CSCI 1300

Intro to Computing

Gabe Johnson

Lecture 26 Mar 15, 2013

Java Graphics and Objects Part 2

Lecture Goals

1. The Automata HW
2. Javadoc
3. Be Like Hendrix
4. More Graphics

Automata HW

Java: ‘prev’ is a member variable. *You will never need to redeclare it.*

These are declarations:

```
char[] prev;
char[] prev = new char[some_length];
```

These are just assignments (you will do this):

```
prev = some_other_char_array;
prev = someString.toCharArray();
```

Declare temp array

You *can* and probably will need to declare a different array—one that isn’t the ‘prev’ member variable. In your *cycle* method, you’ll need a place to store the next round before you reassign it into prev:

```
char[] nextRound = new char[prev.length];
for (i =0; i < prev.length; i++) {
    nextRound[i] = something;
}
prev = nextRound;
```

Python Caveat

In Python, the ‘prev’ member is a string, not a character array. This is ok. It works like this:

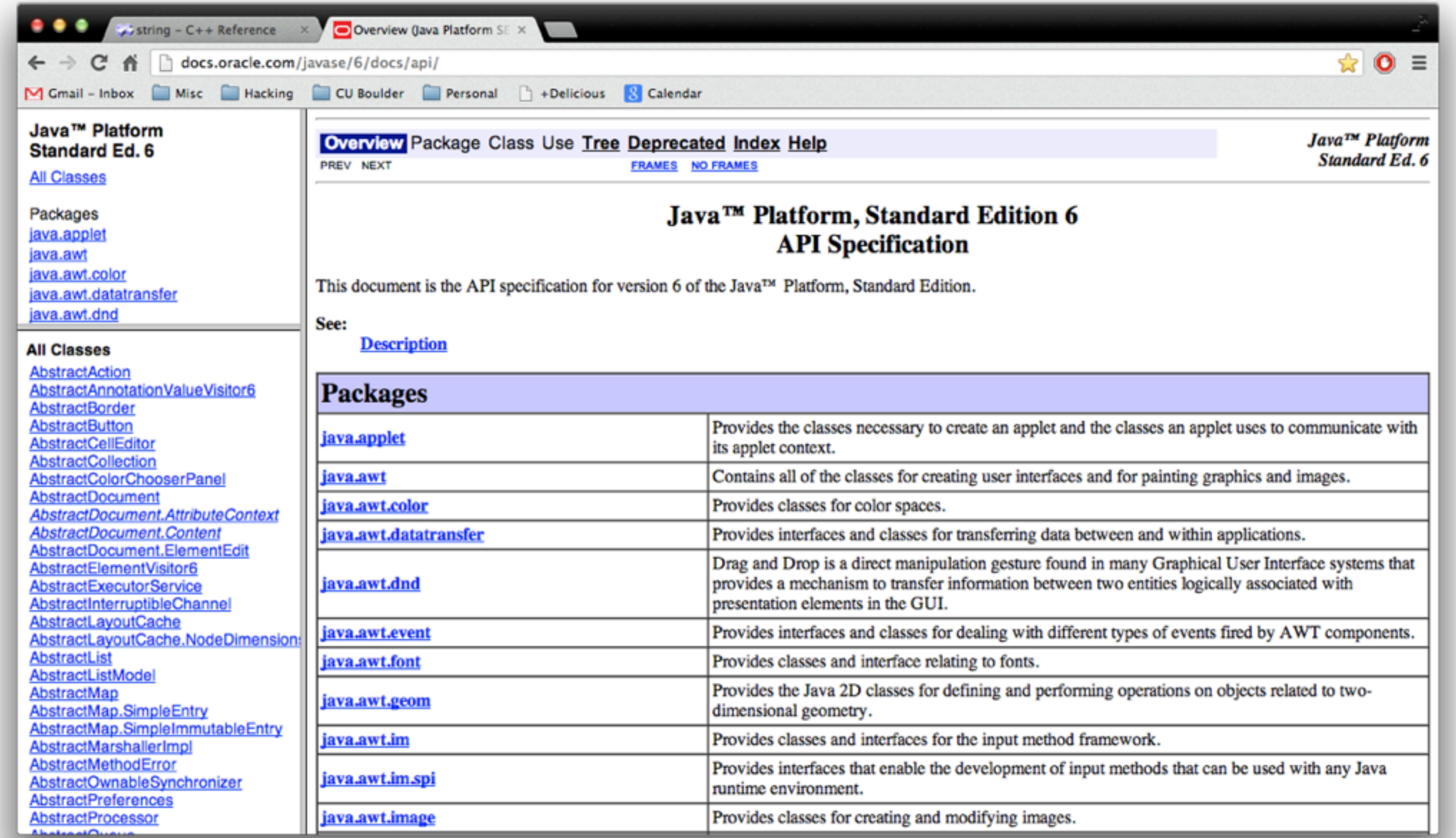
```
def cycle(self):
  next_round = “” # empty string
  for i in range(len(self.prev)):
    # append to end of ‘next_round’
    next_round = next_round + something
  self.prev = next_round
```

Javadoc

Any language you are likely to use will have extensive documentation. For Java, this is called **Javadoc**. Just Google search for Javadoc 1.6 (assuming you want version 1.6).

Bookmark the result. You should keep it open whenever you are hacking.

Top/Left: packages. java.util has basic stuff.



Java™ Platform
Standard Ed. 6

All Classes

Packages
java.applet
java.awt
java.awt.color
java.awt.datatransfer
java.awt.dnd

All Classes
AbstractAction
AbstractAnnotationValueVisitor6
AbstractBorder
AbstractButton
AbstractCellEditor
AbstractCollection
AbstractColorChooserPanel
AbstractDocument
AbstractDocument.AttributeContext
AbstractDocument.Content
AbstractDocument.ElementEdit
AbstractElementVisitor6
AbstractExecutorService
AbstractInterruptibleChannel
AbstractLayoutCache
AbstractLayoutCache.NodeDimensions
AbstractList
AbstractListModel
AbstractMap
AbstractMap.SimpleEntry
AbstractMap.SimpleImmutableEntry
AbstractMarshallerImpl
AbstractMethodError
AbstractOwnableSynchronizer
AbstractPreferences
AbstractProcessor

Java™ Platform
Standard Ed. 6

Java™ Platform, Standard Edition 6
API Specification

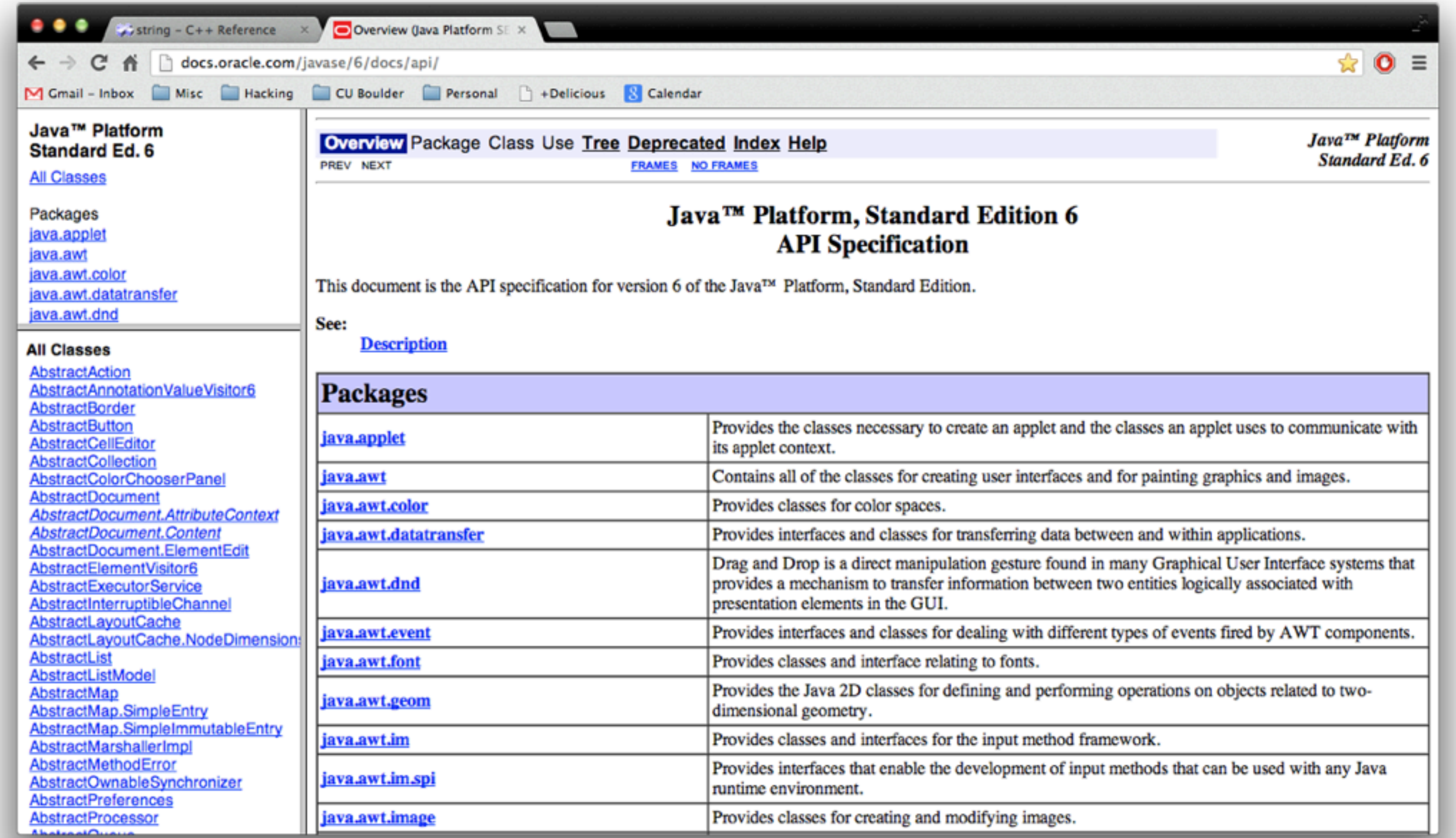
This document is the API specification for version 6 of the Java™ Platform, Standard Edition.

See:
Description

Packages	
java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data between and within applications.
java.awt.dnd	Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI.
java.awt.event	Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font	Provides classes and interface relating to fonts.
java.awt.geom	Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.
java.awt.im	Provides classes and interfaces for the input method framework.
java.awt.im.spi	Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.
java.awt.image	Provides classes for creating and modifying images.

Bottom left: classes in a package.

Main area: documentation or package info.

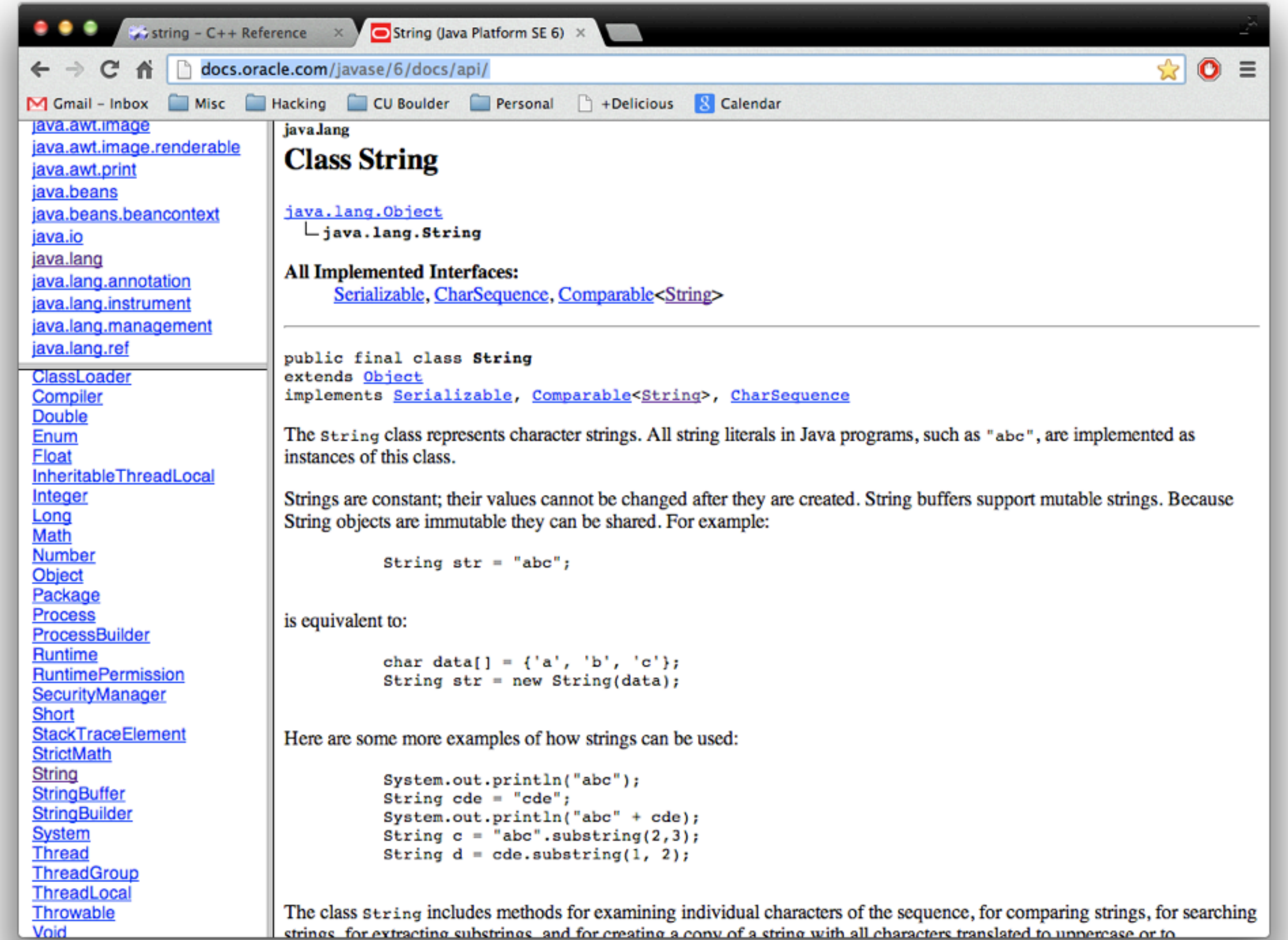


Java™ Platform, Standard Edition 6 API Specification

This document is the API specification for version 6 of the Java™ Platform, Standard Edition.

See:
Description

Packages	
java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data between and within applications.
java.awt.dnd	Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI.
java.awt.event	Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font	Provides classes and interface relating to fonts.
java.awt.geom	Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.
java.awt.im	Provides classes and interfaces for the input method framework.
java.awt.im.spi	Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.
java.awt.image	Provides classes for creating and modifying images.



java.awt.image
java.awt.image.renderable
java.awt.print
java.beans
java.beans.beancontext
java.io
java.lang
java.lang.annotation
java.lang.instrument
java.lang.management
java.lang.ref

ClassLoader
Compiler
Double
Enum
Float
InheritableThreadLocal
Integer
Long
Math
Number
Object
Package
Process
ProcessBuilder
Runtime
RuntimePermission
SecurityManager
Short
StackTraceElement
StrictMath
String
StringBuffer
StringBuilder
System
Thread
ThreadGroup
ThreadLocal
Throwable
Void

java.lang

Class String

```
java.lang.Object
  └java.lang.String
```

All Implemented Interfaces:

Serializable, CharSequence, Comparable<String>

```
public final class String
extends Object
implements Serializable, Comparable<String>, CharSequence
```

The `String` class represents character strings. All string literals in Java programs, such as `"abc"`, are implemented as instances of this class.

Strings are constant; their values cannot be changed after they are created. String buffers support mutable strings. Because String objects are immutable they can be shared. For example:

```
     String str = "abc";
```

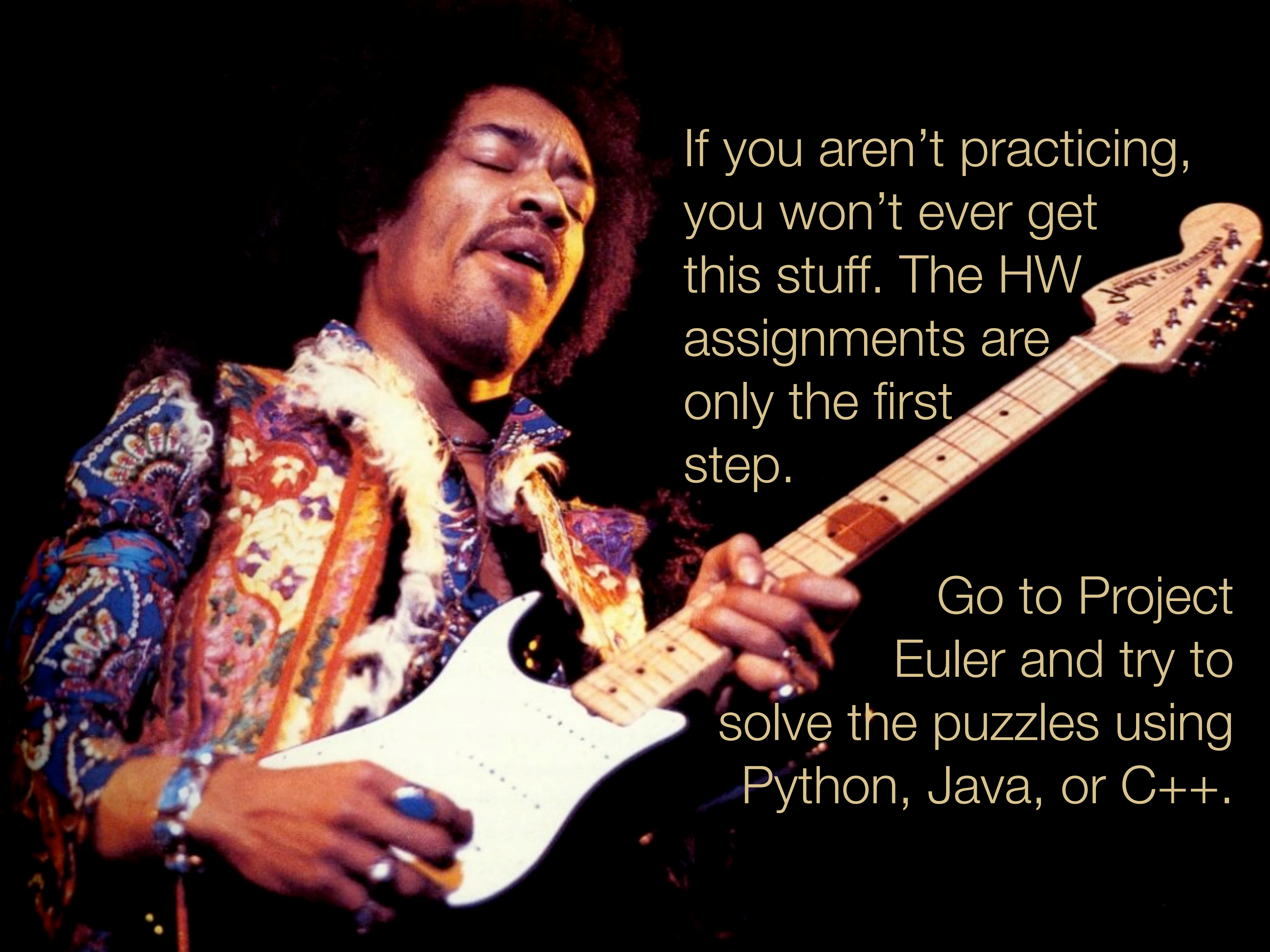
is equivalent to:

```
     char data[] = {'a', 'b', 'c'};
     String str = new String(data);
```

Here are some more examples of how strings can be used:

```
     System.out.println("abc");
     String cde = "cde";
     System.out.println("abc" + cde);
     String c = "abc".substring(2,3);
     String d = cde.substring(1, 2);
```

The class `String` includes methods for examining individual characters of the sequence, for comparing strings, for searching strings, for extracting substrings, and for creating a copy of a string with all characters translated to uppercase or to



If you aren't practicing, you won't ever get this stuff. The HW assignments are only the first step.

Go to Project Euler and try to solve the puzzles using Python, Java, or C++.

PRACTICE
EVERY
DAY

... and
you’ll be
awesome.

More Java Graphics

Today we’re going to program our circles to fly around and have social behavior.

This is Lecture_26.java.