

# Ling 555 — Programming for Linguists

Python — File Input and Output

Robert Albert Felty

Speech Research Laboratory  
Indiana University

Oct. 6, 2008

homework

Theory

File I/O

## While we are waiting

Please download the courseBackground.txt file from:

[http://robfelty.com/teaching/L555Fall2008/  
resources/courseBackground.txt](http://robfelty.com/teaching/L555Fall2008/resources/courseBackground.txt)

```
curl -o courseBackground.txt
```

[http://robfelty.com/teaching/L555Fall2008/  
resources/courseBackground.txt](http://robfelty.com/teaching/L555Fall2008/resources/courseBackground.txt)

homework

Theory

File I/O

1 Homework questions

2 Theory

- Refactoring
- Style

3 File I/O

- files
- file-like

# Refactoring

homework

Theory

refactoring

style

File I/O

## Definition

Refactoring is re-writing and re-structuring code.

## Example

- In assignment 4, you were asked to re-factor some of the code from assignment 3. That is, you switched from using lists to dictionaries.
- In assignment 5, we will once again re-factor, taking advantage of loops, conditionals, and file input/output (I/O)

# Programming style

homework

Theory

refactoring

style

File I/O

## Definition

There's more than one way to do it. Some ways have advantages over others.

## Example

```
entries=`grep -E '^[A-Z]{2,},' devil.txt  
    |cut -f1 -d ','|wc -l`  
letters=`grep -E '^[A-Z]{2,},' devil.txt  
    |cut -f1 -d ','|wc -c`  
echo "$letters/$entries"|bc -l  
# OR, in one fell swoop  
echo "`grep -E '^[A-Z]{2,},' devil.txt  
    |cut -f1 -d ','|wc -c`/`grep  
-E '^[A-Z]{2,},' devil.txt |cut -f1 -d  
    ','|wc -l`"|bc -l
```

## File I/O — slurp

homework

Theory

File I/O

files

file-like

## Example

```
# python cat - read in file and spit  
it back out  
inFile='courseBackground.txt'  
fileObject=open(inFile,'r')  
outFile='courseBckgrndProcessed.txt'  
outFileObject=open(inFile,'w')  
contents=fileObject.read()  
fileObject.close()  
outFileObject.write(contents)  
outFileObject.close()
```

## File I/O — line by line

homework

Theory

File I/O

files

file-like

## Example

```
inFile='courseBackground.txt'  
fileObject=open(inFile,'r')  
outFile='courseBckgrndProcessed.txt'  
outFileObject=open(inFile,'w')  
for line in fileObject:  
    modified=processLine(line)  
    outFileObject.write(modified)
```

## file-like — slurp

homework

Theory

File I/O

files

file-like

## Definition

File-like objects can be written to and read from just like files. The ones we will use most often are stdin, stdout, and stderr.

## Example

```
# python cat - read in file and spit  
    it back out  
import sys  
contents=sys.stdin.read()  
sys.stdout.write(contents)
```



## Example

```
# python process line by line
import sys
for line in sys.stdin:
    modified=processLine(line)
    sys.stdout.write(modified)
```

homework

Theory

File I/O

files

file-like

## Example

```
./slurp.py < input.txt > output.txt
```