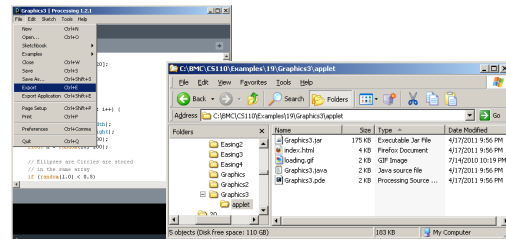


Processing == Java + Extra Utilities

Processing Adds:

- Drawing functions
- Text and font manipulations
- Image and video
- 3D transformations
- Keyboard interaction
- An editor
- ...

- Best way to see what Processing becomes is to generate a Java program and compare
 - File | Export
 - Look in generated applet folder



• What's different?

- Import of core Processing libraries
- Import of numerous Java libraries
- Encapsulation of code in subclass of PApplet
- Explicit visibility – public, private ...
- Number format changes


```
if (random(1.0) < 0.5) -> if (random(1.0f) < 0.5f)
```
- color data types are converted to int
- Some functions are changed


```
numbers = int(snumbers); ->
numbers = PApplet.parseInt(snumbers);
```
- Added `public static void main()` {...

The smallest program in Processing

```
println("Hello World!");
```

The equivalent program in Java.

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

All Java programs start with at the special function ...

```
public static void main() { ...
```

This must be added to the generated Java program

• PApplet

- The top-level class in Processing
- The class in which all your Processing code goes
- Implements most of Processing's core behavior
- Way down the Java hierarchy

```
java.lang.Object
├── java.awt.Component
│   ├── java.awt.Container
│   │   ├── java.awt.Panel
│   │   │   ├── java.applet.Applet
│   │   │   └── processing.core.PApplet
```

<http://processing.googlecode.com/svn/trunk/processing/build/javadoc/core/processing/core/PApplet.html>

Nearly all of Java is available from Processing

<http://download.oracle.com/javase/6/docs/api/>