

# Lab 4

# NetBeans & Debugging

CSCE A201

Computer Programming I

# NetBeans IDE

- Slower to start, but provides more **features**
- Must create a project
  - File > New Project... > Java > Java Application
  - Use a meaningful project name!
  - This will also be the package name
    - code, including import statements, must be below the package name
- Syntax highlighting, interactive error finding, **autocompletion**
- Will suggest fixes for simple problems like missing import statements
- To access a project on a different computer, easiest to just copy (zip) the entire project folder
  - Show directory structure on disk

# Debugging

- *How can we debug a program that isn't working?*
  - Read code hundreds of times
    - helps to use good **style** & **indentation** to make this easier
  - Compile the code
  - Trace variables by adding print statements
  - Use a **debugger**
    - at the Unix command line, use **jdb** instead of **java**
    - much easier to use an IDE for this

# NetBeans Debugger

- Set a **breakpoint** (Ctrl + F8) or click on left border pane
  - point in program where you want the debugger to stop execution
- Run code using the **debugger**
  - Debug menu > Debug Project (Ctrl + F5)
- Program will stop just **before** it executes the line with the breakpoint
  - Inspect variables of interest
    - **Hover** over variables with mouse
    - **Highlight** a portion of a line to evaluate
    - **Variables** tab
    - type in variables to **watch**
  - Use **Step** & **Continue** buttons to control execution
    - For method control, use Step Over, Step Into, Step Out, etc...
- Use **Finish Debugger** (stop button) to end debug session

# Debugging Infinite Loops

- How might you debug an infinite loop?
- What if the loop runs normally most of the time, but only occasionally runs infinitely?
  - One solution is to add a count variable (if one doesn't exist already) and then set a breakpoint if the count variable exceeds some threshold

# Lab 4 Exercise

- Write a program with an infinite loop in NetBeans that simulates a random coin toss.
  - Print the heads or tails outcome for each iteration in the loop
- Debug this loop by assuming it is 'infinite' after it executes 500 times. At this point, set a breakpoint and explore the values of all variables in your program using the NetBeans debugger
  - In a real problem, such exploration would hopefully enable you to find the bug!
- To get credit for the lab show the lab assistant the values of all variables at the 500<sup>th</sup> iteration in the debugger and that you can step through the code to the next iteration