CSCI 136:
Data Structures
and
Advanced Programming

Lecture 2

Java Basics

Instructor: Dan Barowy

Williams

Topics

Quiz

Java basics

Static data types

Object orientation

Your to-dos

1. Lab 0, **due Tuesday 2/8**.
2. Read **before Wed**: Bailey, Ch 1-1.4 (inclusive).

Announcements

- Anonymous feedback form on website
- Senior Thesis Proposals this **Friday, Feb 11 @ 2:35pm in Wege Auditorium (TCL 123)**

Quiz

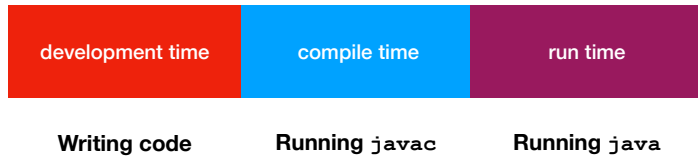https://forms.gle/GKDgJzaUVnj6CLP46

Static data types

Java types

```
byte
short
int
long
float
double
boolean
char
```
and user-defined `class`es

(you should **memorize** this list)

The life of a program

development time	compile time	run time
Writing code	Running `javac`	Running `java`

Definition slide

A **definition slide** defines an important term that you should remember. You are **strongly advised** to write down and study these key terms.

(definition slides **look the same** to make them easy to spot!)
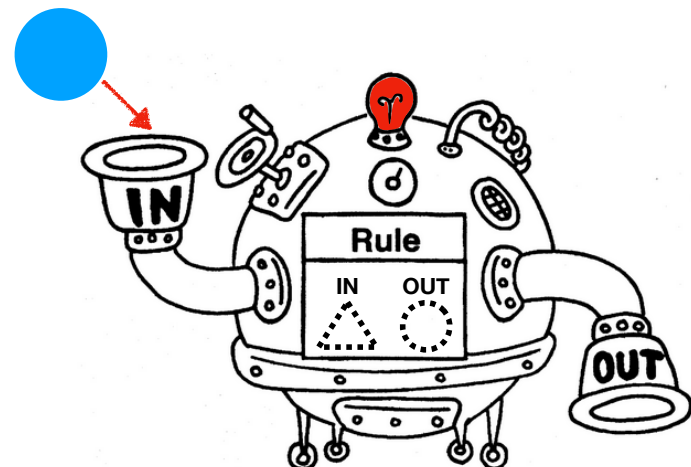
Static data types

A **data type** is a label, assigned to a variable, that ensures that **only valid operations** are allowed using that data.

A **static data type** is a type that is checked **before a program runs**. Static data types are usually checked at **compile time**. A program that fails a static type checks is not compiled.

(**always** be sure to submit lab code that compiles!)

Using data incorrectly yields a type error.

Let's grow a few programs…

Java is Object-Oriented

- OO is a **system** for writing code that has properties **highly valued** by software engineers.
- Those properties are:
  - **Code reuse**
  - **Modularity**
  - **Data abstraction**
- It is sometimes said (**incorrectly**) that OO is about "modeling the real world."
- OO is a very big topic, and it takes awhile to master all the pieces.
- For now, we are going to focus on **data abstraction**.

If you don't understand all these words just yet, **don't worry**.

Recap & Next Class

Today:

- Java basics
- Static data types
- Object orientation

Next class:

- Classes and objects
- Designing programs with classes
- Fancy input using `Scanner`