

Debugging

Debugging

- Bugs happen. Don't panic!
 - (things break, the sun will continue to rise)
- Processing 3.0 + has decent debugging tools
 - This was lacking in earlier versions

Debugging Techniques

- Simplify while coding
 - When writing code, do a small piece at a time and test each piece as you go.
 - Catch bugs quickly
 - Bug is isolated to the new code you just wrote

Debugging Techniques

- Simplify while debugging
 - Same strategy: isolate the source of the bug
 - Comment out sections of code to see what block causes the bug `/* */`
 - Once enough code has been eliminated to get rid of the bug, add lines back in to see which line reintroduced the bug
 - Use as simple a test case as possible that will replicate the bug
 - e.g. instead of running algorithm on a photo, run it on a two color image that just contains a circle or square
 - Less issues to consider, can make bug clearer

Debugging Techniques

- Take a break
 - Let your mind relax and think of other things
- Explain your code to a friend or out loud
 - Especially good for finding logic errors
- Get more information
 - Key technique
 - Use `println` to print the value of key variables at different points in time
 - Look for unexpected values

The Processing Debugger

- Enable Debugger
 - Debug->Enable Debugger
 - OR click on the bug icon in the upper right

The Processing Debugger

- Break point
 - Marks a line in your code where you can pause it during execution to inspect what is happening
 - Set by:
 - Clicking on line number OR
 - Ctrl + B OR
 - Debug -> Toggle Breakpoint
- Variables window
 - Displays the value of all the active variables

Advanced Steps

- Step will not enter any functions that you may have called
- Debug -> Step Into
 - Call when on a line with a function you wrote
 - Will “step into” that function
 - Go to the first line of the function
- Debug -> Step Out
 - Leave the function you are in
 - Return to the code that called the function

The Processing Debugger

- Debug
 - Start the program running
 - Will stop on the first break point
- Step
 - Move to the next line of code
- Continue
 - Run until the next breakpoint
- Stop
 - End the program

