Chapter 1

**1.1**

Computers execute very basic instructions in rapid succession

A computer program is a sequence of instructions and decisions

Java was designed to be *safe* and *portable*, benefiting both Internet users and students.

Java has a very large library. Focus on learning those parts of the library that you need for your programming projects.

An editor is a program for entering and modifying text, such as a Java program

Classes are the fundamental building blocks of Java programs.

Every Java application contains a class with a main method. When the application starts, the instructions in the main method are executed

Each class contains declarations of methods. each method contains a sequence of instructions

A method is called by specifying the method and its arguments

A string is a sequence of characters enclosed in quotation marks.

A compile-time error is a violation of the programming language rules that is detected by the compiler

A run-time error causes a program to take an action that the programmer did not intend

A **computer program** tells a computer, in minute detail, the sequence of steps that are needed to fulfill a task.

The physical computer and peripheral devices are collectively called the **hardware**.

The programs the computer executes are called the **software**

The act of designing and implementing computer programs is called **programming**

**1.2**

**central processing unit (CPU)** structural elements, called *transistors*

CPU performs program control and data processing. That is, the CPU locates and executes the program instructions; it carries out arithmetic operations such as addition, subtraction, multiplication, and division; it fetches data from external memory or devices and places processed data into storage

Primary storage is Ram, short term

Secondary storage is HDD, or other storing media, long term.

**1.3**

**Java** started as **Applets** in web browser, it is rich in **Library**

Java compiler translate instructions for the Java **virtual machine (JVM)** thus Java is **platform-independet**.

**1.4**

On many computers there is an **integrated development environment****(IDE)** in which you can write and test your programs

There can be **editor** or *console window* for typing commands

Java is **case sensitive**

In order to run your program, the Java compiler translates your **source code** (that is, the statements that you wrote) into *class files*. (A class file contains instructions for the Java virtual machine.)

You store your programs in **files**. Files are stored in **folders** or **directories**

**1.5**

Every Java program consists of one or more classes and classes need to be declared, *public, private,* etc.

In Java, every source file can contain ***at most*** one public class, and the name of the public class must match the name of the file containing the class

Every Java application **must have** a *main* method

The main method contains one or more instructions called **statements**

Whenever you call a method in Java, you need to specify

1. The method you want to use (in this case, System.out.println).
2. Any values the method needs to carry out its task (in this case, “Hello, World!”). The technical term for such a value is an **argument**. Arguments are enclosed in parentheses. Multiple arguments are separated by commas

Whenever you call a method in Java, you need to specify

1. The method you want to use (in this case, System.out.println).
2. Any values the method needs to carry out its task (in this case, “Hello, World!”). The technical term for such a value is an **argument**. Arguments are enclosed in parentheses. Multiple arguments are separated by commas

A sequence of characters enclosed in quotation marks is called a **string**

The *System.out.println* method prints a string or a number and **then starts a new line**

The *System.out.print* method prints an item **without starting a new line**

**1.6**

a **compile-time error**. Something is wrong according to the rules of the language and the compiler finds it, compile-time errors are often called *syntax errors*

a **run-time error** The program is syntactically correct and does something, but it doesn't do what it is supposed to do. Because run-time errors are caused by logical flaws in the program, they are often called *logic errors*

Some kinds of run-time errors are so severe that they generate an **exception**: an error message from the Java virtual machine.

**1.7**