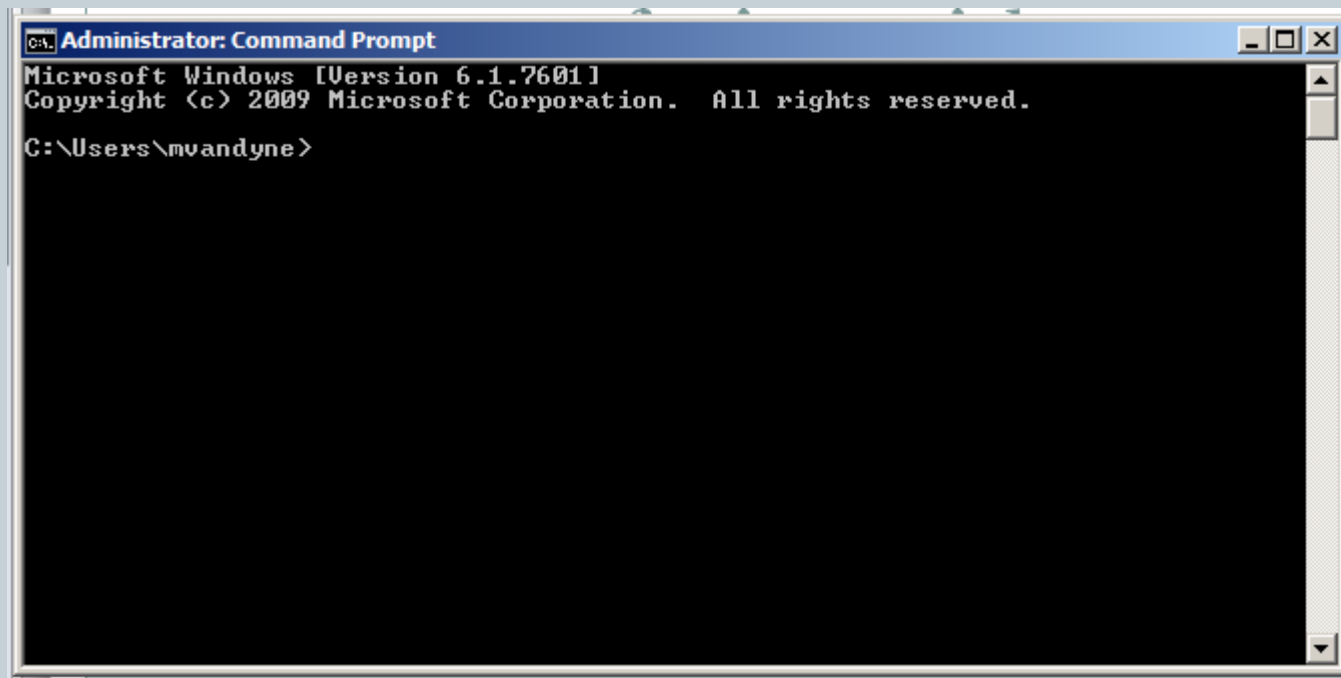


# The Command Shell



# Outline

---

- Starting the Command Shell
  - Locally
  - Remote Host
- Directory Structure
  - Moving around the directories
- Displaying File Contents
- Compiling and Running a Java Program
- Editing a Text File
- Copying Files between Computers
- File Redirection and Piping
- Command Summary

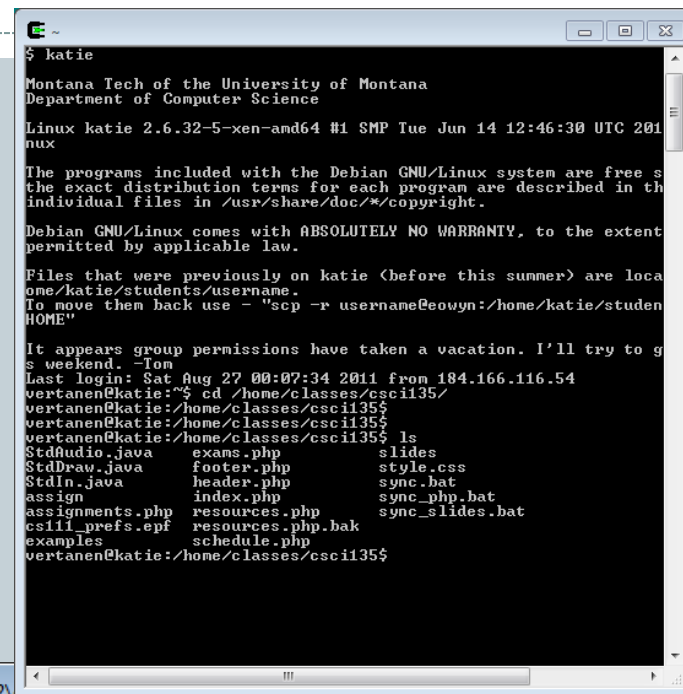
# Interfacing with your Computer

- **GUI (Graphical User Interfaces)**
  - Today: predominant interaction method
  - Windows, buttons, mouse
  - Advantages
    - ✦ Easier for novices
    - ✦ No commands to remember
    - ✦ Rich input and output capabilities



# Interfacing with your Computer

- Command Line Interface (CLI)
  - Originally the only option
  - Input by typing commands
  - Advantages:
    - ✦ Can be faster for experts than a GUI
    - ✦ Easier to automate tasks
    - ✦ Easier to hook programs together



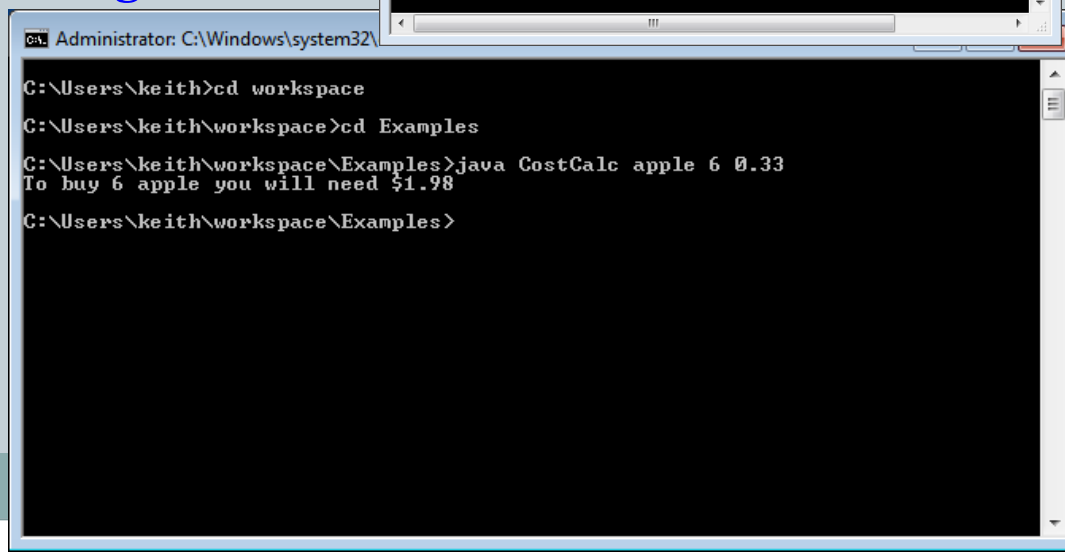
```
$ katie
Montana Tech of the University of Montana
Department of Computer Science
Linux katie 2.6.32-5-xen-amd64 #1 SMP Tue Jun 14 12:46:30 UTC 2011
nux

The programs included with the Debian GNU/Linux system are free s
the exact distribution terms for each program are described in th
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

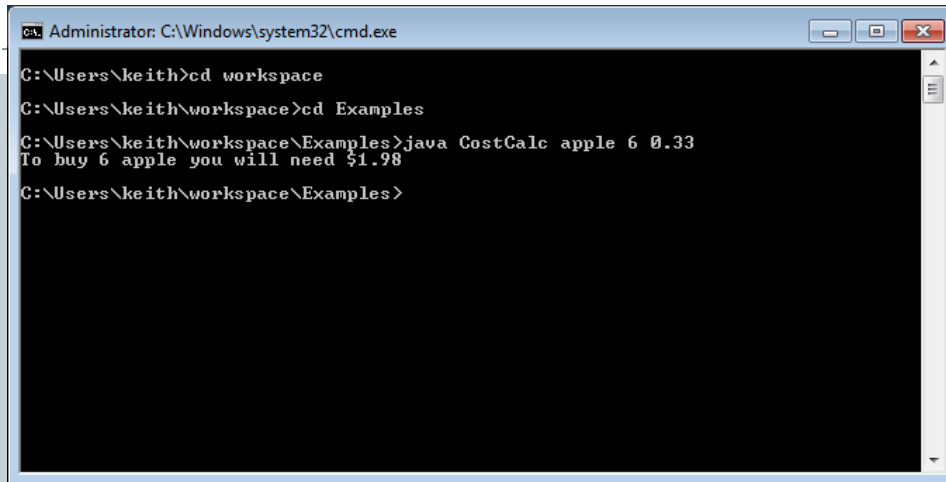
Files that were previously on katie <before this summer> are loca
one/katie/students/username.
To move them back use - "scp -r username@eovyn:/home/katie/studen
HOME"

It appears group permissions have taken a vacation. I'll try to g
s weekend. -Tom
Last login: Sat Aug 27 00:07:34 2011 from 184.166.116.54
vertanen@katie:~$ cd /home/classes/csc1135/
vertanen@katie:/home/classes/csc1135$
vertanen@katie:/home/classes/csc1135$ ls
StdAudio.java      exams.php          slides
StdDraw.java       footer.php         style.css
StdIn.java         header.php        sync.bat
assign             index.php         sync.php.bat
assignments.php    resources.php     sync_slides.bat
cs111_prefs.epf   resources.php.bak
examples           schedule.php
vertanen@katie:/home/classes/csc1135$
```



```
C:\Users\keith>cd workspace
C:\Users\keith\workspace>cd Examples
C:\Users\keith\workspace\Examples>java CostCalc apple 6 0.33
To buy 6 apple you will need $1.98
C:\Users\keith\workspace\Examples>
```

## Starting a Command Shell

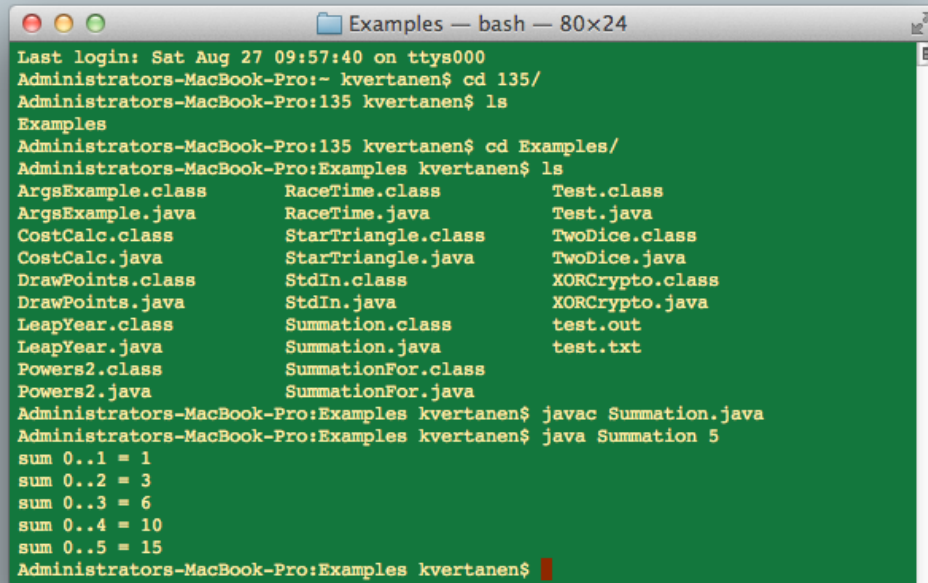
A screenshot of a Windows 7 Command Prompt window. The title bar reads "Administrator: C:\Windows\system32\cmd.exe". The command history shows: "C:\Users\keith>cd workspace", "C:\Users\keith\workspace>cd Examples", "C:\Users\keith\workspace\Examples>java CostCalc apple 6 0.33", and "C:\Users\keith\workspace\Examples>". The output of the last command is "To buy 6 apple you will need \$1.98".

```
Administrator: C:\Windows\system32\cmd.exe
C:\Users\keith>cd workspace
C:\Users\keith\workspace>cd Examples
C:\Users\keith\workspace\Examples>java CostCalc apple 6 0.33
To buy 6 apple you will need $1.98
C:\Users\keith\workspace\Examples>
```

### Windows 7

*Start → type "cmd"*

*All Programs → Accessories → Command Prompt*

A screenshot of a Mac Terminal window titled "Examples — bash — 80x24". It shows the user navigating to the "Examples" directory and listing files. The output of "ls" is a three-column list of files. Then, the user compiles and runs "Summation.java", which outputs a series of sums.

```
Examples — bash — 80x24
Last login: Sat Aug 27 09:57:40 on ttys000
Administrators-MacBook-Pro:~ kvertanen$ cd 135/
Administrators-MacBook-Pro:135 kvertanen$ ls
Examples
Administrators-MacBook-Pro:135 kvertanen$ cd Examples/
Administrators-MacBook-Pro:Examples kvertanen$ ls
ArgsExample.class      RaceTime.class        Test.class
ArgsExample.java       RaceTime.java         Test.java
CostCalc.class         StarTriangle.class    TwoDice.class
CostCalc.java          StarTriangle.java     XORCrypto.class
DrawPoints.class       StdIn.class           XORCrypto.java
DrawPoints.java        StdIn.java            test.out
LeapYear.class         Summation.class       test.txt
LeapYear.java          Summation.java
Powers2.class          SummationFor.class
Powers2.java           SummationFor.java
Administrators-MacBook-Pro:Examples kvertanen$ javac Summation.java
Administrators-MacBook-Pro:Examples kvertanen$ java Summation 5
sum 0..1 = 1
sum 0..2 = 3
sum 0..3 = 6
sum 0..4 = 10
sum 0..5 = 15
Administrators-MacBook-Pro:Examples kvertanen$
```

### Mac

*Spotlight → type "terminal"*

*Go → Applications → Utilities → Terminal*

# Starting a Command Window in Linux: putty

- Use putty.exe
  - cs.mtech.edu then go to the Students tab, select Software
  - Download both putty and winscp
  - Double click on putty
    - ✦ Enter katie.mtech.edu for the Host Name (or you can use an IP)
    - ✦ Click Open button
    - ✦ Enter login name (first initial last name, all lowercase)
      - Except Keith – yours is bocian
    - ✦ Enter password: goldreagana2rcher
      - Except Taylor and Chris – you already have accounts and passwords
    - ✦ You will need to change this immediately – remember what you changed it to.

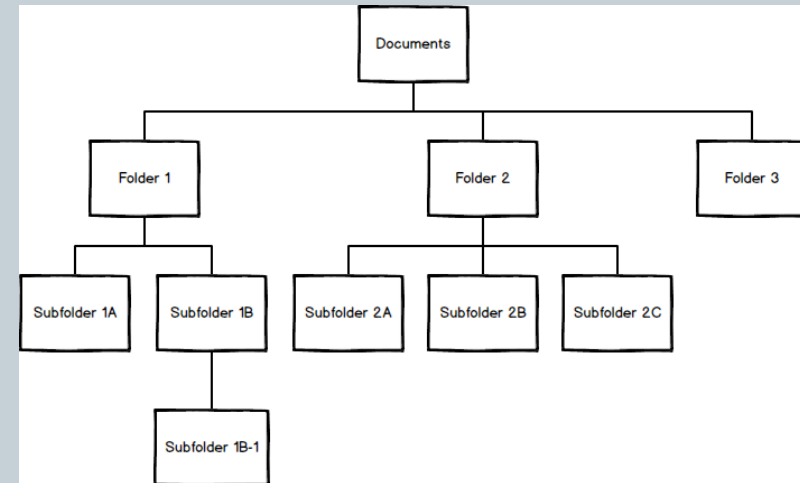
# Directory Structure

- “Folders”/Directories organized in a tree structure

- Root is at the top, branches below
- Files are stored in folders/directories
- On Windows, different devices
- have different letters
  - ✦ Primary hard drive is C:
  - ✦ Flash drives are usually E: onward

- Navigating the tree

- To change to a directory:
  - ✦ Windows: `cd C:\Documents\Folder 1\Subfolder 1A`
  - ✦ Linux: `cd “root/Documents/Folder 1/Subfolder 1B”`
  - ✦ Up one directory level: `cd ..`
  - ✦ The current directory: `.`
  - ✦ Where am I?
    - Linux: `pwd`
    - Windows: usually shown in the “prompt”



## Getting Around in the Command Shell

```
Administrator: Command Prompt
F:\CSCI135\Fall 2015\Workspace\03-CommandWindow>dir
Volume in drive F is Lexar
Volume Serial Number is 4FF2-8A1B

Directory of F:\CSCI135\Fall 2015\Workspace\03-CommandWindow

10/06/2016  11:43 AM    <DIR>          .
10/06/2016  11:43 AM    <DIR>          ..
10/06/2016  11:43 AM                392 .project
10/06/2016  11:43 AM    <DIR>          .settings
10/06/2016  11:43 AM                295 .classpath
10/06/2016  11:47 AM                558 AvgNums.java
10/06/2016  11:43 AM                365 RandomNums.java
               4 File(s)              1,610 bytes
               3 Dir(s)  23,167,860,736 bytes free

F:\CSCI135\Fall 2015\Workspace\03-CommandWindow>
```

```
katie.mtech.edu - PuTTY
ceskridge      helloworld      ksteinmetz      qstormer        twinsor
cfiechtner     hknight         lbblankenship   rbriese         vefoley
chand          hmuth           ljschuler       rcolling        vfoley
chansen        ht              lmullaney       rcunningham     wbrhyne
ckerins        janderson       lstout          rgratwohl       wcflynn
cladouceur     jbbaldwin       lswarner        rjmoon          wgildehaus
cmccclure      jbanderson      malnahwi        rkinsella       zbrown
cmlacey        jbeaudry        malobayshi      rmccarty        zburke
cmschahczenski jwareham        maustin         rpmitchell      zsmith
coneill        jdickson        mblotz          sbdeavours      zvalenzuela

vandyne@katie:/home/students$ cd $HOME
vandyne@katie:~$ ls
cars      CSCI460      Nostradamus.java  results.txt      zebra2.pl
cars.pl   gen_encrypt   Pennell           Stapley          zebra3
CSCI136   Morasco      Predictor.class   temp_gen         zebra3.pl
CSCI446   Nostradamus.class  Predictor.java    zebra            zebra.pl

vandyne@katie:~$ cd CSCI136
vandyne@katie:~/CSCI136$ ls
Dir with spaces  Lines.class      Lines.java        StdDraw.java
Line.class       LineServer.class MarsLander         StdIn.class
LineClient.class LineServer.java  Scratch.class     StdIn.java
LineClient.java  LineServerWorker.class Scratch.java
Line.java        LineServerWorker.java StdDraw.class

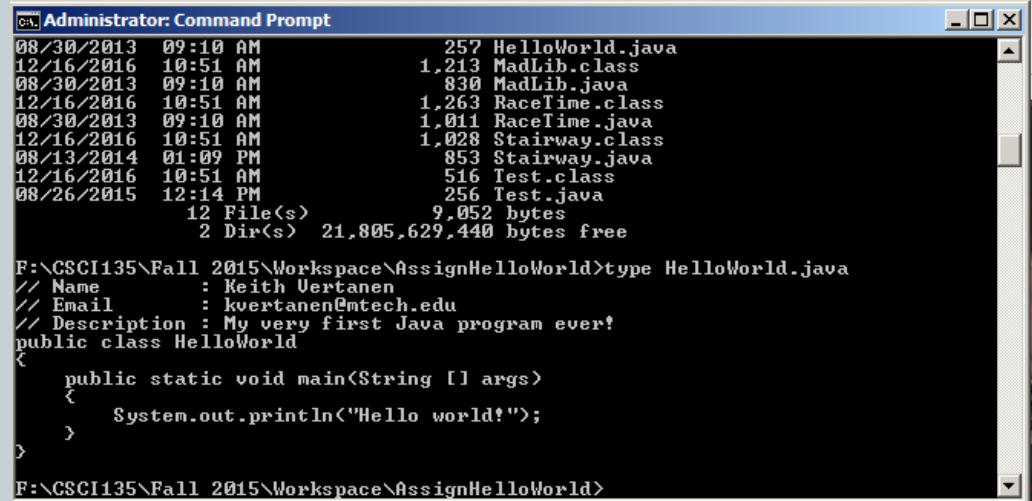
vandyne@katie:~/CSCI136$
```

Looking at the contents of a folder  
Windows: **dir**  
Linux: **ls**



# Displaying the Contents of Text File

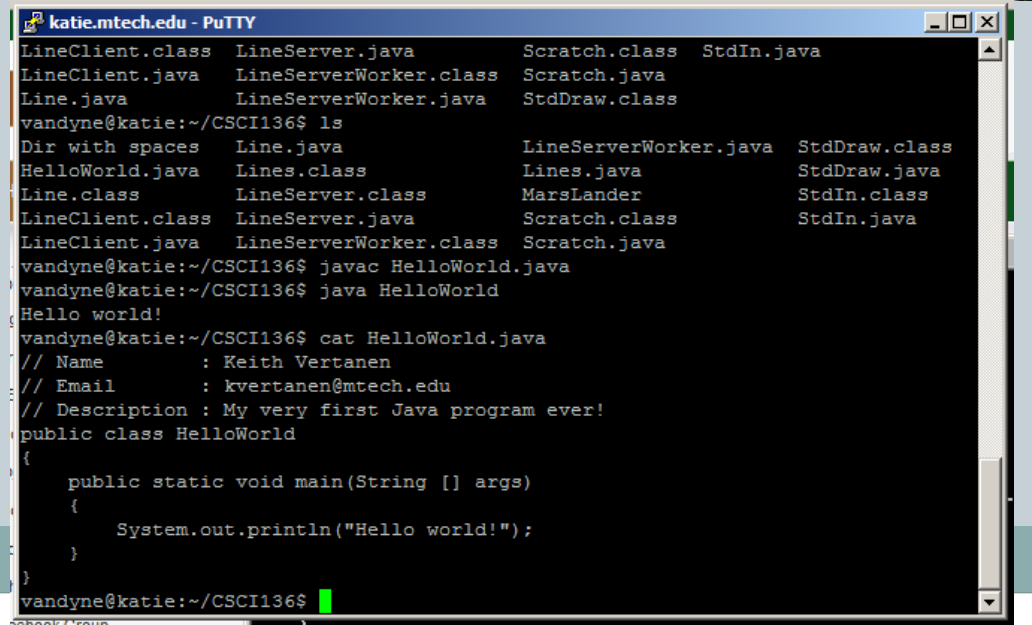
- Windows:
  - type HelloWorld.java



```
Administrator: Command Prompt
08/30/2013 09:10 AM 257 HelloWorld.java
12/16/2016 10:51 AM 1,213 MadLib.class
08/30/2013 09:10 AM 830 MadLib.java
12/16/2016 10:51 AM 1,263 RaceTime.class
08/30/2013 09:10 AM 1,011 RaceTime.java
12/16/2016 10:51 AM 1,028 Stairway.class
08/13/2014 01:09 PM 853 Stairway.java
12/16/2016 10:51 AM 516 Test.class
08/26/2015 12:14 PM 256 Test.java
12 File(s) 9,052 bytes
2 Dir(s) 21,805,629,440 bytes free

F:\CSCI135\Fall 2015\Workspace\AssignHelloWorld>type HelloWorld.java
// Name : Keith Vertanen
// Email : kvertanen@mtech.edu
// Description : My very first Java program ever!
public class HelloWorld
{
    public static void main(String [] args)
    {
        System.out.println("Hello world!");
    }
}
```

- Linux:
  - cat HelloWorld.java



```
katie.mtech.edu - PuTTY
LineClient.class LineServer.java Scratch.class StdIn.java
LineClient.java LineServerWorker.class Scratch.java
Line.java LineServerWorker.java StdDraw.class
vandyne@katie:~/CSCI136$ ls
Dir with spaces Line.java LineServerWorker.java StdDraw.class
HelloWorld.java Lines.class Lines.java StdDraw.java
Line.class LineServer.class MarsLander StdIn.class
LineClient.class LineServer.java Scratch.class StdIn.java
LineClient.java LineServerWorker.class Scratch.java
vandyne@katie:~/CSCI136$ javac HelloWorld.java
vandyne@katie:~/CSCI136$ java HelloWorld
Hello world!
vandyne@katie:~/CSCI136$ cat HelloWorld.java
// Name : Keith Vertanen
// Email : kvertanen@mtech.edu
// Description : My very first Java program ever!
public class HelloWorld
{
    public static void main(String [] args)
    {
        System.out.println("Hello world!");
    }
}
```

# Compiling a Java Program

- Windows or Linux:
  - `javac HelloWorld.java`
- If it all compiles correctly, you'll get back a prompt
  - The file `HelloWorld.class` is generated
- If things go wrong, you will get a list of compiler errors and associated line numbers
  - Not always where the error is, but will give you some clues – just like in Eclipse
- What if you want to compile multiple files?
  - `javac *.java`

# Running a Java Program

- Windows or Linux:
  - `java HelloWorld`
    - ✦ Note: you don't need to type the .class extension
- If it all runs correctly, you'll get the program results and a prompt
- If things go wrong, you will get a runtime error and the name of the exception that was thrown
  - Again, will give you some clues – just like in Eclipse
- You should run the file that contains the main method that you want – the JVM will find the rest of the files it needs

# Editing a Java Program

- Use any text editor
  - In Linux, vim
    - ✦ `vim HelloWorld.java`
    - ✦ Opens up an editor so you can change the code (or create a file)
    - ✦ Reference for vim commands:
      - <http://www.cmrr.umn.edu/~strupp/vim-3.0.refcard.html>
  - In Windows, any text editor
    - ✦ Notepad, Wordpad, Eclipse, etc.
      - Just be careful – some editors try to help you by giving the file a .txt extension when you save it – you need a .java extension in order to compile it

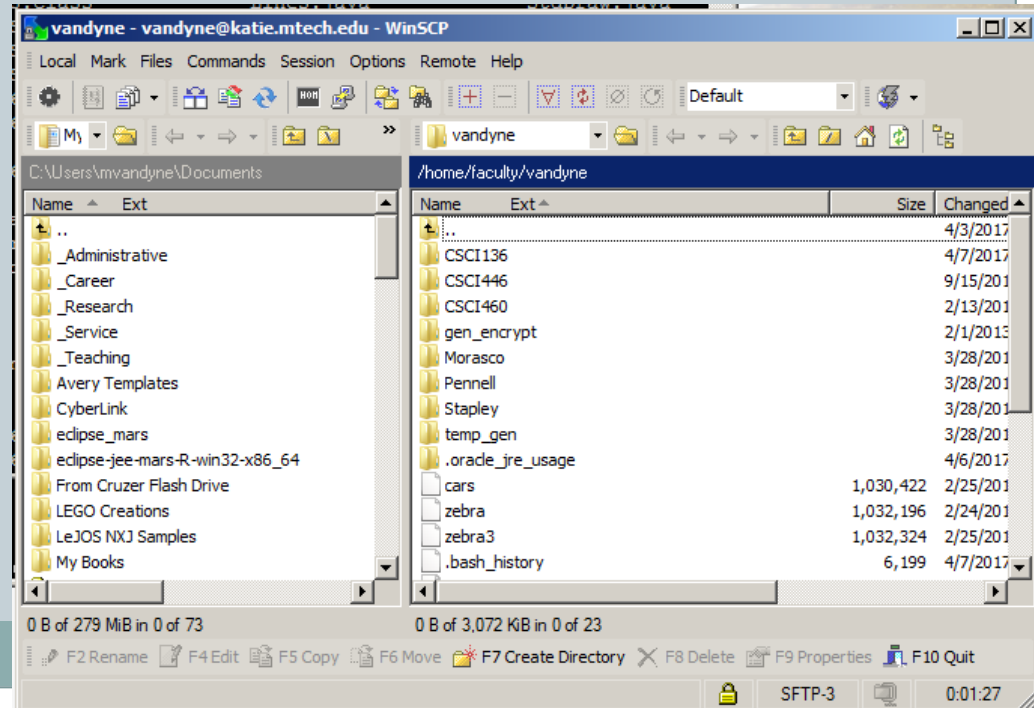
# Copying a File to Another Computer: winscp

---

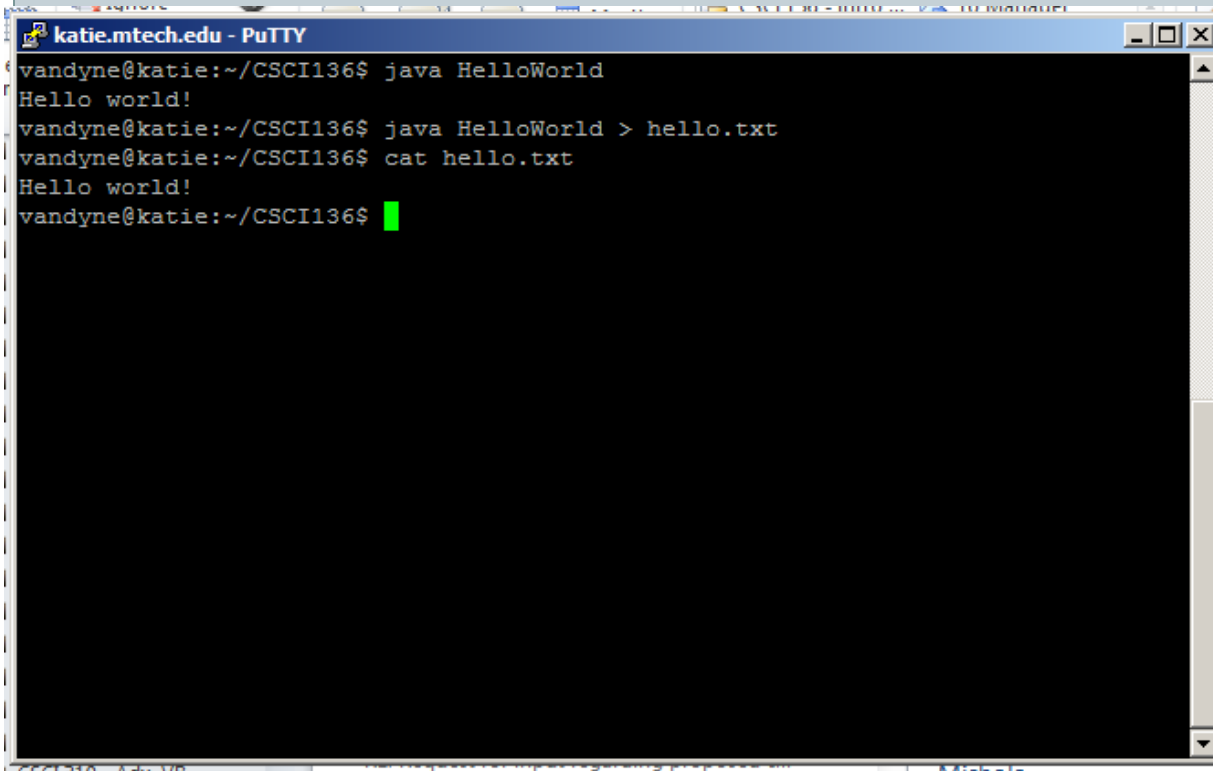
- Double click on winscp (you should have already downloaded it from the [cs.mtech.edu](http://cs.mtech.edu) website)
- Click the Run button
- Enter katie.mtech.edu for the host name
- Enter your user name in the User name box
- Enter your password in the Password box
- Hit Enter or click the Login button

# Copying a File to Another Computer: winscp (continued)

- You will see two panes in a window
  - The left side is your local computer
  - The right side is the computer you just logged in to
  - Can drag and drop files between the two sides
  - Or you can drag files from and to File Manager on your local computer



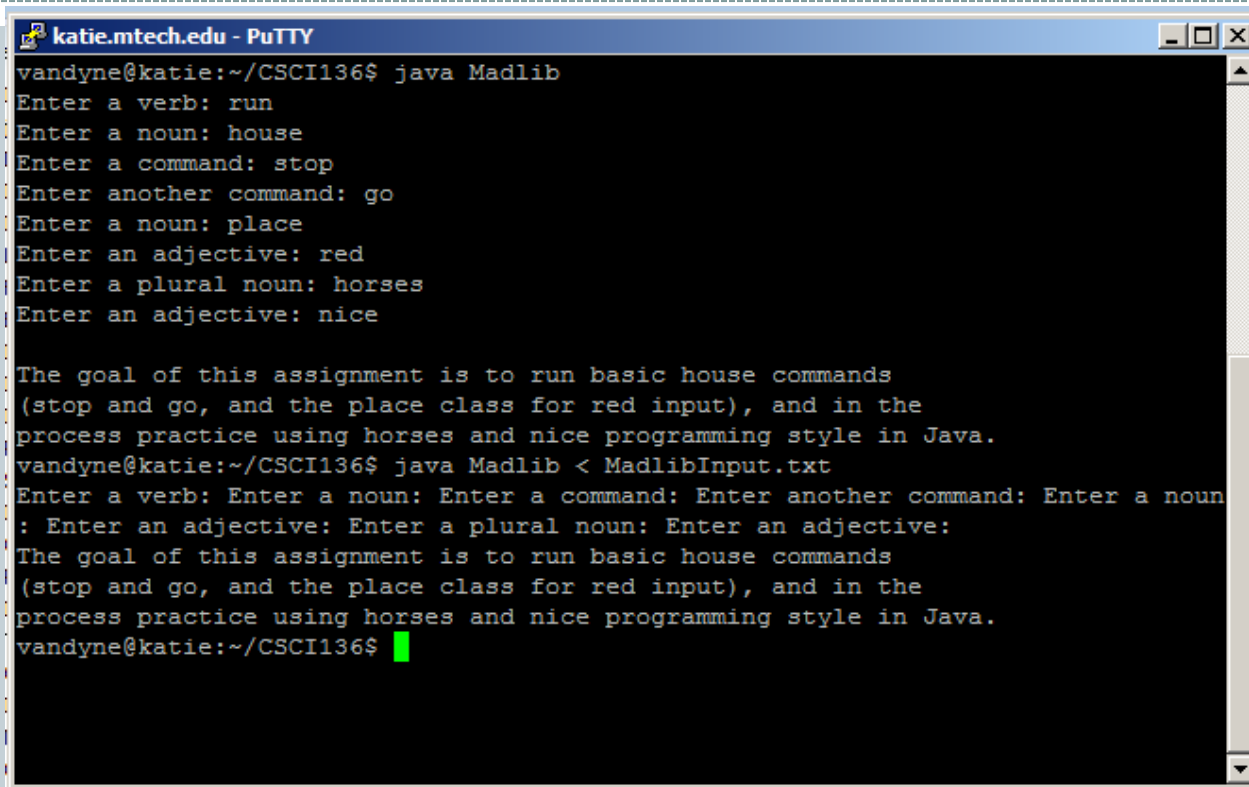
# Redirection

A screenshot of a PuTTY terminal window titled 'katie.mtech.edu - PuTTY'. The terminal shows a user named 'vandyne' at a prompt 'vandyne@katie:~/CSCI136\$'. The user enters 'java HelloWorld', and the output is 'Hello world!'. Then, the user enters 'java HelloWorld > hello.txt', and the output is redirected to a file named 'hello.txt'. Finally, the user enters 'cat hello.txt', and the output is 'Hello world!'. The terminal window has a black background and a green cursor.

```
katie.mtech.edu - PuTTY
vandyne@katie:~/CSCI136$ java HelloWorld
Hello world!
vandyne@katie:~/CSCI136$ java HelloWorld > hello.txt
vandyne@katie:~/CSCI136$ cat hello.txt
Hello world!
vandyne@katie:~/CSCI136$
```

Redirecting program  
**output to a file** using  
> followed by the  
output filename.

# Redirection



```
katie.mtech.edu - PuTTY
vandyne@katie:~/CSCI136$ java Madlib
Enter a verb: run
Enter a noun: house
Enter a command: stop
Enter another command: go
Enter a noun: place
Enter an adjective: red
Enter a plural noun: horses
Enter an adjective: nice

The goal of this assignment is to run basic house commands
(stop and go, and the place class for red input), and in the
process practice using horses and nice programming style in Java.
vandyne@katie:~/CSCI136$ java Madlib < MadlibInput.txt
Enter a verb: Enter a noun: Enter a command: Enter another command: Enter a noun:
: Enter an adjective: Enter a plural noun: Enter an adjective:
The goal of this assignment is to run basic house commands
(stop and go, and the place class for red input), and in the
process practice using horses and nice programming style in Java.
vandyne@katie:~/CSCI136$
```

Reading input from  
file using **<** followed  
by the filename.



# Piping



A screenshot of a PuTTY terminal window titled "katie.mtech.edu - PuTTY". The terminal shows a user named "vandyne" at a prompt "vandyne@katie:~/CSCI136\$". The user enters the command "java Madlib < MadlibInput.txt". The output of this command is a series of prompts: "Enter a verb: Enter a noun: Enter a command: Enter another command: Enter a noun:", "Enter an adjective: Enter a plural noun: Enter an adjective:", and a paragraph of text: "The goal of this assignment is to run basic house commands (stop and go, and the place class for red input), and in the process practice using horses and nice programming style in Java." The user then enters a second command: "java Madlib < MadlibInput.txt | java GerundFinder2". The output of this command is the word "programming". The terminal ends with the prompt "vandyne@katie:~/CSCI136\$" and a green cursor.

```
katie.mtech.edu - PuTTY
vandyne@katie:~/CSCI136$ java Madlib < MadlibInput.txt
Enter a verb: Enter a noun: Enter a command: Enter another command: Enter a noun:
: Enter an adjective: Enter a plural noun: Enter an adjective:
The goal of this assignment is to run basic house commands
(stop and go, and the place class for red input), and in the
process practice using horses and nice programming style in Java.
vandyne@katie:~/CSCI136$ java Madlib < MadlibInput.txt | java GerundFinder2
programming
vandyne@katie:~/CSCI136$
```

Using the output of  
one program as the  
input to another

Use the | operator

# Summary of Helpful Commands

Action	Windows	Mac OS / Unix
Move into a folder	<code>cd myfolder</code>	<code>cd myfolder</code>
Move into parent folder	<code>cd ..</code>	<code>cd ..</code>
Move into a folder, absolute folder	<code>cd \Users\keith</code>	<code>cd /Users/keith</code>
List files in current folder	<code>dir</code>	<code>ls</code>
Compile program in current folder	<code>javac Prog.java</code>	<code>javac Prog.java</code>
Run a compiled program	<code>java Prog</code>	<code>java Prog</code>
See what is in a text file	<code>type Prog.java</code>	<code>more Prog.java</code>
Auto-complete filenames	<code>&lt;tab key&gt;</code>	<code>&lt;tab key&gt;</code>
Previous command	<code>&lt;up arrow&gt;</code>	<code>&lt;up arrow&gt;</code>

# Summary

- Starting the Command Shell
  - Locally
  - Remote Host
- Directory Structure
  - Moving around the directories
- Displaying File Contents
- Compiling and Running a Java Program
- Editing a Text File
- Copying Files between Computers
- File Redirection and Piping
- Command Summary

