

CS1007 lecture #3 notes

tue 10 sep 2002

<http://www.columbia.edu/~cs1007>

today:

- news
- quick and dirty UNIX
- quick and dirty emacs
- creating/editing/compiling/running your first Java program
- homework #1
- submitting homework #1

news.

- Prof Sklar is out of town today and tomorrow so no office hours Tue
- make-up office hours: Fri Sep 13, 10am-11am

files in UNIX.

- hierarchical file system
- example:

```
      cunix_disk/eis2003/
      |
      +-----+-----+-----+
      |               |               |
      classes/        mail/        public_html/
      |               |               |
      +-----+-----+               |
      |               |               |
      cs1007/         cs1003/         index.html
      |               |               |
      hello.java      hello.c         |
      hello.class     |               |
                        |               |
                        simon          |
                        suzanne         |
                        jen             |
                        alex            |

```

quick and dirty UNIX.

- commands have options or parameters or switches
- *switches* start with “-”
- some commands...
 - man
 - pwd
 - cd
 - ls
 - mkdir
 - rmdir
 - cp
 - mv
 - rm
 - chmod

man — get help (display manual page).

man — display manual pages (get help!)

man man — display manual page for the *man* command

man ls — display manual page for the *ls* command

man -k file — list all commands with the keyword *file*

```
unix> man pwd
```

```
PWD(1)
```

```
FSF
```

```
PWD(1)
```

```
NAME
```

```
pwd - print name of current/working directory
```

```
SYNOPSIS
```

```
pwd [OPTION]
```

```
DESCRIPTION
```

```
Print the full filename of the current working directory.
```

```
...
```

pwd — print working directory.

```
unix> pwd  
/home/sklar/teaching/cs1007/slides
```

cd — change working directory.

```
unix> pwd
/home/sklar/
unix> cd classes
unix> pwd
/home/sklar/classes
```

ls — list the files in the current directory

ls -aF — list all files and show their file types

```
unix> ls -aF
```

```
./  
../  
.cshrc  
classes/  
mail/  
hello.java
```

ls -l — list files in long format

```
unix> ls -l hello.java
```

```
-rw-r--r--    1 sklar  faculty   187 Sep 5 10:45 hello.java
```

mkdir — make (create) a directory

```
unix> ls -aF
./
../
.cshrc
classes/
mail/
hello.java
unix> mkdir junk
unix> ls -aF
./
../
.cshrc
classes/
junk/
mail/
hello.java
```

rmdir — remove (delete) a directory.

```
unix> ls -aF
./
../
.cshrc
classes/
junk/
mail/
hello.java
unix> rmdir junk
unix> ls -aF
./
../
.cshrc
classes/
mail/
hello.java
```

cp — copy a file.

```
unix> ls -aF
./
../
.cshrc
classes/
mail/
hello.java
unix> cp hello.java hi.java
unix> ls -aF
./
../
.cshrc
classes/
mail/
hello.java
hi.java
```

mv — move (rename) a file.

```
unix> ls -aF
./
../
.cshrc
classes/
mail/
hello.java
unix> mv hello.java howdy.java
unix> ls -aF
./
../
.cshrc
classes/
mail/
howdy.java
```

rm — remove (delete) a file.

```
unix> ls -aF
./
../
.cshrc
classes/
mail/
hi.java
howdy.java
unix> rm hi.java
unix> ls -aF
./
../
.cshrc
classes/
mail/
howdy.java
```

chmod — change file mode

- 9 characters: -uuugggooo
- WHO: u = user, g = group, o = other users, a = all users (u + g + o)
- WHAT: r = read, w = write, x = execute
- MODE: + = allow, - = don't allow

```
unix> ls -l hi.java
-rwxr-xr-x    1 sklar  faculty    187 Sep  5 10:45 hi.java
unix> chmod a+w hi.java
unix> ls -l hi.java
-rwxrwxrwx    1 sklar  faculty    187 Sep  5 10:45 hi.java
```

quick and dirty *emacs*.

```
unix> ejava hello.java
```

or

```
unix> emacs -nw hello.java
```

Ctrl-B	move cursor Back
Ctrl-F	move cursor Forward
Ctrl-P	move cursor to Previous line
Ctrl-N	move cursor to Next line
Ctrl-D	Delete character under cursor
Ctrl-K	Kill (delete) to end of line
Ctrl-Y	Yank back (undelete) killed text
Ctrl-X Ctrl-S	Save the file
Ctrl-X Ctrl-C	eXit emacs
Ctrl-H	Help
Ctrl-G	Gets you out of trouble!

your first Java application: source code.

file name = hello.java

```
/* -----  
   Prof Sklar, 05-Sep-02, hello.java  
  
   This class demonstrates output from a Java application.  
-----*/  
public class hello {  
    public static void main ( String[] args ) {  
        System.out.println( "hello world!\n" );  
    } // end of main()  
} // end of class hello()
```

homework #1.

DUE TUE SEP 17!!!

- create a Java application that outputs instructions for how to change a lightbulb
- compile and run it
- submit it electronically

⇒ description available on web page:

<http://www.columbia.edu/~cs1007/hw1.html>