

■ 今日の講義で学ぶ内容 ■

- J a v a 言語とC言語の違い
- プログラム開発環境 CPad の使用方法
- プログラムの実行の仕方

1. J a v a とは？

1995 年に Sun Microsystems 社（現在、Oracle 社）*1 から発表されたオブジェクト指向言語（Object-oriented language）です。オブジェクト指向とは、ソフトウェアが扱う対象の振る舞いを物理的または抽象的な実体とそれらの相互作用として捉える考え方です。オブジェクト指向は、近年複雑・大規模化するソフトウェアシステムを見通し良く記述できる特徴をもち、様々なプログラミング言語に取り入れられています。

J a v a 言語 → オブジェクト（物体）に着目したプログラミングスタイル
たとえば、

PCは、CPUとメモリ、マウス、スクリーン・・・をもつ

と“何を”もつかを考えるスタイル



C 言語 → 手続きに着目したプログラミングスタイル
たとえば、

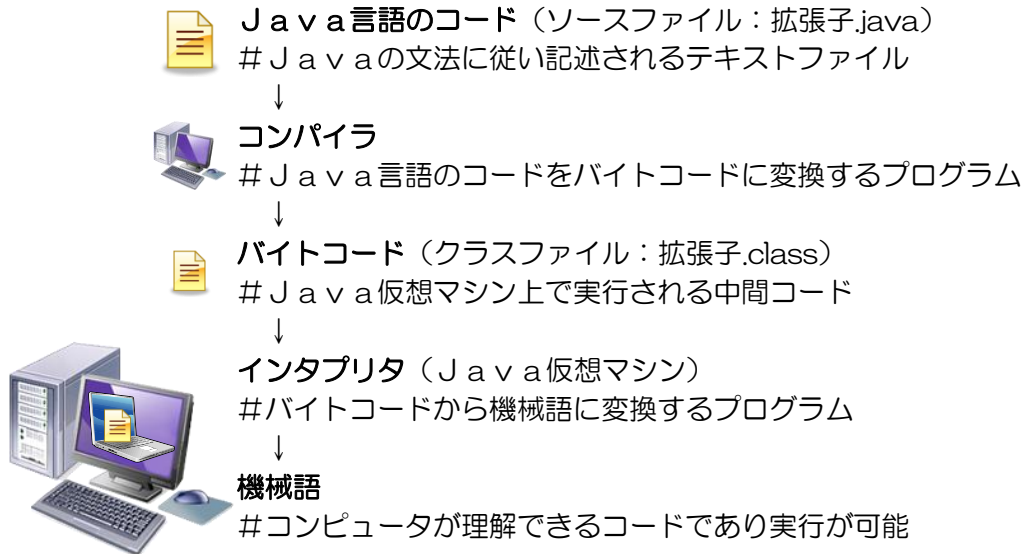
最初に〇〇して、次に〇〇して・・・、最後に〇〇をする

と“どのように”するかを考えるスタイル



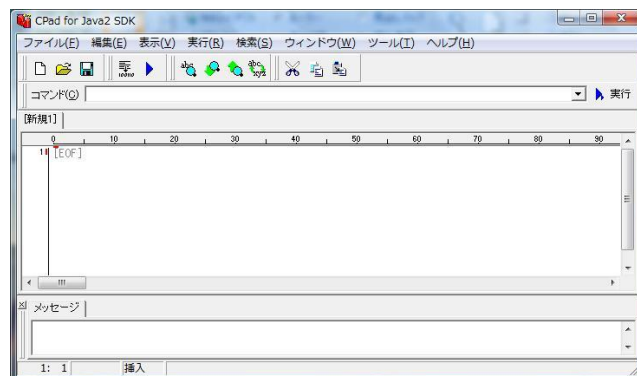
*1 2010年1月27日（米国時間）Oracle 社が Sun Microsystems 社を買収

2. 実行までの流れ



3. Java言語開発環境

コンパイラやインタプリタは Oracle 社から JDK (Java 2 Standard Edition Development Kit) として無償で提供されています。本講義ではコンパイル・実行を GUI (Graphical User Interface) から簡単に起動できるフリーソフトである CPad for Java2 SDK を使用します。



4. 次のコードを打ち込んで実行してみよう

ソースファイル名: Sample1.java

```
class Sample1
{
    public static void main(String[] str)
    {
        System.out.println("ようこそ Java へ!!");
    }
}
```

The diagram shows the Java code for Sample1.java with several annotations explaining syntax rules:

- [Tab]タブ**: Points to the indentation of the code blocks.
- ファイル名 (拡張子より前) と一致させます**: Points to the class name 'Sample1', indicating it must match the filename before the extension.
- 英数字は半角で入力します**: Points to the code, indicating that alphanumeric characters should be entered in half-width (ASCII).
- 大文字と小文字を区別します**: Points to the code, indicating that uppercase and lowercase letters are treated as distinct characters.

※入力上の注意点

- 英数字は全角ではなく、半角で入力します
- 英字の大文字と小文字は異なる文字として処理されます