Yunu Jung (정윤우)
15655197127

AP Computer Science A

Lesson 1

Overview of the Java Language

- Java is a **platform-independent programming language** that runs on the **JVM** and is widely used for web, mobile, and enterprise applications.
- The core goal of AP CSA: master the fundamentals of Java programming to solve **real-world problems**.

Program Execution Principles

- **Java program flow**: Write source code (.java) → Compile to bytecode (.class) → JVM executes.
- **JVM (Java Virtual Machine)**: Allows Java programs to run on different devices.
- **JRE (Java Runtime Environment)**: Contains the JVM and standard libraries, providing runtime support.

Code Organization: Package and Class

- **Package**: A "folder" for organizing code to prevent naming conflicts (e.g., java.util).
- **Class**: The basic unit of code that contains the program's logic.
- **main method**: The entry point of a program, with the format public static void main(String[] args).

Data Types

- **Primitive Types**: int (integers), double (floating-point numbers), boolean (true/false), char (characters), etc.
- **Reference Types**: Such as String, used to store complex data.
- **Variables**: Declaring (specifying a type), initializing (assigning a value), and scope (the valid range of a variable).

Basic Math Operations

- **Operators**: +, -, *, /, % (modulus).
- **Type Casting**: Converting one data type to another, e.g., int to double (implicit) or double to int (explicit).
- **Example**: int x = 5; double y = x / 2.0; // y = 2.5.

Control Structures

- **Conditional Statements**: if (condition) { code } else { code }.
- **Loops**: for (fixed number of iterations), while (conditional loop), do-while (executes at least once).
- **Example**: for (int i = 0; i < 5; i++) { System.out.println(i); }.

Input & Output

- **Output**: System.out.println("Hello, World!");.
- **Input**: Using the Scanner class, for example, Scanner scanner = new Scanner(System.in); int num = scanner.nextInt();.
- Example program: Input two numbers and output their sum.

Compiler and JRE

- **Compiler** (javac): Compiles .java files into .class bytecode files.
- **JRE**: Runs bytecode and provides Java's standard library support.
- Run commands: javac MyProgram.java ➜ java MyProgram.

Error Handling

- **Syntax Errors**: Incorrect code format (e.g., missing a semicolon).
- **Runtime Errors**: Errors that occur while the program is running (e.g., division by zero).
- **Logic Errors**: The program runs but produces an unexpected result.
- **Debugging** Tip: Check error messages and verify code logic step-by-step.