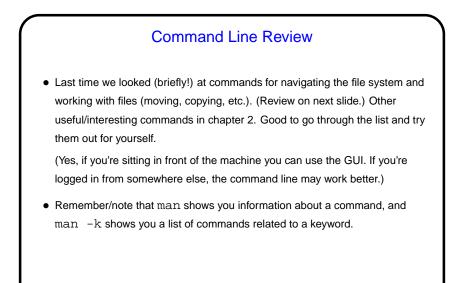


Minute Essay From Last Lecture

 Sometimes people comment "lots of commands to learn" or "lecture moved pretty fast". Yes. Practice/experiment outside class will help. True for many (most?) things we do in this course. If you have trouble remembering all the commands (which you might at first!): In times past beginners got paper "cheat sheets" of commonly-used commands. Maybe make yourself an electronic equivalent?

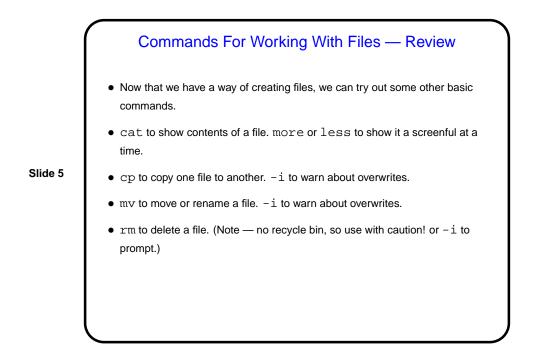
Slide 2

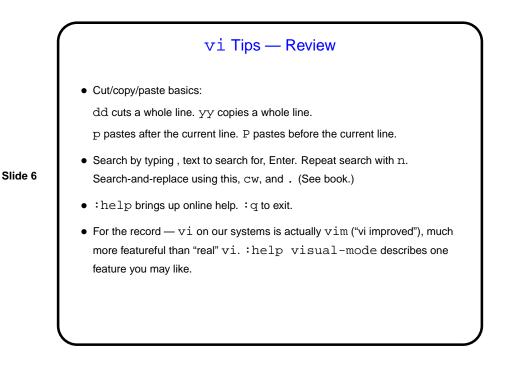
• Why does : q sometimes not seem to work? (Later.)

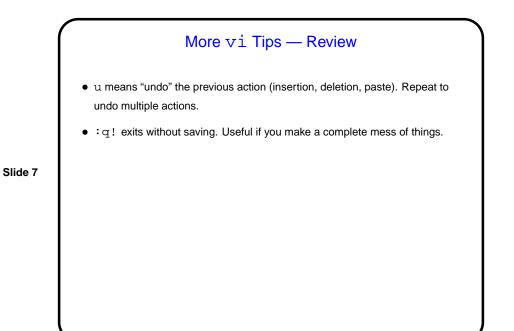


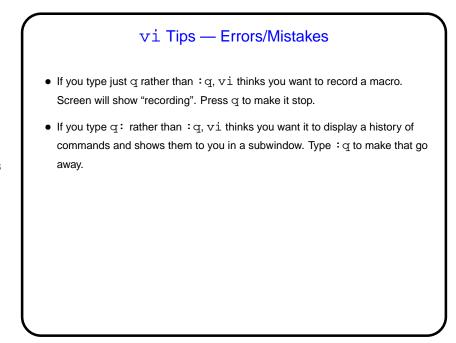
Commands For Navigating the Filesystem - Review

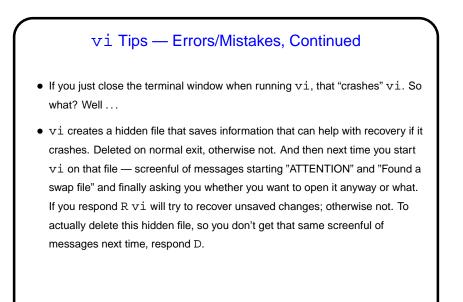
- Unlike GUIs (at least sometimes!), shell programs (mostly?) have a notion of "current/working directory". pwd shows what it is. cd changes it.
- mkdir to create a new directory. rmdir to delete one (must be empty).





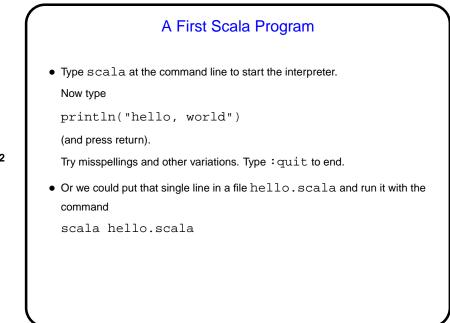


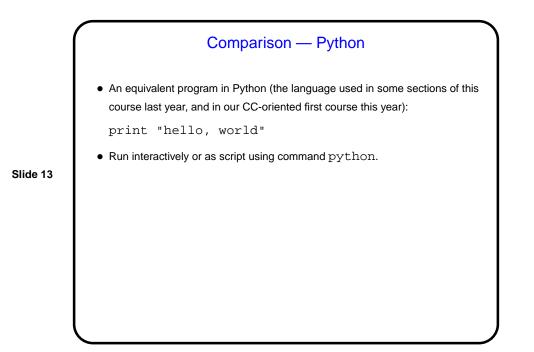




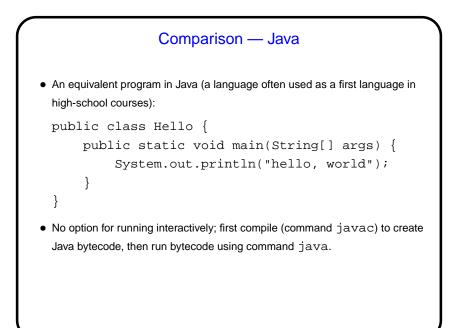
UNIX Filesystem Basics
Unlike in Windows (and Mac?), UNIX filesystems are case-sensitive (so hello and Hello are different files).
Files have two levels of ownership — "owner" (user) and "group". Groups allow sharing files with some but not all users.
File access is controlled by "permissions". Three levels (owner, group, and everyone else), three types of access (read, write, execute).
Is -1 shows permissions. chmod changes them.

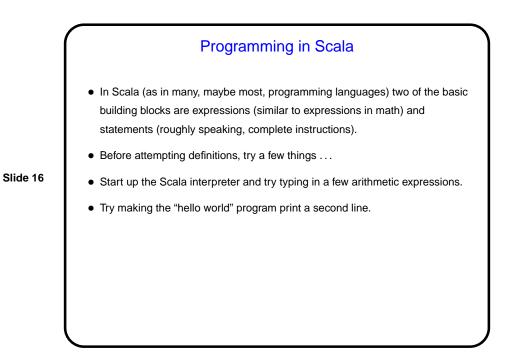
Scala is short for "scalable programming language". (We may talk more later about what that means.) Relatively new language, but we think good for a first course.
Various options for running Scala source code. Today we will look at two of them — typing it in interactively, and executing "scripts". scala starts an interactive environment ("REPL" – "read, evaluate, print" loop), good for trying things out. scala program.scala runs the program in file program.scala.
By tradition (established by the inventors of the C language, in 1970-something), our first program will just write to the screen "hello, world").

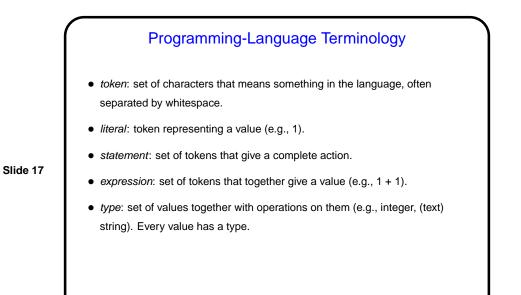


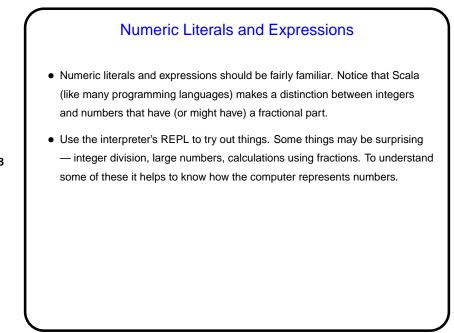


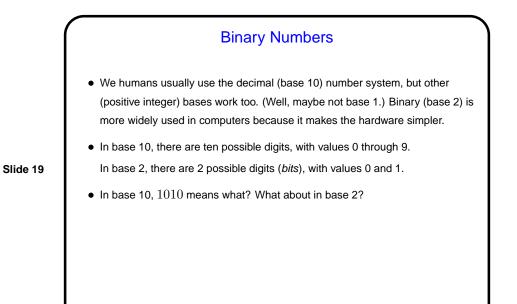
Comparison — C
• An equivalent program in C (the language previously used in this course):
 #include <stdio.h>
 int main(void) {
 printf("hello, world\n");
 return 0;
 }
• No option for running interactively; first compile (command gcc) to create
 executable file, then "run" the file.

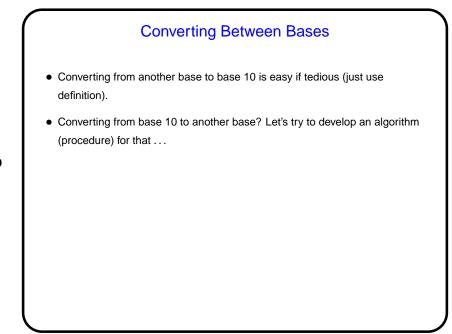


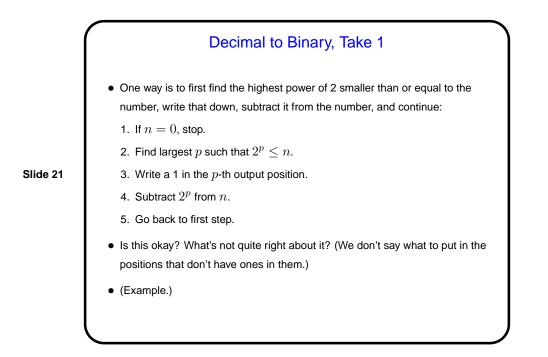












Decimal to Binary, Take 2 Another way produces the answer from right to left rather than left to right, repeatedly dividing by 2 (again n will be the number we want to convert): If n = 0, stop. Divide n by 2, giving quotient q and remainder r. Write down r. Set n equal to q. Go back to first step. Is this okay? What's not quite right about it? (We don't say to write down the remainders from right to left.) (Example.)

