Getting Started in Java

CIS 110
Your First Program

```java
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World");
    }
}
```

Section 1.1
Your First Program

Program Name

Section 1.1
Your First Program

Section 1.1

Scaffolding
Your First Program

Print the text "Hello, World".

Section 1.1
Your First Program

Statements end with a ;

Section 1.1
Your First Program

Compile to translate to machine code

Section 1.1
Your First Program

Run the compiled program

```java
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World");
    }
}
```

Welcome to DrJava. Working directory is /Users/bjgbrown/introcs
> run HelloWorld
Hello, World

Why Java?

Java

```java
public class Hello World {
    public static void main(String[] args) {
        System.out.println("Hello, World.");
    }
}
```

C/C++

```c
#include <stdio.h>

int main(int argc, char** argv) {
    printf("Hello, World.\n");
    return 0;
}
```

Matlab

```
disp('Hello, World.)
```

JavaScript

```
document.write("Hello, World.");
```

Fortran

```
PROGRAM HELLO
PRINT *, 'Hello, World.'
END
```

OCaml

```
print_endline "Hello, World."
```

Lisp

```
(princ "Hello, World.")
```

sh

```
echo Hello, World.
```

Logo

```
print [Hello, World.]```

Java is:
- Widely used
- Practical for many problems
- Includes most modern language abstractions

"There are only two kinds of [programming] languages: the ones people complain about and the ones nobody uses."  - Bjarne Stroustrup
Why Not English?

Kids Make Nutritious Snacks
Red Tape Holds Up New Bridge
Police Squad Helps Dog Bite Victim
Local High School Dropouts Cut in Half
[real newspaper headlines compiled by Rich Pattis]

'At Last' Singer Etta James Terminally Ill
[New York Time Online, 16 December 2011]

"Santorum is an Unpleasant By-Product of Sex"
[deredactie.be, 6 January 2012]
Computational Art

Examples
Protobytes by Ira Greenberg
Shepard Fairey
Abstract

Art
Red & Blue States
Summertime,
And the livin' is easy
Fish are jumpin'
And the cotton is high

Your daddy's rich
And your mamma's good lookin'
So hush little baby
Don't you cry

One of these mornings
You're going to rise up singing
Then you'll spread your wings
And you'll take to the sky

But till that morning
There's a'nothing can harm you
With daddy and mamma standing by

Summertime,
And the livin' is easy
Fish are jumpin'
And the cotton is high

Your daddy's rich
And your mamma's good lookin'
So hush little baby
Don't you cry

Lyrics by George Gershwin
Drawing in Java Using the StdDraw Library: MyHouse.java

CIS 110
public class MyMouse {
    public static void main(String[] args) {
        //set the size of the window to 500 pixels by 500 pixels
        StdDraw.setCanvasSize(500, 500);
        StdDraw.clear(StdDraw.BLUE); // draw a blue sky
public class MyHouse {
    public static void main(String[] args) {
        // set the size of the window to 500 pixels by 500 pixels
        StdDraw.setCanvasSize(500, 500);
        StdDraw.clear(StdDraw.BLUE); // draw a blue sky
Color the entire window blue

```java
public class MyHouse {
    public static void main(String[] args) {
        // set the size of the window to 500 pixels by 500 pixels
        StdDraw.setCanvasSize(500, 500);
        StdDraw.clear(StdDraw.BLUE); // draw a blue sky
    }
}
```
public class MyHouse {
    public static void main(String[] args) {
        // set the size of the window to 500 pixels by 500 pixels
        StdDraw.setCanvasSize(500, 500);

        StdDraw.clear(StdDraw.BLUE); // draw a blue sky
    }
}
Set the color to grass green

```java
//draw a green field
StdDraw.setPenColor(0, 170, 0);
StdDraw.filledRectangle(0.5, 0.25, 0.6, 0.3);
```
Colors

Composed of three elements:

1. Red
2. Green
3. Blue

Values from 0 .. 255
### Why 0 ... 255?

Each color is represented by 32 bits:

<table>
<thead>
<tr>
<th>Decimal</th>
<th>Binary</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>00000000</td>
</tr>
<tr>
<td>1</td>
<td>00000001</td>
</tr>
<tr>
<td>2</td>
<td>00000010</td>
</tr>
<tr>
<td>3</td>
<td>00000011</td>
</tr>
<tr>
<td>4</td>
<td>00000100</td>
</tr>
<tr>
<td>5</td>
<td>00000101</td>
</tr>
<tr>
<td>6</td>
<td>00000110</td>
</tr>
<tr>
<td>7</td>
<td>00000111</td>
</tr>
<tr>
<td>8</td>
<td>00001000</td>
</tr>
<tr>
<td>9</td>
<td>00001001</td>
</tr>
<tr>
<td>10</td>
<td>00001010</td>
</tr>
<tr>
<td>11</td>
<td>00001011</td>
</tr>
<tr>
<td>12</td>
<td>00001100</td>
</tr>
<tr>
<td>13</td>
<td>00001101</td>
</tr>
<tr>
<td>14</td>
<td>00001110</td>
</tr>
<tr>
<td>15</td>
<td>00001111</td>
</tr>
<tr>
<td>16</td>
<td>00010000</td>
</tr>
<tr>
<td>17</td>
<td>00010001</td>
</tr>
<tr>
<td>18</td>
<td>00010010</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>255</td>
<td>11111111</td>
</tr>
</tbody>
</table>

Notice there are 8 bits per color component.

The maximum value (all 1’s) that can be represented in 8 bits is 255 in decimal.

Therefore, the range for each color component is 0 (off) ... 255 (full).
Set the color to grass green

```java
// draw a green field
StdDraw.setPenColor(0, 170, 0);
StdDraw.filledRectangle(0.5, 0.25, 0.6, 0.3);
```
Solid rectangle

//draw green field
StdDraw.setPenColor(0, 0.17, 0);
StdDraw.filledRectangle(0.5, 0.25, 0.6, 0.3);
//draw a green field
StdDraw.setPenColor(0, 170, 0);
StdDraw.filledRectangle(0.5, 0.25, 0.6, 0.3);
Coordinate System

(0, 0)  1.0  +x

+y
Lists of 3 x- and y-coordinates

double[] x = {0.255, 0.745, 0.49};
double[] y = {0.70, 0.70, 0.90};
StdDraw.filledPolygon(x, y);

We'll explain this syntax in a couple of weeks.
Draw a solid triangle with corners at (0.255, 0.7), (0.745, 0.7), (0.49, 0.9)

double[] x = {0.255, 0.745, 0.49};
double[] y = {0.70, 0.70, 0.90};
StdDraw.filledPolygon(x, y);
Set line thickness (default is 0.002)

```java
StdDraw.setPenRadius(0.005); // thicken the pen for outline drawing
```
Draw a rectangle outline

```
StdDraw.rectangle(250 / 500.0, 260 / 500.0, 120 / 500.0, 90 / 500.0);
```
Keep repeating the instructions in this block forever

```java
// draw a circular cloud at the mouse location as long
// as the mouse is within bounds
while (true) {
    double cloudX = StdDraw.mouseX();
    double cloudY = StdDraw.mouseY();
    StdDraw.setPenColor(StdDraw.WHITE);
    if (cloudY > 0.55) {
        StdDraw.filledCircle(cloudX, cloudY, 0.005);
    }
    StdDraw.show(30);
}
```

We'll explain this syntax soon too.
Check the mouse's x- and y-coordinates. Call them cloudX and cloudY.

```java
// draw a circular cloud at the mouse location as long
// as the mouse is within bounds
while (true) {
    double cloudX = StdDraw.mouseX();
    double cloudY = StdDraw.mouseY();
    StdDraw.setPenColor(StdDraw.WHITE);
    if (cloudY > 0.55) {
        StdDraw.filledCircle(cloudX, cloudY, 0.005);
    }
    StdDraw.show(30);
}
```
Draw a circle centered at the cursor with radius 0.005, only if the y-coordinate is greater than 0.55!

```java
// draw a circular cloud at the mouse location as long as the mouse is within bounds
while (true) {
    double cloudX = StdDraw.mouseX();
    double cloudY = StdDraw.mouseY();
    StdDraw.setPenColor(StdDraw.WHITE);
    if (cloudY > 0.55) {
        StdDraw.filledCircle(cloudX, cloudY, 0.005);
    }
    StdDraw.show(50);
}
```

We'll explain this syntax soon too.
Show the changes we just made; Wait to show any further changes until we encounter `StdDraw.show()` again and at least 30 milliseconds have past.

```java
// draw a circular cloud at the mouse location as long // as the mouse is within bounds
while (true) {
    double cloudX = StdDraw.mouseX();
    double cloudY = StdDraw.mouseY();
    StdDraw.setPenColor(StdDraw.WHITE);
    if (cloudY > 0.55) {
        StdDraw.filledCircle(cloudX, cloudY, 0.005);
    }
}
StdDraw.show(30);
```