

---

# Writing to files

## COMP 102

Victoria University of Wellington

# Files

---

- The UI text pane window is transient:
  - Typing large amounts of input into the text pane is a pain!
  - It would be nice to be able to save the output of the program easily.
- Large amounts of text belong in files
- How can your program read from a file and write to a file?

# Files

---

- Writing to files is like writing to the UI text pane!
  - Use print, println, printf methods
  - But, need a **PrintStream** object instead of UI
- Reading from files is a bit different
  - Doesn't use "ask..." methods
  - Lots of different ways of reading from files
  - We will just use a very simple one that reads a list of lines from a text file
  - We will use **Scanner** objects to break up the lines into separate values.

# Text with Files

Writing data to a file:

- Use a `PrintStream` object.

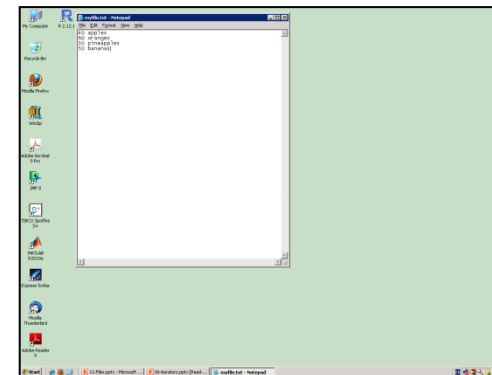
## My Program

```
:  
int r =scan.nextInt();  
int g =scan.nextInt();  
int b =scan.nextInt();  
UI.setColor(new Color(r,g,b));  
:  
outFile.println("all done");
```

print...

**PrintStream  
object**

A real file: "output.txt"



# Writing to a file:

---

```
/** Read lines from a user and print them to a file. */  
public void makeFile(String filename){  
    ArrayList<String> lines = UI.askStrings("Type in file contents:");  
    try {  
        PrintStream outfile = new PrintStream(filename);  
        for (String line : lines) {  
            outfile.println(line);  
        }  
        outfile.close();  
    } catch (IOException e) { UI.println("File failure: " + e); }  
}
```

- **PrintStreams** work just like printing to UI
- Close the file when finished.
- Need a `try { ... } catch (...){....}` around printing to files also.

# Writing to a file, using `UIFileChooser`

```
/** Read lines from a user and print them to a file. */  
public void makeFile(){  
    //Find full filename  
    String filename = UIFileChooser.save("Filename to save to");  
    makeFile(filename);  
}  
public void makeFile(String filename){  
    ArrayList<String> lines = UI.askStrings("Type in file contents:");  
    try {  
        PrintStream outfile = new PrintStream(filename);  
        for (String line : lines) {  
            outfile.println(line);  
        }  
        outfile.close();  
    } catch (IOException e) { UI.println("File failure: " + e); }  
}
```

# Writing to a file, using UIFileChooser

```
/** Read lines from a user and print them to a file. */
```

```
public void makeFile(){
```

```
    //Find position and data to save to file
```

```
    makeFile(UIFileChooser.save("Filename to save to"), UI.askStrings("Type in file contents:"));
}
```

```
public void makeFile(String filename, ArrayList<String> lines){
```

```
    try {
```

```
        PrintStream outfile = new PrintStream(filename);
```

```
        for (String line : lines) {
```

```
            outfile.println(line);
```

```
        }
```

```
        outfile.close();
```

```
    } catch (IOException e) { UI.println("File failure: " + e); }
}
```