1.1 Introduction

• *Java How to Program, Sixth Edition*
  – Java 2 Platform, Standard Edition (J2SE)
  – J2SE Development Kit version 5.0
  – Object-Oriented Programming
1.2 What Is a Computer?

- **Computer**
  - Performs computations and makes logical decisions
  - Millions / billions times faster than human beings

- **Computer programs**
  - Sets of instructions for which computer processes data

- **Hardware**
  - Physical devices of computer system

- **Software**
  - Programs that run on computers
1.3 Computer Organization

• Six logical units of computer system
  – Input unit
    • Mouse, keyboard
  – Output unit
    • Printer, monitor, audio speakers
  – Memory unit
    • Retains input and processed information
  – Arithmetic and logic unit (ALU)
    • Performs calculations
  – Central processing unit (CPU)
    • Supervises operation of other devices
  – Secondary storage unit
    • Hard drives, floppy drives
1.4 Early Operating Systems

• **Batch processing**
  – One job (task) at a time
  – Operating systems developed
    • Programs to make computers more convenient to use
    • Switch jobs easier

• **Multiprogramming**
  – “Simultaneous” jobs
  – Timesharing operating systems
1.5 Personal, Distributed and Client/Server Computing

• Personal computing
  – Computers for personal use

• Distributed computing
  – Computing performed among several computers

• Client/server computing
  – Servers offer common store of programs and data
  – Clients access programs and data from server
1.6 The Internet and the World Wide Web

• Internet
  – Developed more than four decades ago with DOD funding
  – Originally for connecting few main computer systems
  – Now accessible by hundreds of millions of computers

• World Wide Web (WWW)
  – Allows for locating/viewing multimedia-based documents
1.7 Machine Languages, Assembly Languages and High-Level Languages

• Machine language
  – “Natural language” of computer component
  – Machine dependent

• Assembly language
  – English-like abbreviations represent computer operations
  – Translator programs convert to machine language

• High-level language
  – Allows for writing more “English-like” instructions
    • Contains commonly used mathematical operations
  – Compiler converts to machine language

• Interpreter
  – Execute high-level language programs without compilation

© 2005 Pearson Education, Inc. All rights reserved.
1.8 History of C and C++

- **C++**
  - Evolved from C
    - Evolved from BCPL and B
  - Provides object-oriented programming capabilities

- **Objects**
  - Reusable software components that model real-world items
1.9 History of Java

• Java
  – Originally for intelligent consumer-electronic devices
  – Then used for creating Web pages with dynamic content
  – Now also used to:
    • Develop large-scale enterprise applications
    • Enhance WWW server functionality
    • Provide applications for consumer devices (cell phones, etc.)
1.10 Java Class Libraries

• Classes
  – Include methods that perform tasks
    • Return information after task completion
  – Used to build Java programs

• Java provides class libraries
  – Known as Java APIs (Application Programming Interfaces)
1.11 FORTRAN, COBOL, Pascal and Ada

- **FORTRAN**
  - FORmula TRANslator

- **COBOL**
  - COmmon Business Oriented Language

- **Pascal**
  - Structured programming

- **Ada**
  - Multitasking
1.12 BASIC, Visual Basic, Visual C++, C# and .NET

• BASIC
  – Beginner’s All-Purpose Symbolic Instruction Code

• .NET
  – .NET platform

• Visual Basic .NET
  – Based on BASIC

• Visual C++
  – Based on C++

• C#
  – Based on C++ and Java
Fig. 1.1 | Typical Java development environment.
1.15 Test-Driving a Java Application

• Test-driving the ATM application
  – Check system setup
  – Locate the ATM application (Fig. 1.2)
  – Run the ATM application (Fig. 1.3)
  – Enter an account number (Fig. 1.4)
  – Enter a PIN (Fig. 1.5)
  – View the account balance (Fig. 1.6)
  – Withdraw money from the account (Fig. 1.7)
  – Confirm that the account information has been updated (Fig. 1.8)
  – End the transaction (Fig. 1.9)
  – Exit the ATM application

• Additional applications
1.16 Software Engineering

• Objects
  – Reusable software components that model real-world items
  – Look all around you
    • People, animals, plants, cars, etc.
  – Attributes
    • Size, shape, color, weight, etc.
  – Behaviors
    • Babies cry, crawl, sleep, etc.
• **Object-oriented design (OOD)**
  – Models real-world objects
  – Models communication among objects
  – *Encapsulates* attributes and operations (behaviors)
    • Information hiding
    • Communication through well-defined interfaces

• **Object-oriented language**
  – Programming in object-oriented languages is called *object-oriented programming (OOP)*
  – Java
1.16 Software Engineering (cont.)

- **Object-Oriented Analysis and Design (OOA/D)**
  - Essential for large programs
  - Analyze program requirements, then develop solution
  - UML
    - Unified Modeling Language