Javaプログラミング（1）

成蹊大学理工学部
情報科学科

Java言語とは

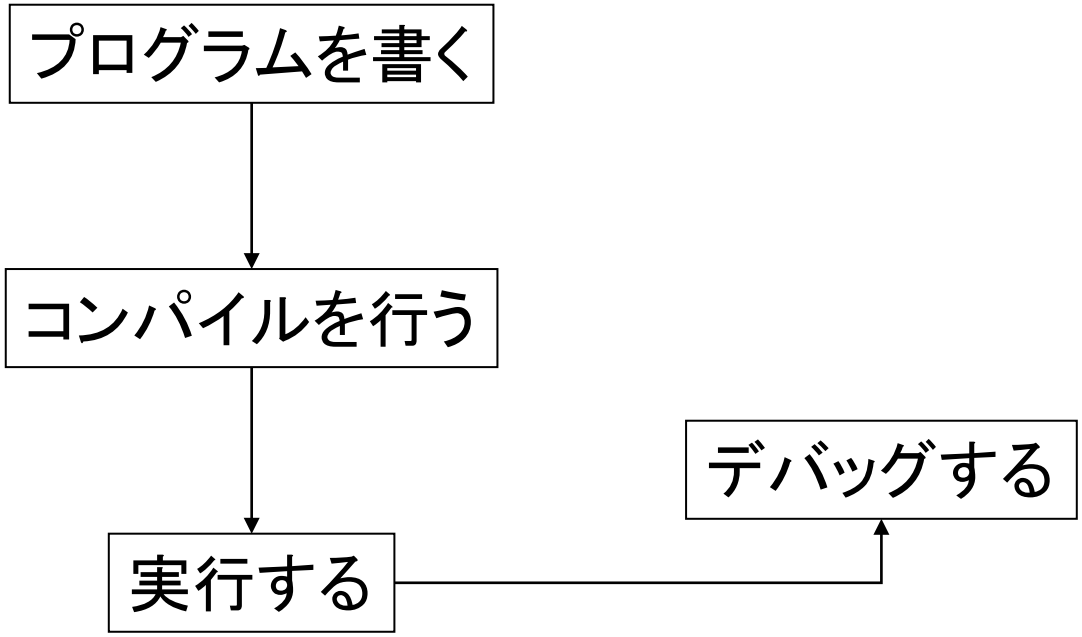
- 1990年代前半にSun Microsystems社が開発
- 現在はOracle社が開発
- オブジェクト指向言語
  - ソフトウェアの開発効率，保守性が高い
- プラットフォームに依存性しない
  - Windows，LinuxなどOSが異なっていても実行可能
  - CやC++では，コンピュータの機種が異なる場合，「移植」が必要
- 企業の大規模情報システムから携帯端末用ソフトまで幅広く用いられている
- コア部分に加え，様々な拡張（データベース，マルチメディア，XML）が用意されている

Javaの環境(Java2プラットフォーム)

- (a) J2SE (Java 2 Platform, Standard Edition)：一般パソコン向け
  - (1) J2SE Development Kit (SDK)：Javaの開発および実行環境. JDKとも呼ばれる
  - (2) Java Runtime Environment (JRE)：Javaの実行環境
  - JDK=JRE+コンパイラ
- (b) J2EE (Java 2 Technology, Enterprise Edition)：サーバ開発向け
- (c) J2ME (Java 2 Platform, Micro Edition)：携帯端末向け

- Javaの入手
  - 別資料を参照のこと

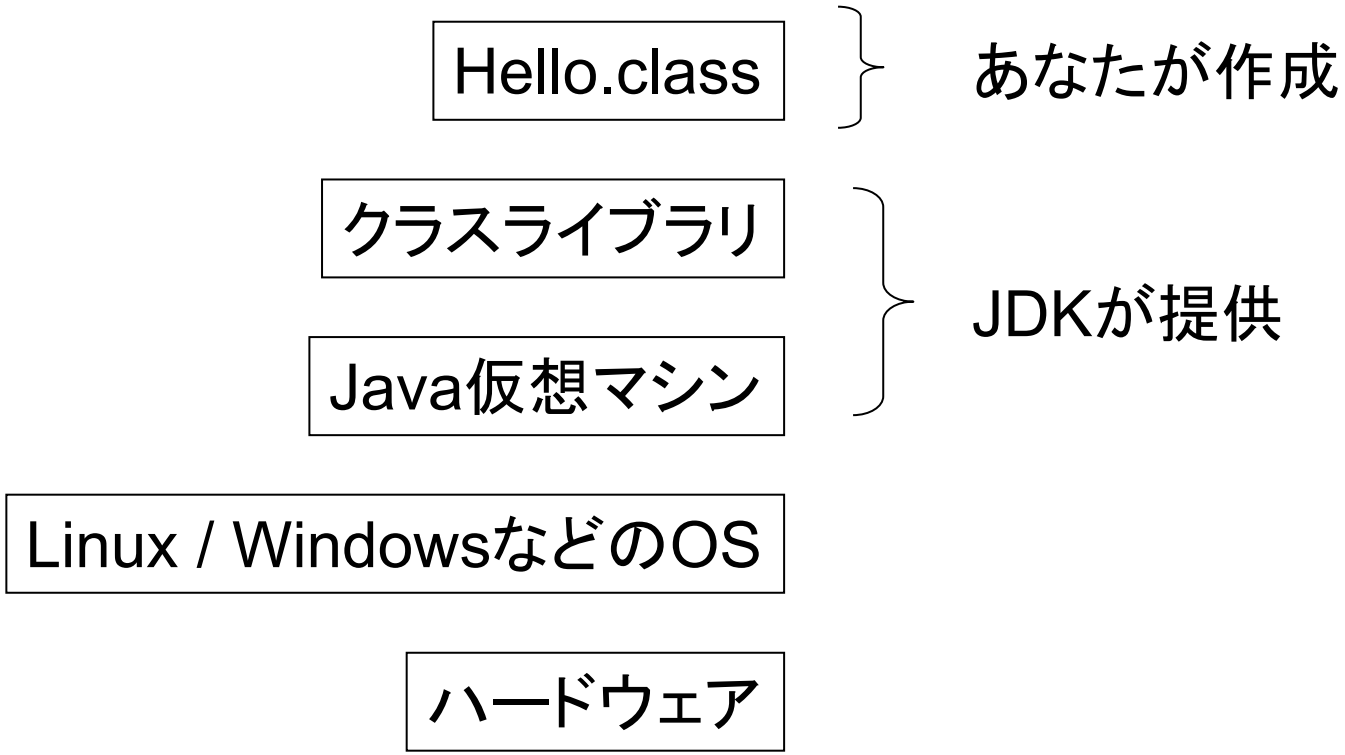
Javaプログラムの開発の流れ (1)



Javaプログラムの開発の流れ (2)

- プログラムを書く
  - ファイルの拡張子は“.java ”にすること
    (ex.) Hello.java
- コンパイルする
  - 拡張子 “.class ”というファイルができる (ex.)
    Hello.class
- 実行する
  - クラスファイルをプログラムとして実行
- （うまく行かなかったら）デバッグする

Java実行環境


Hello.class
あなたが作成
クラスライブラリ
JDKが提供
Java仮想マシン
Linux / WindowsなどのOS
ハードウェア

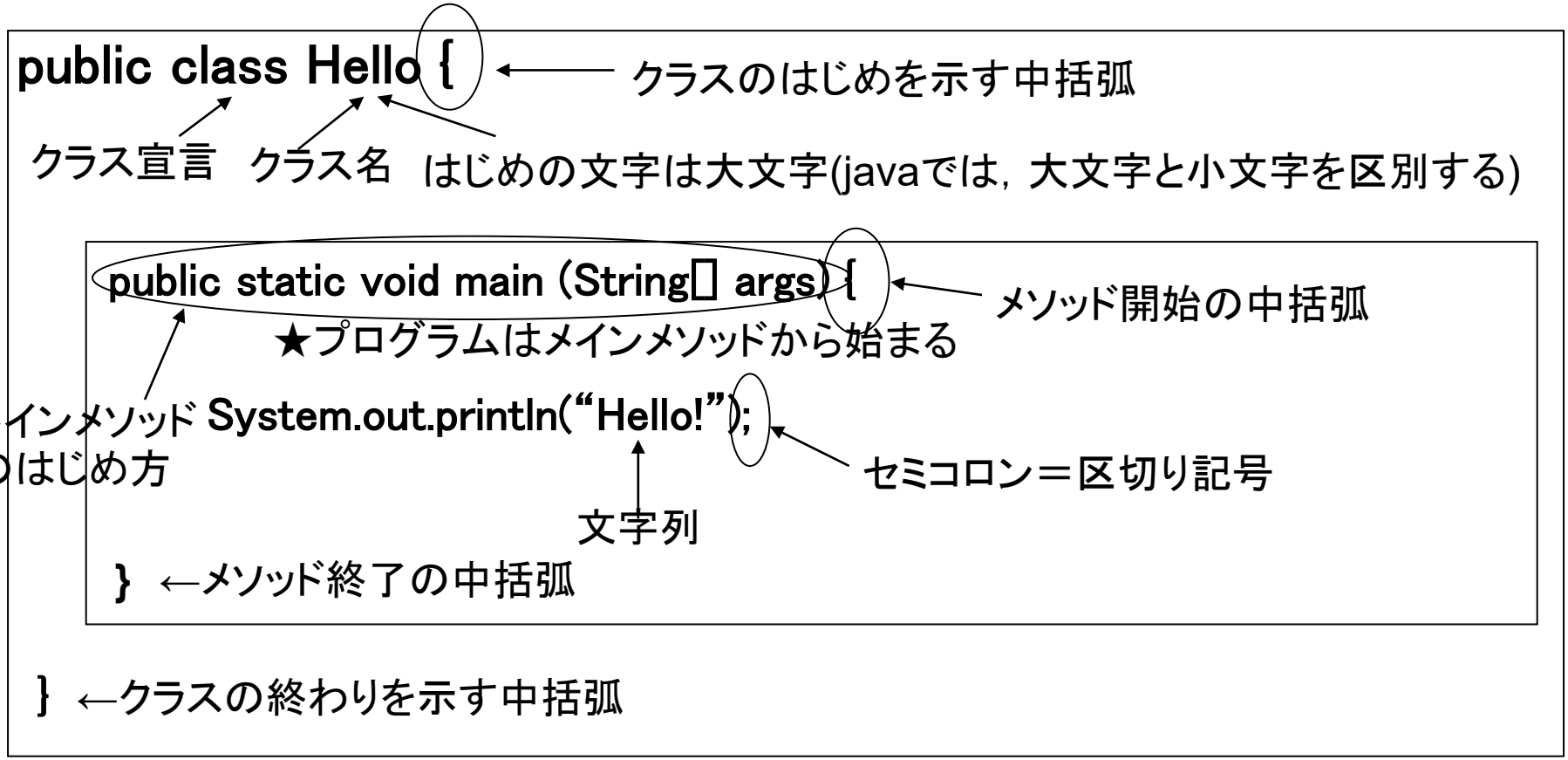
Java実行環境のポイント

- Java仮想マシン(Java Virtual Machine):.classのクラスファイル（バイトコード）を実行するための仮想的な機械
- クラスライブラリ:Javaプログラムから呼び出して使えるライブラリ群
- クラスファイル:.javaをコンパイルすると生成されるファイル

★ポイント

- クラスファイルはJava仮想マシンが実行する
- Java仮想マシンのおかげでハードウェアやOSに依存せずにクラスファイルを実行できる

Javaのプログラムを作ってみる



基本事項

- class: クラス宣言
- クラス名の最初は大文字
- ファイル名とクラス名が一致していること
  例では，Hello.javaというファイル名前で保存
- クラス，メソッドの開始，終了は{ }
- 各文の終りにはセミコロン”;”をつける
- コメントアウトの仕方
  - 行単位のコメントアウト　// これはコメントです
  - 任意の領域のコメントアウト　/*　これはコメントです　*/
- メインメソッドのはじめ方
  public static void main (String[] args) {
- プリント文
  System.out.println(“Hello”);

コンパイルと実行

- コンパイル
  - Windowsコマンドプロンプトで
    > javac Hello.java
  - エラーが表示されていたら修正
  - クラスファイル(Hello.class)が生成されていることを確認
- 実行
  > java Hello
  Hello!
  - > java Hello.classとしないこと！
  - > java Hello.javaとしないこと！

開発環境 Eclipse（エクリプス）

- 統合開発環境
- もともとIBMが開発環境として作り，フリーで公開
- ソースコードの編集，プログラム実行，デバッグを1つの環境で行える
- Eclipseの入手
  - 別資料を参照のこと

加減乗除の演算

- 足し算 : +　3+5;
- 引き算 : -　4-2;
- 掛け算: *　2*3;
- 割り算 : /　6/2;
- 演算の優先順序
  - +-よりも*/が優先される
  - 同じ優先順序の場合は，左から順に計算される

演算子を使ったプログラム

```
public static void main (String[] args) {
    System.out.println(”2たす3は”+(2+3)+”です．”;
}
```

プリント文の引数を分解すると
”2たす3は”
+　←文字列の連結を行う＋記号
(2+3)　←加算を行う演算子
+
”です．”

実行結果
2たす3は5です.

変数宣言

int x;
整数の変数
int型：32ビットで表現できる符号付の整数
-2147483648～2147483647

char c;
文字変数
char型：16-bit 符号無しUnicode

変数の代入

x=4;
変数xに4を代入
x=1+2*3;
変数xには計算結果の7が代入される

c='A';
変数cにAを代入

複合演算子
x+=10← x=x+10

インクリメント／デクリメント演算子
x++／x--：xを評価してから式を演算し，x+1／x-1に更新
++x／--x：x+1／x-1に更新してから式の評価，演算

変数の初期化，参照

- 変数の初期化
  - 変数宣言と同時に値を代入する

  int x = 4;

  char c = ‘A’;

  変数は初期化しておくほうが望ましい
- 変数の参照
  - 変数xの中身をプリント文で出力
    System.out.println(x);
  - 変数cの中身をプリント文で出力
    System.out.println(“変数cの値は”+c);
    (実行結果）変数cの値はA

本日のまとめ

- Java言語とは?
- Javaの実行環境を理解する
- Javaのプログラムを作ってみる
  - Javaのプログラムのコンパイルと実行
  - 演算子(加減剰余)