

## A quick primer of linux and use the Kate editor.

### Linux quick primer:

Linux is command line based (but has windows)

cd changes directory

cd Documents willchange to the sub directory Documents

cd .. changes to the parent directory

rm removes a file

rm myfile.txt will delete the file myfile.txt

mv will move a file and/or rename a file

mv myfile.txt Documents/. Moves the file myfile.txt to the directory Documents.

Note the period is required.

mv myfile.txt Documents/myotherfile.txt

Moves the file myfile.txt to the subdirectory Documents and renames the file

mv myfile.txt myotherfile.txt

changes the name of myfile.txt to myotherfile.txt

cp is the copy command

cp myfile.txt Documents/. Copies the file to the subdirectory Documents  
again the period is required.

cp myfile.txt Documents/anotherfile.txt Copies the file and names it anotherfile.txt

ls lists the directories and files in the current directory

ls -la will list all files including dot files (such as .cshrc) in a column format.

whereis <filename> will search the path and if found, where you where the filename is found

whereis kate

kate: /usr/bin/kate

whereis firefox

firefox: /usr/bin/firefox /usr/local/bin/firefox /usr/local/firefox4.0 /usr/local/firefox

which <filename> will tell you which filename will be run.

which firefox

/usr/local/bin/firefox

input and output redirections

file >filename.out redirects any output from the program file to the file filename.out It should be noted, the error message (stderr) may not be redirected.

file <filename.in redirects the input to the filename.in

file <filename.in >filename.out The input to the program file comes the filename.in and directs the output to filename.out.

**Compiling in c++ and running from the command line. the compiler is called g++**

g++ filename.cpp

This will attempt to compile the file and produces a executable called a.out

g++ filename.cpp -o filename

this will attempt to compile the file and produces a executable called filename  
To run the files at the command line (note you may have to have the ./ as part of the filename).  
./a.out OR ./filename depending on how you compiled it.

### **Java compiling and running from the command line.**

javac filename.java

this will compile the java file and produces a file named filename.class

java filename (do not include the .class)

this will run the program.

## **Kate overview**

kate is a simple editor environment. It will do syntax highlighting, based on the file name extension. It has a command line window, plus it allows you have multiple files open at the same time. The editor is pretty straight forward with cut and paste and most feature you would expect to find. If you want to use the vim style editor Settings=>Configure Kate .. => Editing =>VI Input Mode

To launch kate, at the command prompt: kate

In the left pane is your file list. In the right top windows is the open file

At the bottom look for Terminal and click it. This open the terminal window. Click it again to hide the terminal window.

It should start with an Untitled document. You can save that as the file you want to start the Syntax highlighting.

### **C++ demo:**

For this save it as helloicpc.cpp

Type in the following code:

```
#include <iostream>
using namespace std;

int main() {
    cout<<"hello ICPC\n";
    return 0;
}
```

Now save the code again. In the command window (or other terminal) you can compile and run the code. You may need to cd in the subdirectory (example cd Documents)

```
g++ helloicpc.cpp -o helloicpc
helloicpc
```

I would suggest you go back and make syntax error and compile again, so you can see the error messages. A note with g++, always try and fix the top/first message first. The other error message may go away. And don't forget to check the warning messages as well!

### **Java Demo:**

Click the new button to get another Untitled document. Save it as helloicpc.java

Type in the following code:

```
class helloicpc {  
    public static void main(String[] args) {  
        System.out.println("Hello ICPC"); // Display the string.  
    }  
}
```

Now save the code again. In the command windows (or other terminal) compile and run the java code

```
javac helloicpc.java
```

```
java helloicpc
```

Again, I would suggest you make a syntax error, so you can see the compiler error messages.