

Using the command line



VERSION .03

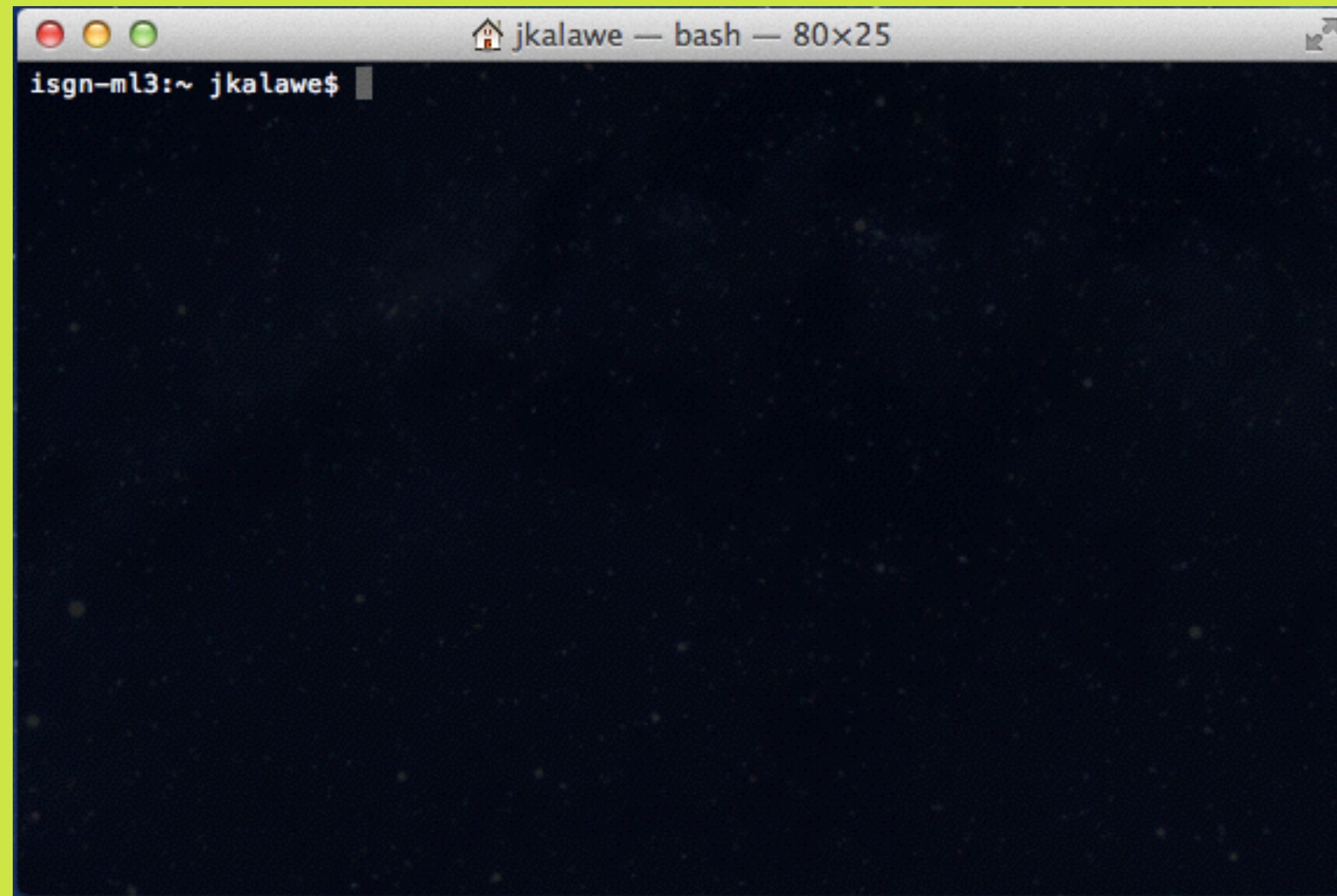
A black and white photograph of a puppy, likely a Labrador Retriever, looking up at the camera with a curious expression. The puppy is positioned in the upper left quadrant of the frame, with its head tilted slightly upwards and to the right. Its dark fur is contrasted against a lighter background.

unix is your
special friend

The Terminal

(don't be scared)





isgn-ml3:~ jkalawe\$





input

output

FOLLOW ALONG

commands take input to create output

```
echo
```

commands can take arguments

```
echo "this is an argument"
```



FOLLOW ALONG

OUTPUT YOUR USERNAME

```
whoami
```

OUTPUT COMPUTER UPTIME

```
uptime
```



FOLLOW ALONG

OUTPUT YOUR OWN TEXT

echo UNIX's is cool!

ESCAPE THE TEXT THIS TIME

echo “Unix’s is cool!”



Escaping Text



FOLLOW ALONG

USE QUOTES FOR TEXT

```
echo "This is my text"
```

USE BACKTICKS IF IT'S A COMMAND

```
echo "Today's date is " `date`
```



A black and white photograph of a man with glasses looking thoughtfully at a stack of books on a shelf in a library.

the
filesystem

TERMINOLOGY

Path

Location in the filesystem of a file

Slashes are used to denote depth

/Users/yourusername



TERMINOLOGY

SAMPLE PATH

```
/Users/jkalawe/Sites/coding100
```

The first slash is known as the ROOT

Every slash represents depth

The last part of the path is either a FILE OR DIRECTORY

coding100 is the child of 3 parents



FOLLOW ALONG

OUTPUT YOUR CURRENT PATH

```
pwd
```



FOLLOW ALONG

LIST FILES IN THE CURRENT DIRECTORY

```
ls
```

OUTPUT A DETAILED LIST

```
ls -l
```



Home directory



TERMINOLOGY

HOME DIRECTORY

Your user account directory

~ (tilde) represents YOUR home directory



FOLLOW ALONG

LIST FILES IN YOUR HOME DIRECTORY

```
ls ~
```



Navigation



TERMINOLOGY

cd

Use cd to change directories



FOLLOW ALONG

Change the directory to your documents

```
cd ~/Documents
```



TERMINOLOGY

. (PERIOD)

Represents the current directory

.. /

Represents the parent of the current directory



FOLLOW ALONG

LIST THE FILES IN THE PARENT DIRECTORY

```
ls ..
```

LIST FILES IN THE PARENT'S PARENT

```
ls ../../
```



FOLLOW ALONG

Change into the parent directory

```
cd ..
```



Create Files & Directories



FOLLOW ALONG

MAKE a diRECTORY

```
mkdir prg-01
```

change into this diRECTORY

```
cd prg-01
```



FOLLOW ALONG

CREATE A FILE

```
touch test.txt
```

REMOVE THE FILE

```
rm test.txt
```



Redirection



TERMINOLOGY

Redirection

By default output is sent to the terminal

Redirection instead sends output to a command or file



FOLLOW ALONG

Redirect text to a file using >

```
echo "First line" > test.txt
```

Append more text to a file using >>

```
echo "Second line" >> test.txt
```

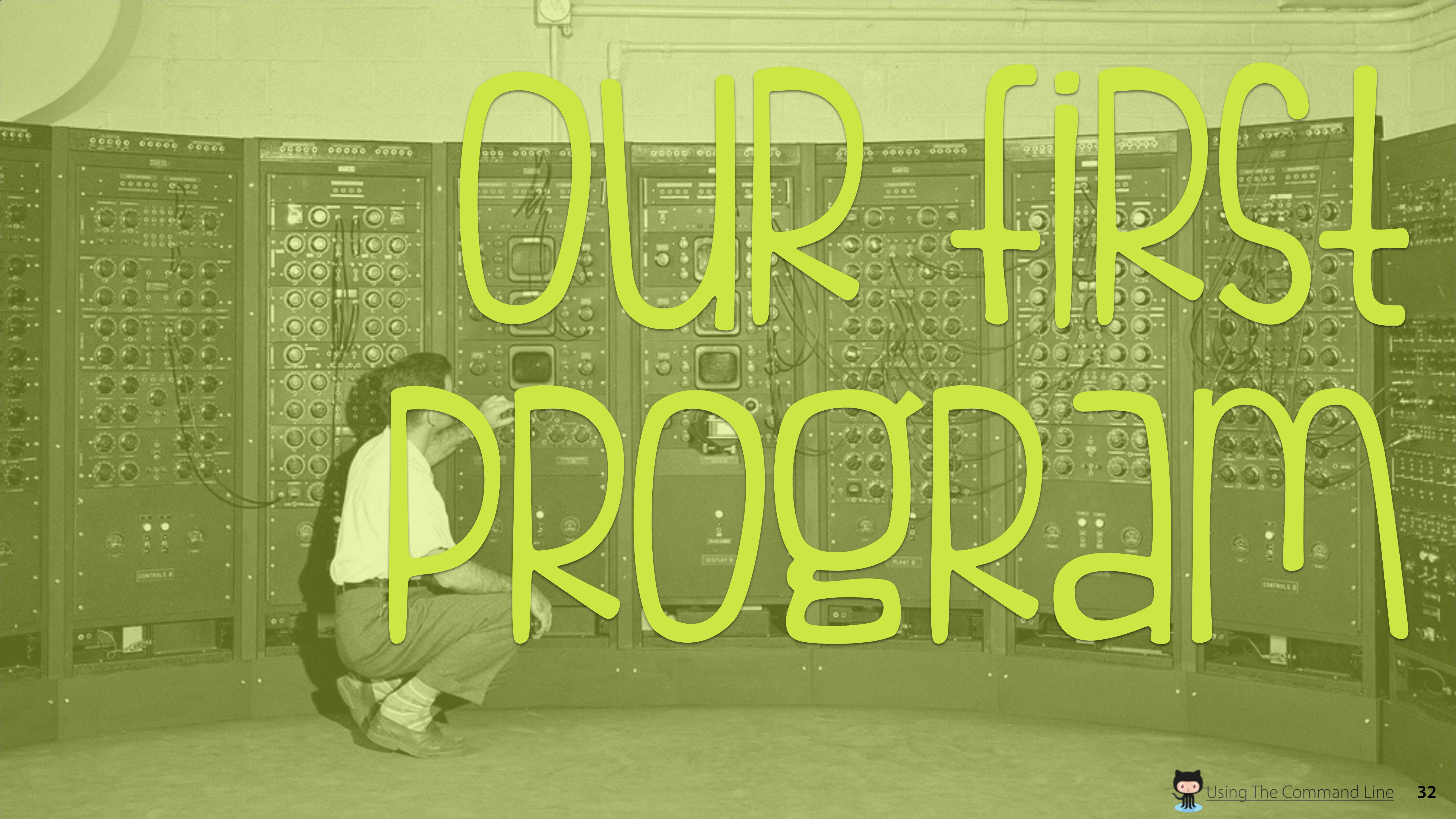


FOLLOW ALONG

view the contents of a file

```
cat test.txt
```





our first
program



TERMINOLOGY

COMPUTER PROGRAM

Just a bunch of commands in a text file



FOLLOW ALONG

OUR PROGRAM WILL

OUTPUT TEXT

TO A FILE

FOLLOW ALONG

LET'S MAKE A PROJECT DIRECTORY

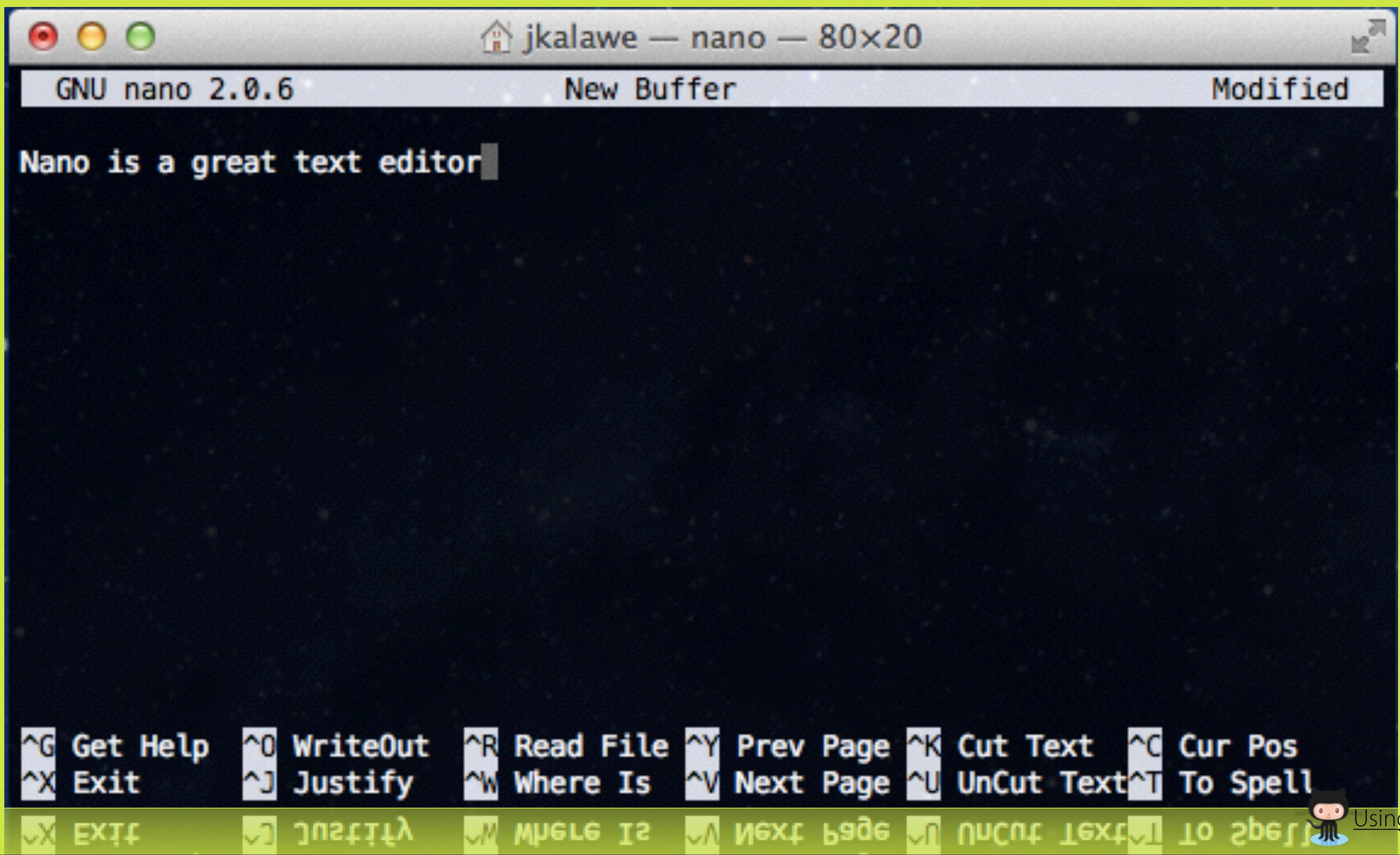
```
mkdir ~/Documents/Coding100
```

GO TO OUR PROJECT DIRECTORY

```
cd ~/Documents/Coding100
```



nano is a text editor



^G Get Help
^X Exit

^O WriteOut
^J Justify

^R Read File
^W Where Is

^Y Prev Page
^V Next Page

^K Cut Text
^U UnCut Text

^C Cur Pos
^T To Spell



FOLLOW ALONG

LET'S WRITE OUR PROGRAM USING nano

```
nano hello.bash
```



FOLLOW ALONG

Identify the Language We're Using

```
#!/bin/bash
```

WRITE TEXT TO A TEXT FILE

```
echo 'hello world' > hello.txt
```



FOLLOW ALONG

WRiTE another Line

```
echo 'hello again' >> hello.txt
```

LET'S SAVE OUR file

```
type 'control' and 'o'
```



FOLLOW ALONG

Exit nano

type ‘control’ and ‘x’



TEST OUR
PROGRAM



FOLLOW ALONG

Run the program using bash

```
bash hello.bash
```



FOLLOW ALONG

check if our program wrote the file

```
ls
```

check contents of file

```
nano hello.txt
```





Questions?
Need help? [Post an issue @github](#)