

# Documentation

javadoc

# Documentation

- not a programmer's first love
- lives in a separate file somewhere
- usually a deliverable on the schedule
- often not updated when program is changed
- incorrect documentation is arguably worse than none

# Javadoc

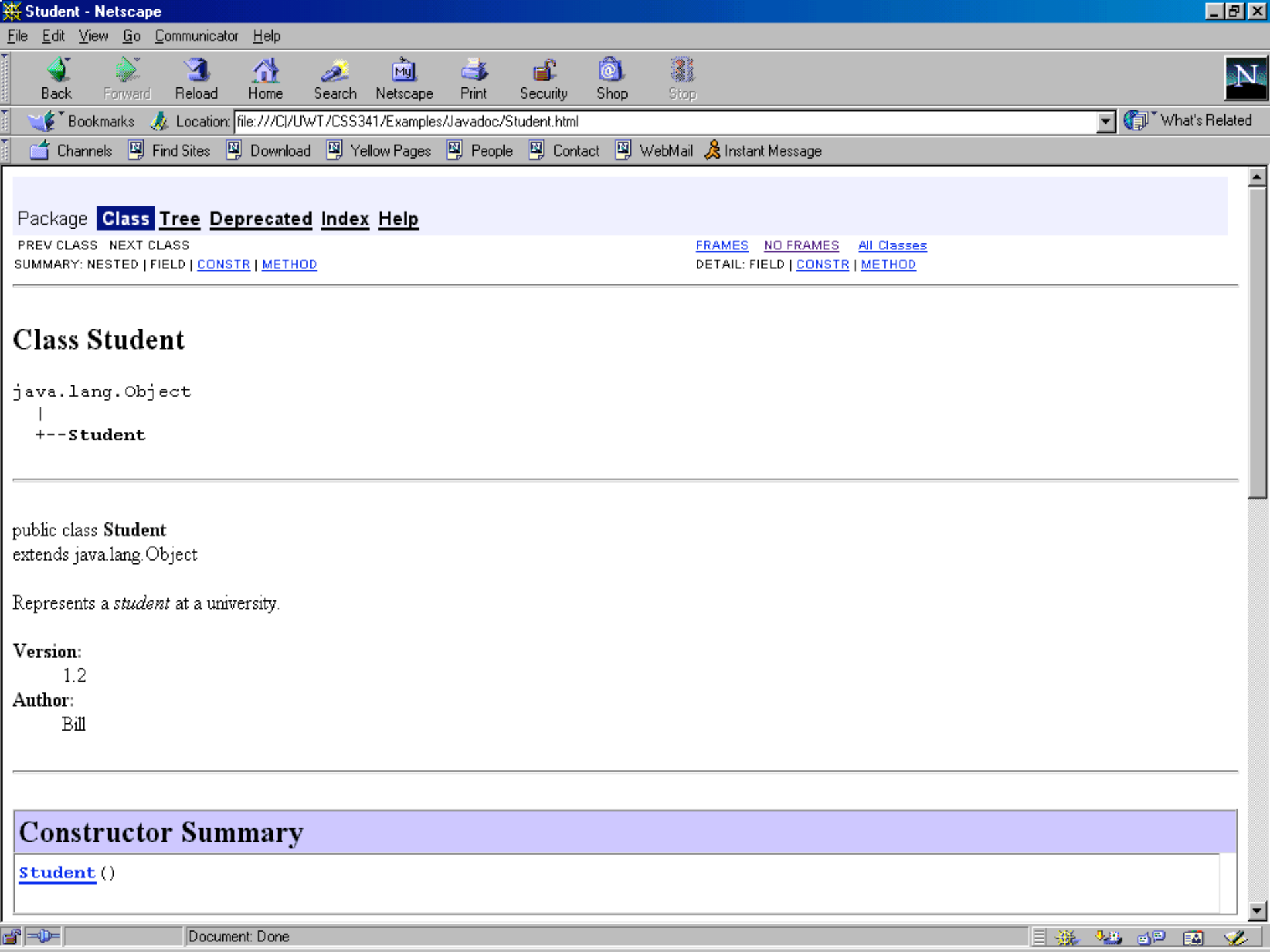
- API documentation is included in the source code as comments
- the *javadoc* program will analyze the source code and produce formatted html documentation
- same style as Java Runtime Library documentation
- hopefully, programmers will update documentation when they change the code

# Javadoc Comment Format

- block comments introduced with `/**`
- may include html formatting tags
- ***placed immediately*** ahead of class or method definition
  - `/**`
  - `* Represents a <i>student<i> at a university.`
  - `* @author Bill`
  - `* @version 1.2`
  - `*/`
  - `public class Student {`

# Method Documentation

- up to first period included in method summary
- everything included in detail documentation
- `@param` tag to describe **each** parameter
- `@return` tag **if not void**
- tags do not specify data type
- `/**`
- `* Adds a course grade to the student's record.`
- `* Updates both gpa and credits.`
- `* @param grade the course grade`
- `* @param credits the course credits`
- `* @return the new gpa`
- `* @throws IllegalArgumentException if`
- `* grade is > 4.0`
- `*/`



Package [Class](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#)

DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

## Class Student

```
java.lang.Object
|
+--Student
```

```
public class Student
extends java.lang.Object
```

Represents a *student* at a university.

**Version:**

1.2

**Author:**

Bill

## Constructor Summary

[Student](#) ()

Back Forward Reload Home Search Netscape Print Security Shop Stop

Channels Find Sites Download Yellow Pages People Contact WebMail Instant Message

## Method Summary

```
double addGrade(double grade, int credits)
    Adds a course grade to the student's record.
```

### Methods inherited from class java.lang.Object

```
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait
```

## Constructor Detail

### Student

```
public Student()
```

## Method Detail

### addGrade

```
public double addGrade(double grade,
                        int credits)
```

Adds a course grade to the student's record. Updates both *gpa* and *credits*.

#### Parameters:

- grade* - the course grade
- credits* - the course credits

#### Returns:

the new *gpa*

