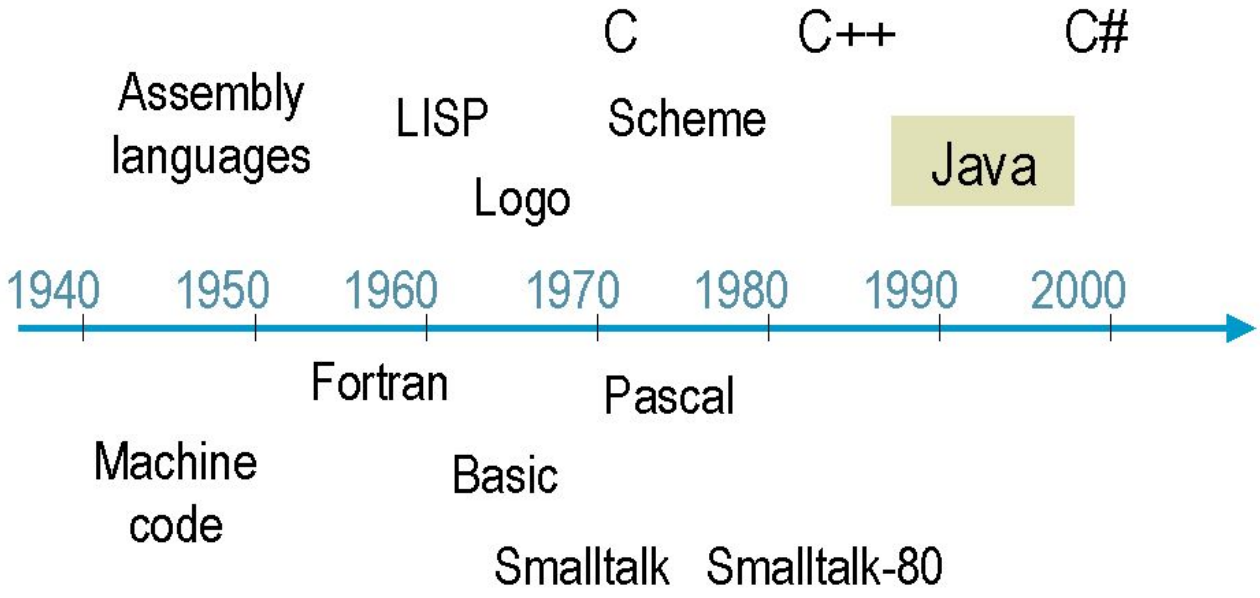
Some Basics

Mr. Neat
Java

Programming Languages

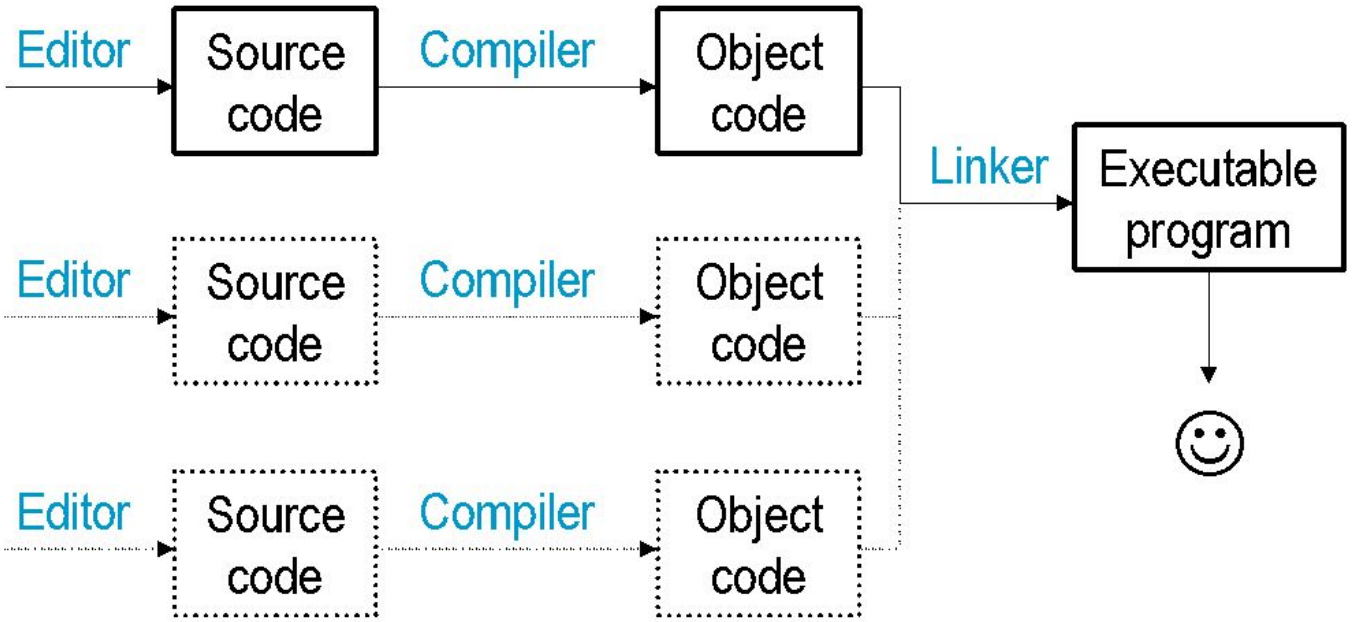


Compiled Languages

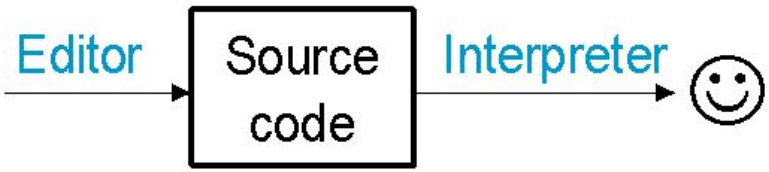
Software Development Tools

- Editor
  - programmer writes *source code*
- Compiler
  - translates the source into *object code* (instructions specific to a particular CPU)
- Linker
  - converts one or several object modules into an executable program
- Debugger
  - stepping through the program "in slow motion," helps find logical mistakes ("bugs")

Compiled Languages: Edit-Compile-Link-Run



Compiler vs. Interpreter



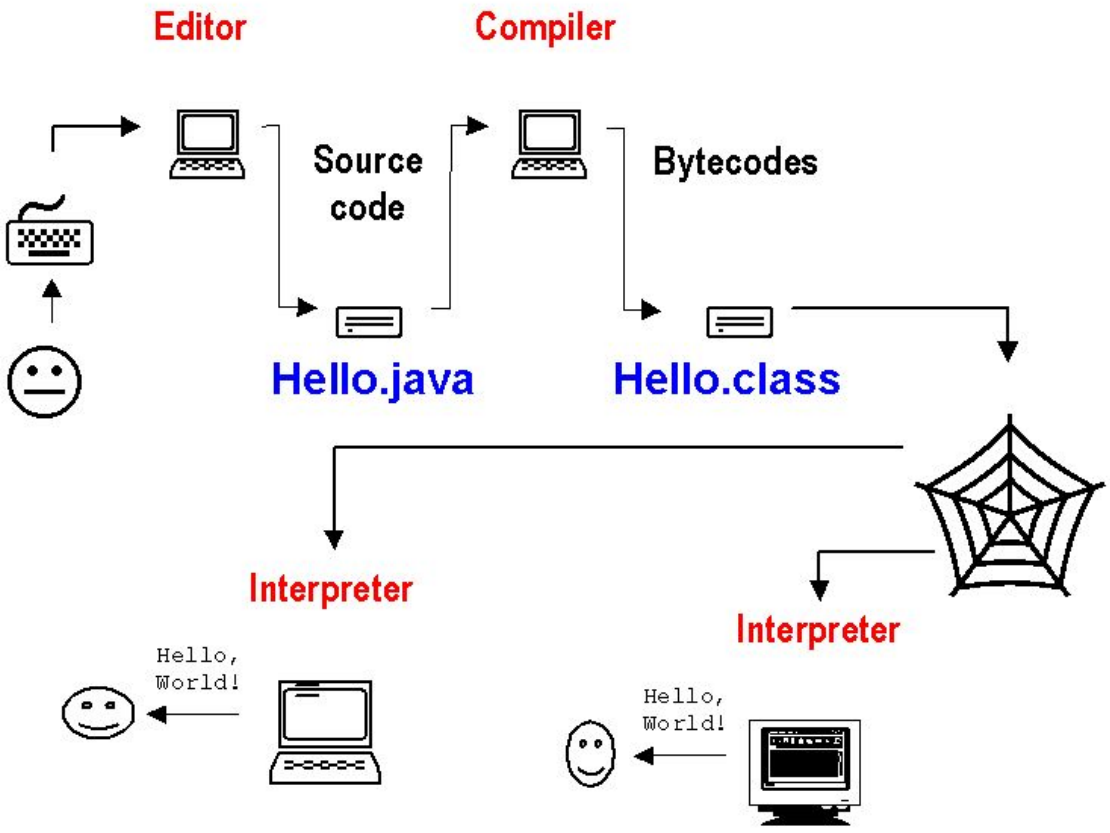
- **Compiler:**
  - checks syntax
  - generates machine-code instructions
  - not needed to run the executable program
  - the executable runs faster

- **Interpreter:**
  - checks syntax
  - executes appropriate instructions while interpreting the program statements
  - must remain installed while the program is interpreted
  - the interpreted program is slower

Java's Hybrid Approach: Compiler + Interpreter

- A Java compiler converts Java source code into instructions for the *Java Virtual Machine.*
- These instructions, called *bytecodes*, are the same for any computer / operating system.
- A Java interpreter executes bytecodes on a particular computer.

Java's Compiler + Interpreter



Why Bytecodes?

- Platform-independent.
- Load from the Internet faster than source code.
- Interpreter is faster and smaller than it would be for Java source.
- Source code is not revealed to end users.
- Interpreter performs additional security checks, screens out malicious code.