

# Static Variables, Scope, Javadoc and Style

---

CSC 116 – Section 002  
March 28, 2005

---

© 2005 Sarah E. Smith

## Static Variables

---

- A variable the whole class shares
- Each instance of a class sees the same value for the static variable
- If one object changes the value of the static variables, then the value of the variables changes for all classes
- Initial value should be given when you declare the variable, not in the constructor.

---

© 2005 Sarah E. Smith

## Static Variables (2)

---

- The assignment of a value to a static variable occurs the first time an object is created
    - All subsequent objects that are created do not assign a value to the static variable, it already exists for the object.
  - Example
- ```
private static int lastAssignedActNumber = 0;
```

---

© 2005 Sarah E. Smith

## Scope and Shadowing

---

- A local variable and instance variable may have the same name
  - This is known as shadowing
- The local variable is just referred to by its name
- The instance variable is referred to by `this.name`

---

© 2005 Sarah E. Smith

## Scope and Shadowing

---

- Example:

```
public class BankAccount {  
    private double balance;  
    public BankAccount(double balance) {  
        balance = balance; //ERROR  
        this.balance = balance;  
    }  
}
```

---

© 2005 Sarah E. Smith

## Javadoc

---

- Java has a javadoc program that creates a website containing documentation on your code
- You write the documentation inside the Javadoc tags in your code – the javadoc program creates the webpages
- Command:  
*javadoc file1.java file2.java ...*

---

© 2005 Sarah E. Smith

# Javadoc

---

- Recommendation: create an empty directory (called doc) and run the javadoc command in there and give the path to the directory/\*.java files you wish to javadoc
- Open index.html to see documentation
- Default is to only show documentation for public portions of the class
- Use `-private` flag to print javadoc for private portions of a class

---

© 2005 Sarah E. Smith

# Javadoc Comments

---

- Javadoc comments go inside `/**` and `*/` comment symbols
- You can use HTML tags inside of your Javadoc comments
  - They'll show up properly when the page is rendered by running javadoc

---

© 2005 Sarah E. Smith

## Common Javadoc Tags

---

- `@author author name (email)`
  - Specifies the author(s) of the file
  - One author tag for each author
- `@version x.x`
  - Specifies the version of the file
- `@param varName description`
  - Specifies the name of a parameter and the description

---

© 2005 Sarah E. Smith

## Common Javadoc Tags

---

- `@return type description`
  - Specifies the return type and what the return value means

---

© 2005 Sarah E. Smith

## Style Guidelines

---

- These are common guidelines that makes code easier to read, understand, **debug**, and maintain
- Indent Nested Code
  - Lines up curly braces so its easier to find a missing brace
- Using Braces
  - Will be easier to add content to blocks of code later

---

© 2005 Sarah E. Smith

## Style Guidelines (2)

---

- Be consistent with the location of your braces.
- White space!!
- Break long lines
  - Around 80 characters
- Use meaningful names

---

© 2005 Sarah E. Smith

## References

---

- Jason Schwarz's Lecture 19 & 20 slides:  
<http://courses.ncsu.edu/csc116/>