Let's Make a Game!

Simple game development with Slick2D and Java

Slick2D

- Lightweight 2D game engine
- Built on top of LWJGL
- Takes care of the basics

- http://slick.ninjacave.com/
- http://slick.ninjacave.com/wiki

Setting up Slick2D in jGRASP

- You'll need to have the Slick2D project downloaded and unzipped from http://slick.ninjacave.com/slick.zip (or the class website)
- Open jGRASP and open the menu

  Settings -> PATH / CLASSPATH -> Workspace

- Under the PATHS tab, click New and add the **slick** folder.
- Under the CLASSPATHS tab, add the following files from slick/lib:

  slick.jar, jinput.jar, lwjgl_util.jar, lwjgl.jar

Setting up Slick2D in Eclipse

- See the instructions at http://slick.ninjacave.com/wiki/index.php?title=Setting_up_Slick2D_with_Eclipse
  - Look at the section titled **Setting Up Slick2D and LWJGL in Eclipse**

- jGRASP works fine, but I recommend learning Eclipse.

Brief Math Time: 2D Vectors

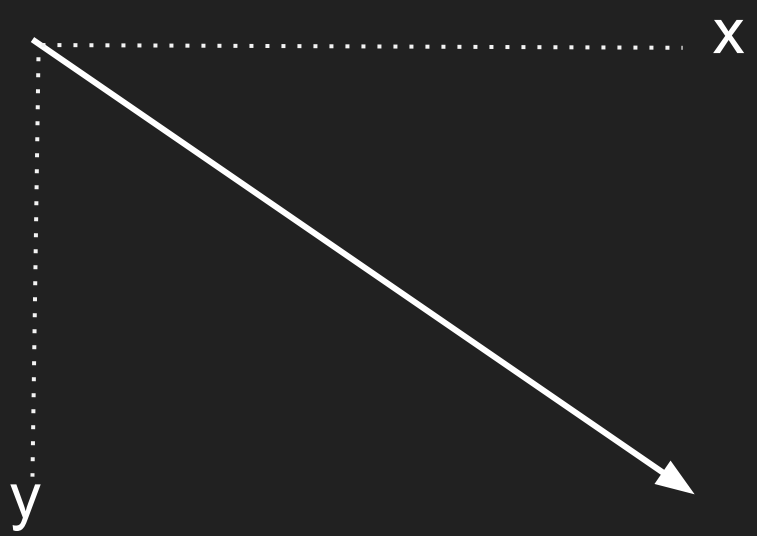
- Encodes a 2D direction and distance.
- <x, y>
- Very useful for game programming
- Can be added to each other:

  <x, y> + <a, b> = <x + a, y + b>

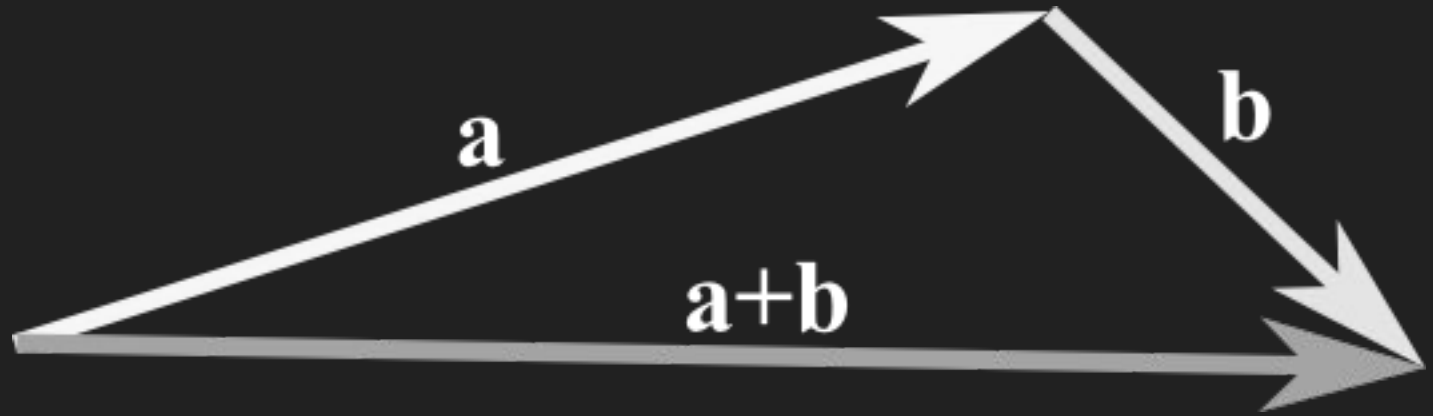
- In Slick2D:

```
new Vector2f(x, y);
```

- In Slick2D, y is *down*.



Brief Math Time: 2D Vectors



2D Transformations

- A bit tricky, but very useful!
- 3 main ones
  - `rotate(x, y, angle);`
  - `translate(x, y);`
  - `scale(x, y);`
- Apply to *every subsequent thing drawn*, in ***reverse order!!!***
  - I.e. rotate(...); translate(...); means translate and then rotate.
  - Because of the way the math works internally.
- Remove all transformations with `resetTransform();`

Time to write some code!