

Java

Eclipse

- www.eclipse.org
- Opensource/free

Terminology

- Machine language
 - Ones and zeros
 - Can be data or instructions
 - Machine specific (Mac, PC, android)
- Programming language
 - A collection of syntax and rules that are easier for humans to understand
 - Allows a programmer to specify ANYTHING that is possible in machine language
 - Java, Python, C++ qualify. Matlab, javascript do not
 - NOT machine agnostic, unless the language allows it (Java)
- Compiler (C++)
 - A special tool/program for converting programming language into machine language
 - This is done once, then the machine language (.exe) can be run any time
- Interpreter(Python)
 - A special tool that turns program into machine code while you run it.
- Virtual machine (Java)
 - A mid level step . Compile to a fast virtual machine, but save machine language for run time.
 - This middle step is what allows Java to work on many OSs

Terminology

- Integrated development environment (IDE)
 - A software package for compiling/interpreting, debugging and organizing code
 - You can write code using a text editor, but I wouldn't advise it

Types of programming languages

- Assembly languages
 - Barely above machine code
- Functional/Logic/Other
 - Specialty styles, not discussed
- Procedural
 - Separation of procedure and data
 - Procedures operate on data
 - C is a procedural language.
 - Matlab simulates this style of programming
- Object oriented
 - Bundles procedure and data into Objects
 - Objects contain their own data and their own procedures to operate on that data
- **DRASTIC OVERSIMPLIFICATION:**
 - procedural has less overhead for simple programs
 - Object oriented provides more structure for larger complicated programs
- Visual programming languages
 - Prograph. These have yet to catch on, unfortunately

Typing

- Data and information is stored, manipulated
- Simple
 - Integers, floating point, double precision floating point, characters
 - A Variable is a simple data type
- Complex
 - Words (strings), arrays, images,...
- If you must specify the type of variable, the language is **STRONGLY** Typed
 - (C++, Java)
- If you don't have to specify and can (sometimes) change types in mid code it is **WEAKLY** typed
 - (Python, Matlab)

Java

- Object Oriented
 - Created as OOP from the start
 - This is a different way of THINKING about your code
 - Not something you add in later
- Platform independent
 - Every user downloads a Java Virtual Machine (JVM) for their computer. This JVM deals with hardware specific machine code
 - Your code is compiled to run on the Java virtual machine
- Fast(ish)
 - Nothing beats C/C++ (except assembly)
 - But it's much faster than interpreted languages
- High level
 - Closer to human level languages than C
 - NO POINTERS!!!! (hardest part of C)

- JRE – Java Runtime Environment
 - Needed to run Java expts
- JDK – Java Development kit
 - Needed to program and develop, but....
- Eclipse only needs a JRE, and one is included in the download

variable

- Storage location in computer memory
- Contains information
- `Int j = 5;`
- `Boolean crazyname = true;`

Function

- Procedure to act on information or change state of action
- Function `drawdot()`
- Can receive information or return information
- Function result = `drawdot(x,y)`

Why do we need to pass information?

- Encapsulation
 - Keep data local
 - Other people can use my `drawdot()` without knowing how it works
- Reusability
 - Same function can be used in multiple situations
 - Drawdot on multiple different windows? At many different coordinates?
- Extensibility
 - Hmm, not so much. If we want to improve this function for a more specialized use, we have to write a new function
 - We can call this function from other functions though, and that's nice
 - OOP has this, with INHERITANCE but we'll set that aside for now
- Global variables
 - Information that can be accessed from anywhere
 - In general, this is a bad idea

Class

- Blueprint for new objects
 - And objects do all the real work
- Class Student()
 - Student Tanya = new Student();
 - Student Roopali = new Student();
 - Tanya and Roopali are INSTANCES of the class student
- Contains all of the data (now called STATE variables)
- Contains all of the functionality (now called METHODS)
- Every instance of a class has its own copy of the class's state and methods

- Functions no longer change the global state
- You ask an object to change its state using its methods

State and methods for Student class

- State variables?
- Methods?

- Workbench
- New javaproject
 - Default settings
- Name
 - Look at the libraries. Includes JRE
- Animals-SRC
 - Nothing there
 - Add a class AnimalFarm
 - Check main() (entry point)
- `String noise = new String("Moo");`
- `System.out.println(noise);`
 - Notice the autocomplete

Coding help (IDE)

- Syntax mistakes flagged with red x
 - Try removing ;
- Autocompletion
 - For slow typists or poor memory like me
- Debugger
 - Step, step in, step out, breakpoints
- Try/Catch
- Mouse hover

- Create class dog, cat and cow
- Give each of them a method to speak