Chapter 2: Java Fundamentals

Java Program Structure
Evolution of programming languages
Programming methodologies
History of Java
Processing a Java Program
Java Program Structure
  - Welcome, simpleJava, Salam Programs
  - Comments, Identifiers, class definitions.
Saving, Compiling and Running Java Programs
Evolution of Programming Languages

- High-level languages make programming easier.
- Closer to spoken languages.
- Examples:
  - Basic
  - FORTRAN
  - COBOL
  - C/C++
  - Java
Programming Methodologies

Two basic approaches to programming design:

- Structured design

- Object-oriented design
Structured Design

1. A problem is divided into smaller sub-problems.

2. Each sub-problem is analyzed, solved and a solution for this sub-problem is obtained.

3. The solutions of all sub-problems are combined to solve the overall problem.

4. Is called structured programming, top-down design approach, or modular programming.
Object-Oriented Design (OOD)

- In OOD, a program is a collection of interacting objects.
- An object consists of data and operations.
- **Steps in OOD:**
  1. Identify the objects which form the basis of the solution, then determine how these objects interact with each other.

  **Example**: write a program that automates the video rental process for a video store.

  The two main objects are: 1- video  
  2- customer
Object-Oriented Design (OOD)

**Steps in OOD:**

2. Specify the relevant data for each object and the possible operations to be performed on that data.

**Example:** for the video object

- the data might be:
  - movie name, Starring actors, and Number of copies in stock.
- The operations on video object might include:
  - checking the name of the movie, reducing the # of copies in stock by 1 after renting a copy.
Object-Oriented Design (OOD)

- Each object consists of data and operations on those data
- The final program is a collection of interacting objects.
History Of Java

- Developed by Sun Microsystems – a company known for its workstations.
- Java is well known for developing internet applications. It is used to:
  - Create web pages with dynamic and interactive content.
  - Develop large-scale enterprise applications.
  - Enhance the functionality of WWW servers.
  - Provide applications for customer devices (e.g., Cell phones).
- Java programs can run from a web browser.
Processing a Java Program

A Java program undergoes several stages:

1. **Editing**: use java code and save in a text file named `className.java` (source program).

2. **Compiling**: the compiler checks the source program for any syntax errors then translates the program into code understood by interpreter called bytecode saved in a file named `className.class`.

3. **Loading**: the `.class` file is loaded into computer main memory for execution, and connected to all classes.

4. **Verifying**: to validate and secure against damage.

5. **Interpreting**: the Interpreter reads and translates each bytecode instruction into machine language and then executes it, one instruction at a time.
Processing a Java Program

Figure 1-3  Processing a Java program
Processing a Java Program

Two types of Java programs:

- **applications**: standalone programs stored and executed on a local computer.
- **applets**: small programs stored on remote computers that users connect to via a WWW browser. Applets are loaded into the browser, executed, then discarded.
Java Program Structure

- The basic unit of a Java program is a **class**.
- Every class consists of one or more methods.
- A method is a set of statements that accomplish something.
- A Java class **must contain one main method** if it is an application.
- Execution always begins with method main in Java application program.
A Java Program 1- Welcome

- A simple java application: an application executes using the java interpreter.

Example:

```java
// This prints a line of text
public class Welcome
{
    public static void main (String args[] )
    {
        System.out.println("welcome to java");
    }
}
```
A Java Program 1 - Welcome

- // single line comment
- /* */ multiple line comment
- Every java program must have at least one class.
- Each class begins with a class declaration that defines data and methods for the class.
- The class name here is Welcome, and contains a method main().
- Welcome is an identifier.
A Java Program - Comments

- *Comments* are used to describe what your code does and aid reading your code.
- The Java compiler ignores them.
- Comments are made using
  - `//`, which comments to the end of the line,
  - or `/* */`, everything inside of it is considered a comment (including multiple lines). The comment begins after the first `/*`. It ends just before the first `*/`.

- Examples:
  ```java
  /* This comment begins at this line.
  This line is included in this comment
  It ends at this line. */
  
  // This comment starts here and ends at the end of this line.
  ```
Java Program - Java Identifiers

- Names of things. Such as methods, class names...
- Consists of:
  - Letters
  - Digits
  - The underscore character (_)
  - The dollar sign ($)
- Must begin with a letter, underscore, or the dollar sign. (i.e. does not begin with a digit)
- Java is case sensitive. A and a are different.
Java Program- class definition

- Always begin a class name with a capital letter.

- The class definition should be saved in a file that contains the class name. (i.e. Welcome.java)

- A file cannot contain two public classes.

- public static void main (String args[]) is a part of every java application program.
Java Program- class definition

- Java applications automatically begin executing at `main()`. 
- The parentheses () after `main` indicate that `main` is a method.
- Class definitions normally contain one or more methods.
- One of those methods must be called `main`.
- The `void` before `main()` means that `main` will not return any info.
public class SimpleJavaProgram {

    public static void main(String[] args) {
        System.out.println("My first Java program.");
        System.out.println("The sum of 2 and 3 = " + 5);
        System.out.println("7 + 8 = " + (7 + 8));
    }
}
A Java Program 2 - SimpleJavaProgram

- A java output statement causes the program to evaluate whatever is in the parentheses and display the result on screen.
- Anything in double quotation marks, called string, evaluates to itself.
- + is used to concatenate the strings. The system automatically converts the number 5 into a string, joins that string with the first string, and displays it.
A Java Program 2-
SimpleJavaProgram

- The parentheses around 7+8 causes the system to add the numbers 7 and 8, resulting in 15.
- The number 15 is then converted to string 15 and joined with string “7+8” = “".

Sample Run:

My first Java program.
The sum of 2 and 3 = 5
7 + 8 = 15
Defining the Problem

- The problem must be defined in terms of:
  - **Input:** Data to be processed.
  - **Output:** The expected result.
  - and **processing:** The statements to achieve.

- Look for nouns in the problem statement that suggest output and input.
- Look for verbs to suggest processing steps.
// import Section – import used java libraries
public class MyProgramName {

// main method
    public static void main( String args[] ){
        // Declaration section – Declare needed variables

        // Input section – Enter required data

        // Processing section – Processing Statements

        // Output section – Display expected results

    } // end main

} // end class
A java Program 3 – Salam Program

```
// import section: Empty
public class Salam {

    // main method
    public static void main(String args[]) {
        // Declaration section: Empty

        // Input section: Empty

        // Processing section: Empty

        // Output section
        System.out.println("... Assalamo Alaikom ...");
    } // end main

} // end class
```
Saving, Compiling and Running Java Programs

• Saving a Java program.
  – A file having a name same as the class name should be used to save the program. The extension of this file is ".java".
    – Salam program should be saved in a file called “Salam.java”.

• Compiling a Java program.
  – Call the Java compiler
    – The Java compiler generates a file called "Salamclass" (the bytecode).

• Running a Java program
  – Call the Java Virtual Machine
    – More in the Lab .....