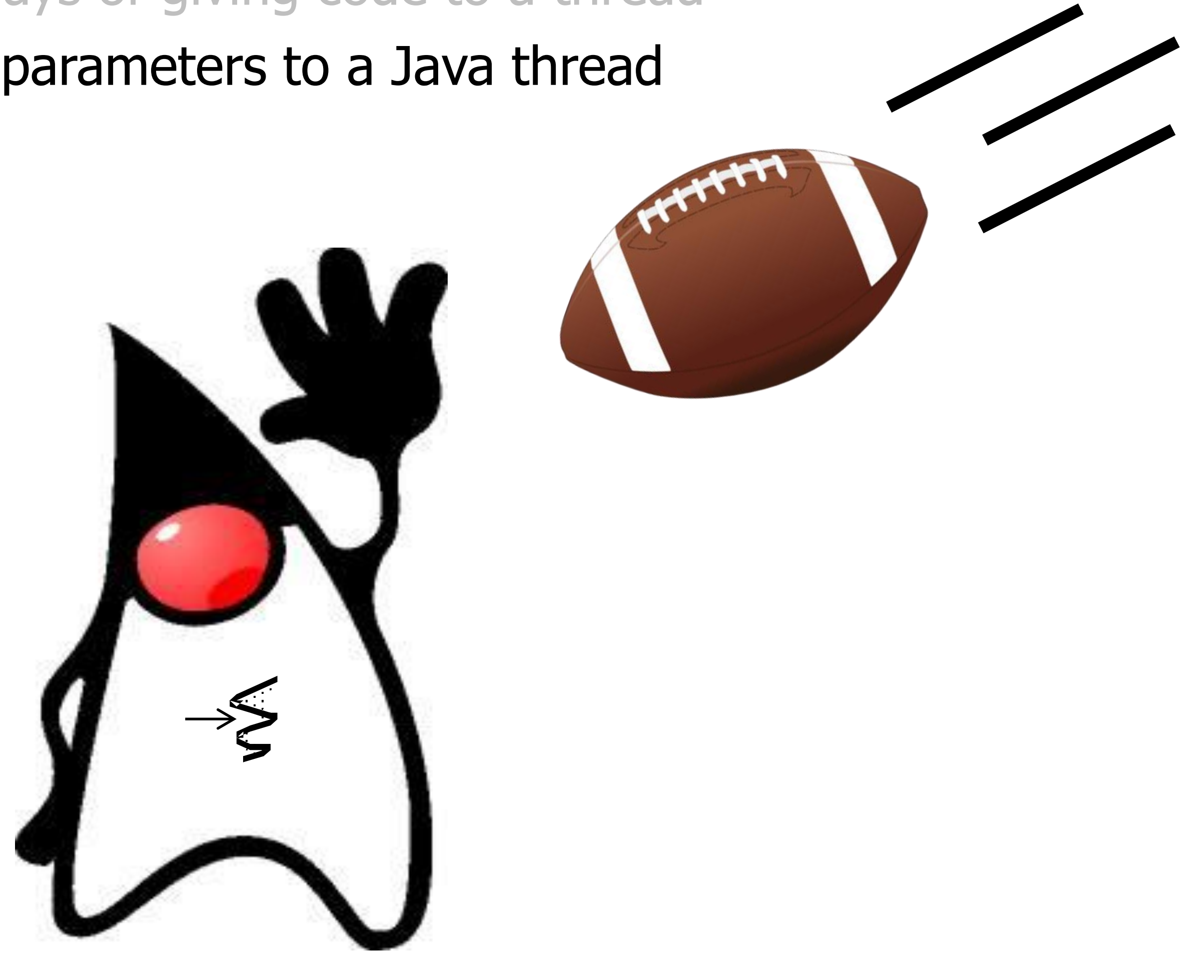
Java Thread: Passing Parameters to a Thread

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt

Institute for Software
Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA

Learning Objectives in this Part of the Lesson

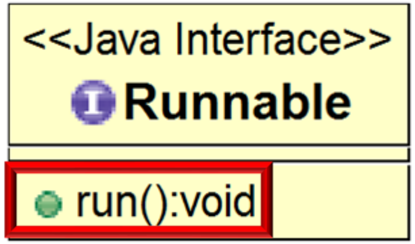
- Understand how Java threads support concurrency
- Learn how our case study app works
- Know alternative ways of giving code to a thread
- Learn how to pass parameters to a Java thread



Passing Parameters to a Java Thread

Passing Parameters to a Java Thread

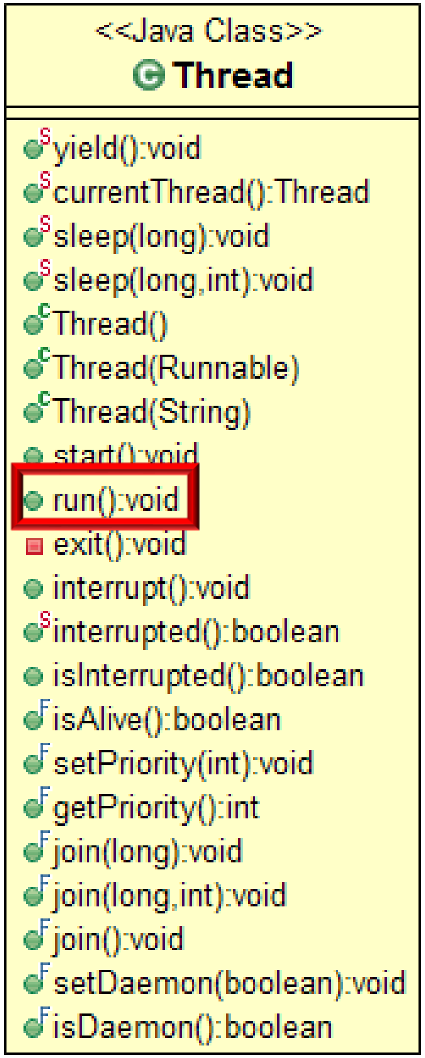
- The run() methods defined in Java Thread & Runnable take no parameters



<<Java Interface>>

Runnable

run():void



<<Java Class>>

Thread

yield():void

currentThread():Thread

sleep(long):void

sleep(long,int):void

Thread()

Thread(Runnable)

Thread(String)

start():void

run():void

exit():void

interrupt():void

interrupted():boolean

isInterrupted():boolean

isAlive():boolean

setPriority(int):void

getPriority():int

join(long):void

join(long,int):void

join():void

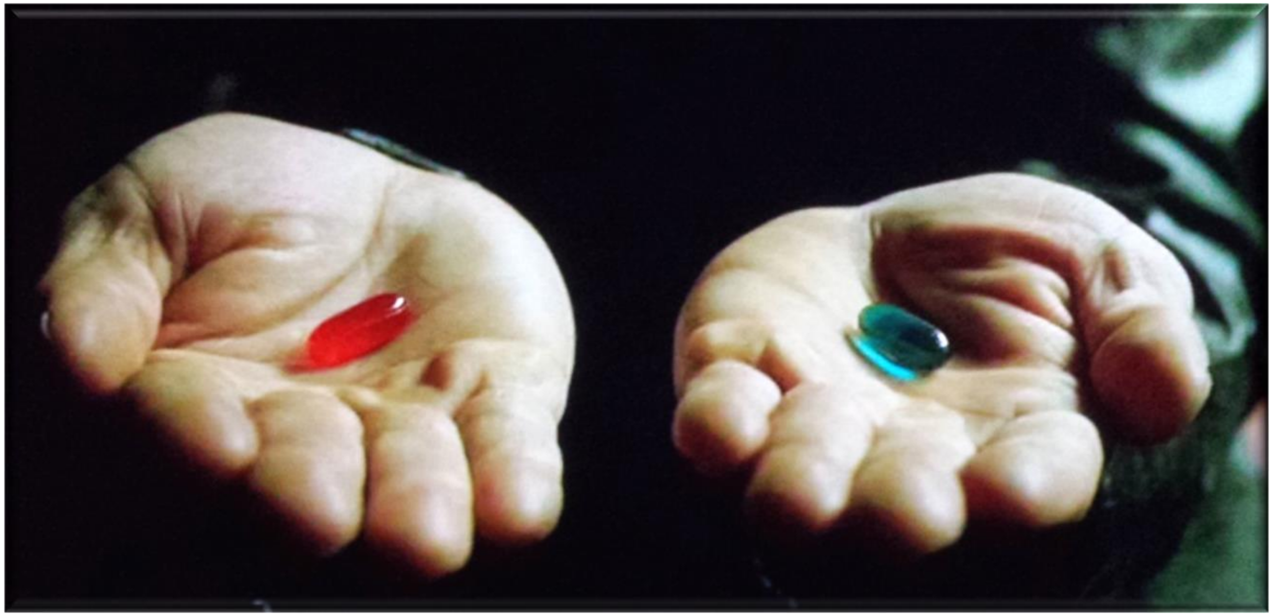
setDaemon(boolean):void

isDaemon():boolean

This raises the question of how to pass parameters to a Java thread!

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means



Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor

```
public class GCDRunnable extends Random implements Runnable {
```

See github.com/douglascraigschmidt/POSA/tree/master/ex/M3/GCD/Concurrent/app/src/main/java/vandy/mooc/gcd/activities/GCDRunnable.java

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor

```
public class GCDRunnable extends Random implements Runnable {
  private final MainActivity mActivity;
  ...
```

Define field(s) to store parameters passed to a runnable or thread object

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor

```
public class GCDRunnable extends Random implements Runnable {
  private final MainActivity mActivity;

  public GCDRunnable(MainActivity mainActivity)
  { mActivity = mainActivity; }
  ...
```

Add the parameter(s) to the constructor signature & store them in the field(s)

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor

```
public class GCDRunnable extends Random implements Runnable {
  private final MainActivity mActivity;

  public GCDRunnable(MainActivity mainActivity)
  { mActivity = mainActivity; }

  public void run() {
    final String threadString =
      " with thread id " + Thread.currentThread();
    mActivity.println("Entering run()" + threadString);
    ...
```

Use the field(s) within the thread's run() hook method to customize its behavior

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor

```
public class GCDRunnable extends Random implements Runnable {
  private final MainActivity mActivity;

  public GCDRunnable(MainActivity mainActivity)
  { mActivity = mainActivity; }

  public void run() {
    final String threadString =
      " with thread id " + Thread.currentThread();
    mActivity.println("Entering run()" + threadString);
    ...
```

```
public class MainActivity ... { ...
  public void runRunnable(View v) { ...
     new Thread(new GDCRunnable(this));
  ...
```

Pass the parameter(s) when the runnable or thread is created

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor
  - As parameters to “setter” methods

```
public class GCDThread extends Thread {
```

See github.com/douglascraigschmidt/POSA/tree/master/ex/M3/GCD/Concurrent/app/src/main/java/vandy/mooc/gcd/activities/GCDThread.java

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor
  - As parameters to "setter" methods

```
public class GCDThread extends Thread {
  private MainActivity mActivity; private Random mRandom;
  ...
```

Define field(s) to store parameters passed to a runnable or thread object

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor
  - As parameters to "setter" methods

```
public class GCDThread extends Thread {
  private MainActivity mActivity; private Random mRandom;

  public GCDThread setActivity(MainActivity activity)
  { mActivity = activity; return this; }

  public GCDThread setRandom(Random random)
  { mRandom = random; return this; }
  ...
```

Define setter methods that update field(s)

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor
  - As parameters to "setter" methods

```
public class GCDThread extends Thread {
  private MainActivity mActivity; private Random mRandom;

  public GCDThread setActivity(MainActivity activity)
  { mActivity = activity; return this; }

  public GCDThread setRandom(Random random)
  { mRandom = random; return this; }
  ...
```

Note use of "fluent interfaces," which enables "chaining" of method calls

See en.wikipedia.org/wiki/Fluent_interface

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor
  - As parameters to "setter" methods

```
public class GCDThread extends Thread {
  private MainActivity mActivity; private Random mRandom;

  public GCDThread setActivity(MainActivity activity)
  { mActivity = activity; return this; }

  public GCDThread setRandom(Random random)
  { mRandom = random; return this; }
  ...

  public void run() { ...
    mActivity.println("Entering run()" + threadString);
    ...
      int number1 = mRandom.nextInt();
      int number2 = mRandom.nextInt(); ...
```

Use the fields within the thread's run() hook method to customize its behavior

Passing Parameters to a Java Thread

- Parameters passed to run() can be supplied via one of two other means, e.g.
  - As parameters to a class constructor
  - As parameters to "setter" methods

```
public class GCDThread extends Thread {
  ...

public class MainActivity ... { ...
  public void runThread(View v) { ...
    Thread thread =
      new GCDThread()
        .setActivity(this)
        .setRandom(new Random());
  ...
```

Use the fluent interface to pass parameter(s) when the runnable or thread is created

End of Java Thread: Passing Parameters to a Thread