Introduction to Java Basics

Programming in Java

Why do programming?

- Humans communicate in a **natural language**
  - Large vocabulary (10 000s words)
  - Complex syntax
  - Semantic ambiguity
    - The chair’s leg is broken.
    - The man saw the boy with the telescope.

Why do programming?

- Machines communicate in binary code / machine language
  - Small vocabulary (2 words… 1, 0)
  - Simple syntax
  - No semantic ambiguity

Why do programming?

- Programming language
  - Vocabulary: restricted
  - Syntax: small and restricted
  - Semantic: no ambiguity (almost)

Origins of the Java Language

- Created by Sun Microsystems (1991)
- Originally designed for programming home appliances
- introduced in 1995 and its popularity has grown quickly since
- is an **object oriented programming** (OOP) language

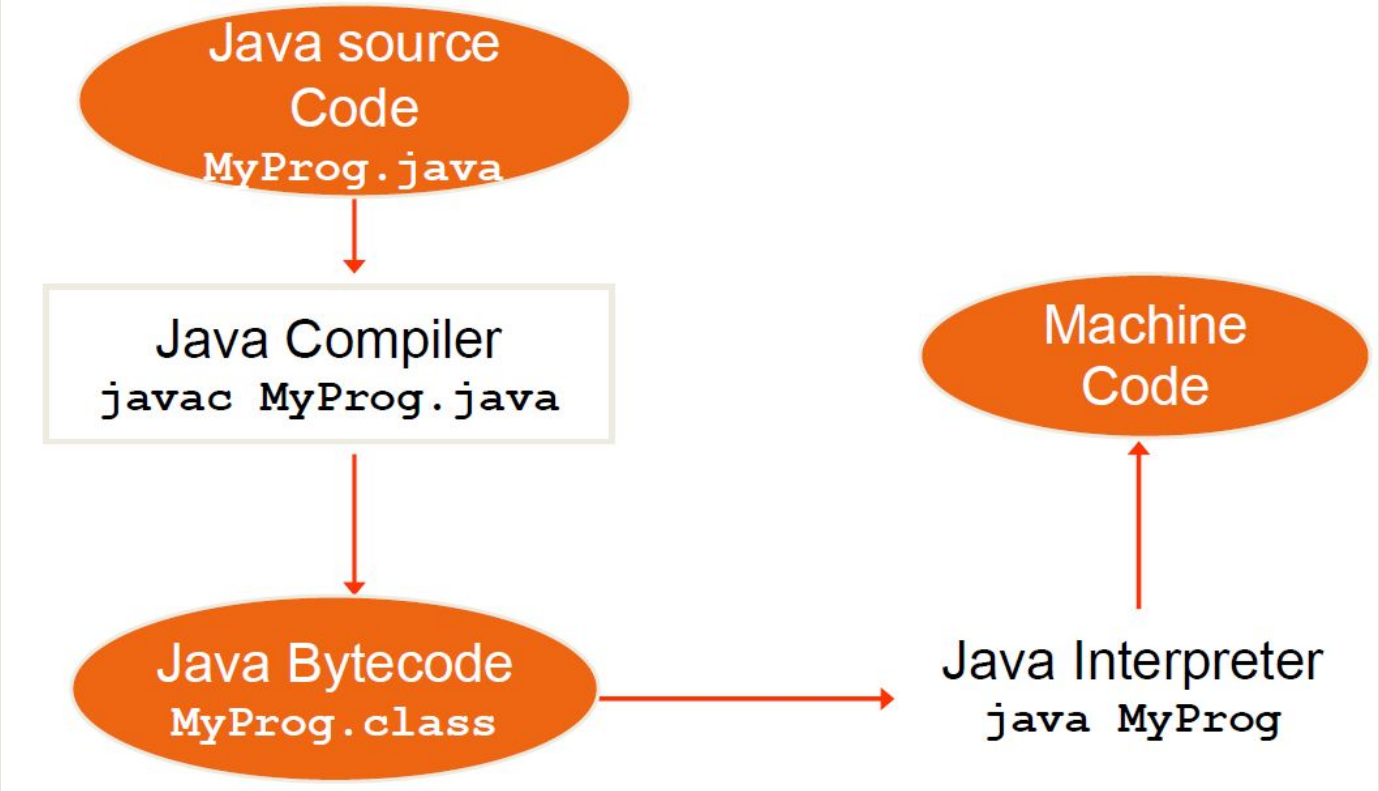
Origins of the Java Language

- A compiler:
  - is a software tool which translates source code into another language
- Usually (ex. C, C++)
  - the compiler translates directly into machine language
  - but each type of CPU uses a different machine language
  - … so same executable file will not work on different platforms
  - … need to re-compile the original source code on different platforms
- Java is different…

Java Translation

- Java **compiler**:
  - Java source code ----> **bytecode**
  - a machine language for a fictitious computer called the *Java Virtual Machine*
- Java **interpreter**:
  - executes the Java Virtual Machine **(JVM)**
  - Java bytecode ----> into machine language and executes it
  - Translating byte code into machine code is relatively easy compared to the initial compilation step
- So the Java compiler is not tied to any particular machine
- Once compiled to *bytecode* , a Java program can be used on any computer, making it very portable

Java Translation



Some definitions

- **Algorithm:**
  - A step by step process for solving a problem
  - Expressed in natural language
- **Pseudocode:**
  - An algorithm expressed in a more formal language
  - Code like, but does not necessarily follow a specific syntax
- **Program:**
  - An algorithm expressed in a programming language
  - Follows a specific syntax

Problem Solving

- The purpose of writing a program is to solve a problem

- The general steps in problem solving are:

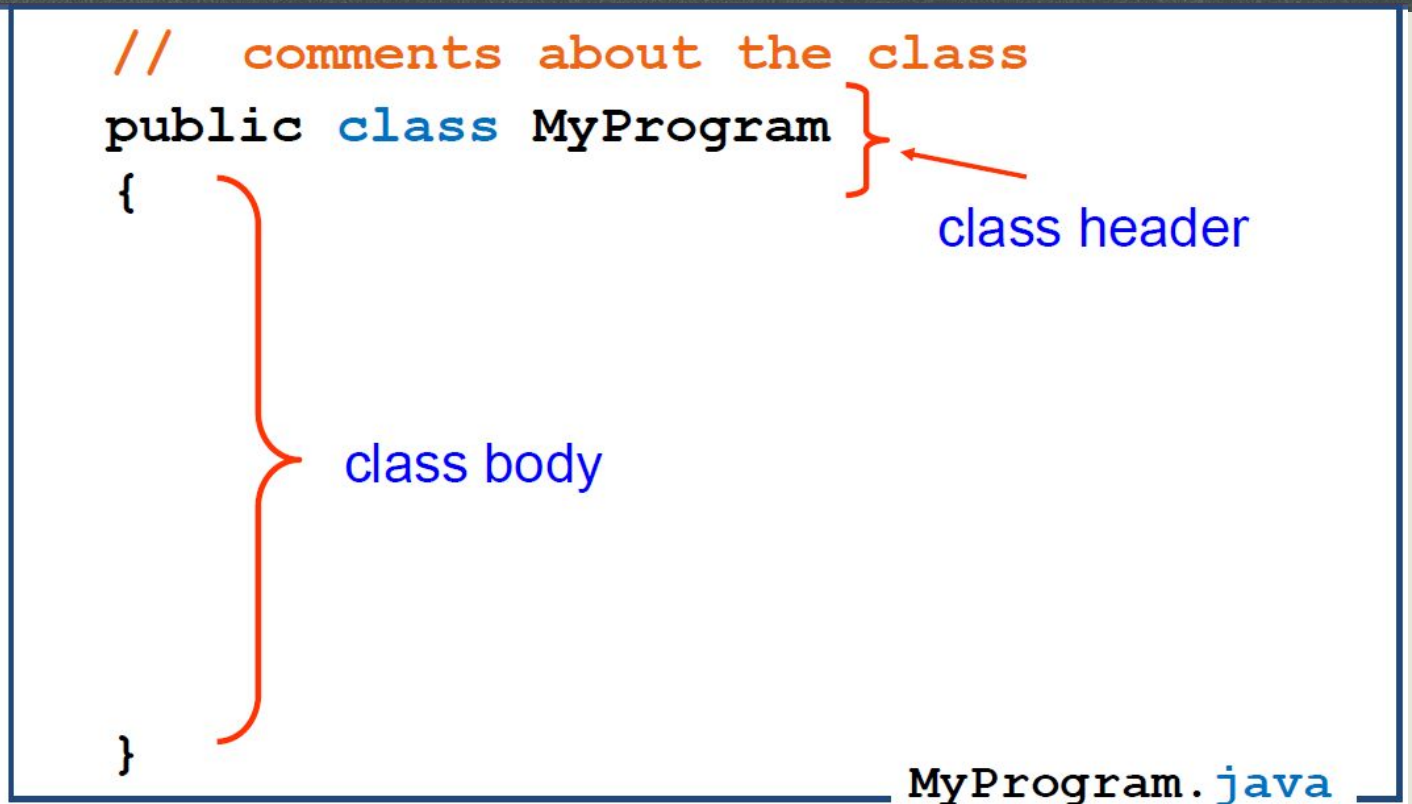
    1. Understand the problem.
    2. Design a solution (find an algorithm).
    3. Implement the solution (write the program).
    4. Test the program and fix any problems.

Java Program Structure

- A java program:
  - is made up of one or more *classes* (collection of actions)
  - a class contains one or more *methods* (action)
  - a method contains program *statements/instructions*

- A Java program always contains a method called main

Java Program Structure



Java Program Structure

```
//  comments about the class
public class MyProgram
{

   //  comments about the method
   public static void main (String[] args)
   {

   }

}
```

method header

method body

MyProgram.java

A small Java Program

```
//*************************************************************************
// Author: N. Houari
//
//  Demonstrates the basic structure of a Java application.
//*************************************************************************

public class Hello
{
   //----------------------------------------------------------------------
   //  Prints a message on the screen
   //----------------------------------------------------------------------
   public static void main (String[] args)
   {
      System.out.println ("Hello World!!!");
   }
}
```

Hello.java

extension of java programs

Java is case sensitive!

Java Program Structure

- Syntax rules
  - define how we can put together symbols, reserved words, and identifiers to make a valid program
- Semantics
  - define what a statement means
- A program that is syntactically correct is not necessarily logically (semantically) correct

3 types of errors

- **Compile-time (syntax) errors**
  - The compiler will find syntax errors and other basic problems
  - An executable version of the program is not created
    - Examples: ?
- **Run-time errors**
  - A problem can occur during program execution
  - Causes the program to terminate abnormally
    - Examples: ?

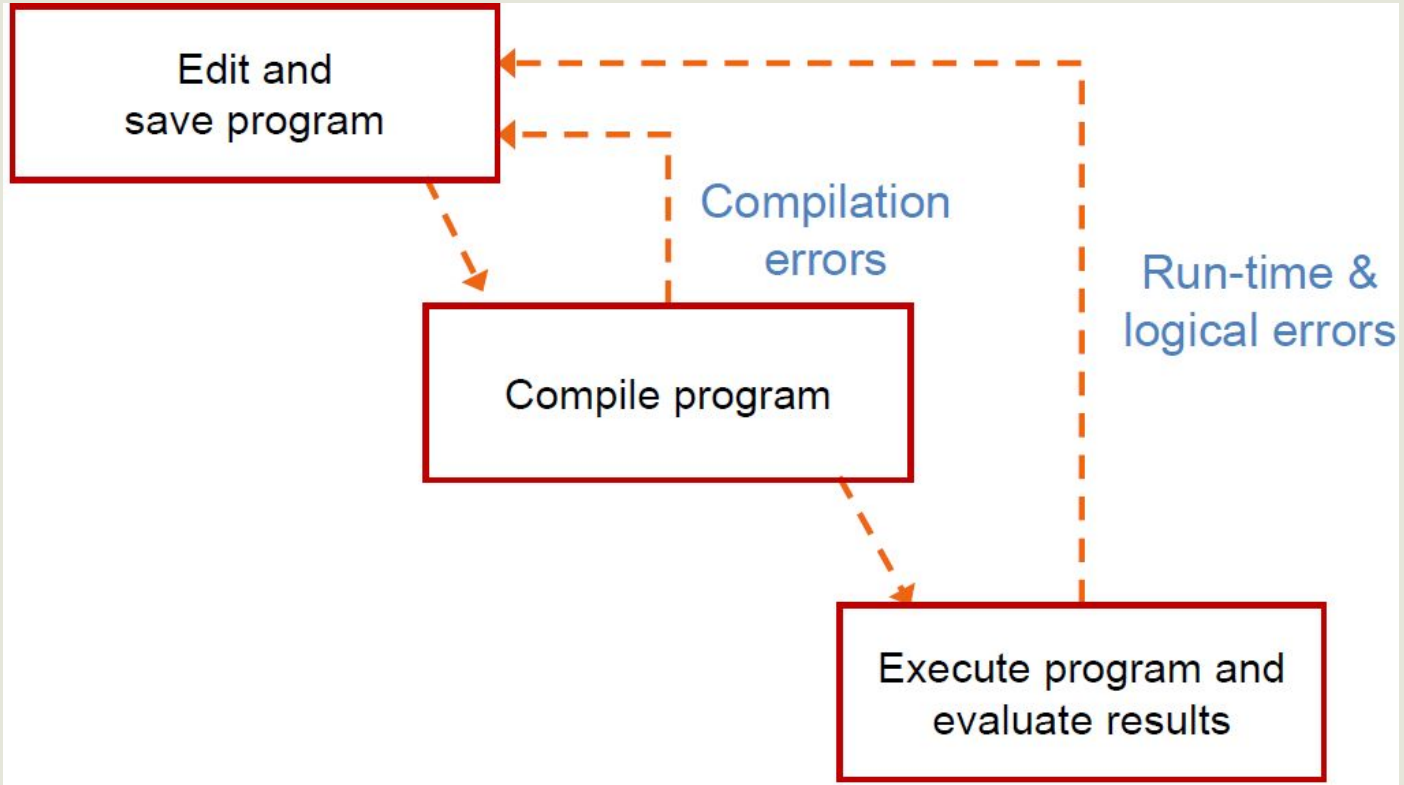
3 types of errors

- **Logical (semantic) errors** - Also known as a ***bug***
  - A mistake in the algorithm
  - Compiler cannot catch them
  - A program may run, but produce incorrect results
    - Ex: ??
  - The process of eliminating bugs is called

Detecting errors

- The hardest kind of error to detect in a computer program is:
  - A) Syntax error
  - B) Run-time error
  - C) Logical error
  - D) All of the above

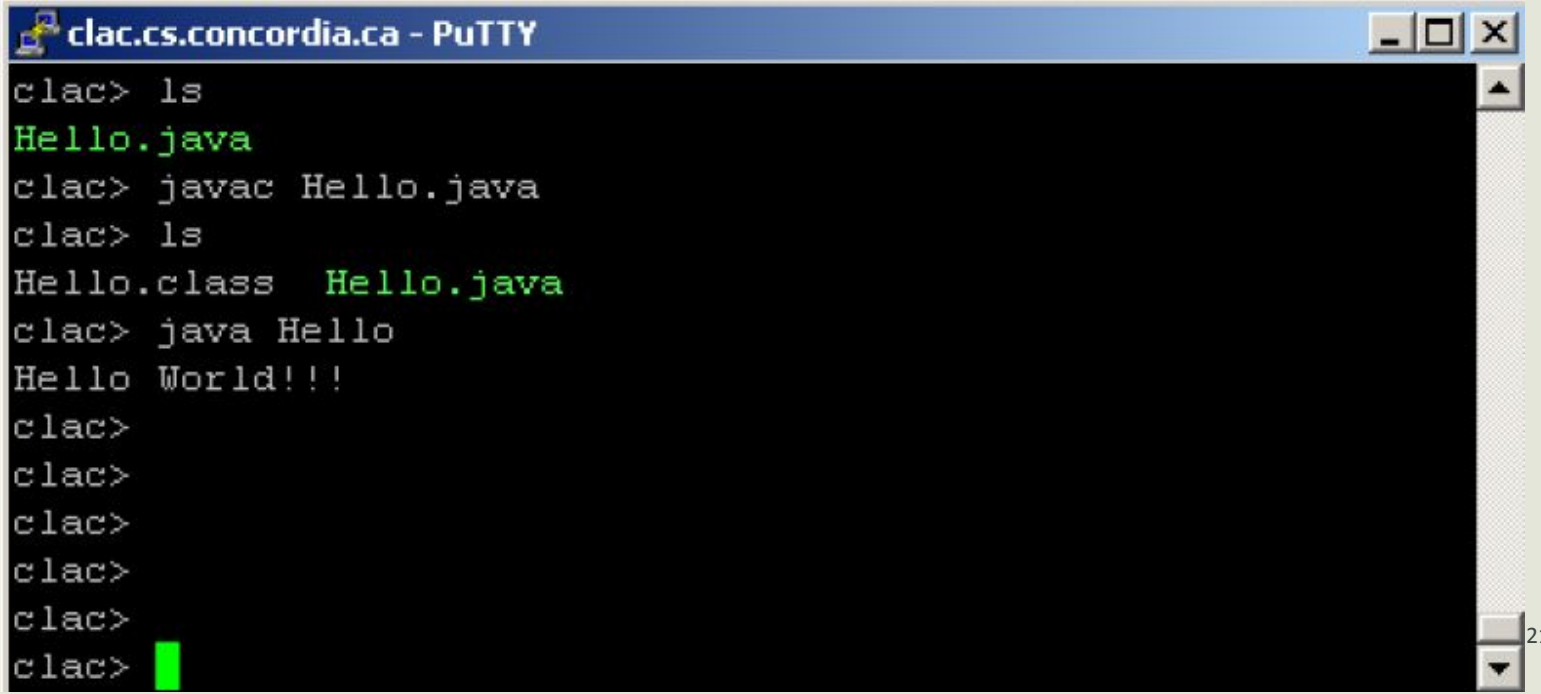
Basic Program Development Flow



Development Environments

- Basic compiler & interpreter
  - Java Development Kit (JDK)
  - Compiler: javac Hello.java
    - The result is a bytecode program called: Hello.class
  - Interpreter: java Hello

Development Environments



Development Environments

- **IDE (Integrated Development Environment)**
  - Eclipse
  - JCreator
  - Borland JBuilder
  - Microsoft Visual J++