

## ■Java プログラムの形式

Java プログラムは「クラス」の集合体

```
import 文

class クラス名 {
    public static void main(String[] arg) {
        (ここが main メソッド)
        java の文 ;
        java の文 ;
        :
    }
}
```

- ①import文 → 他のクラスの機能を利用したいときに指定する
- ②クラス → プログラムを構成する単位 (クラス名=プログラム名)
- ③mainメソッド → プログラム本体を記述する部分

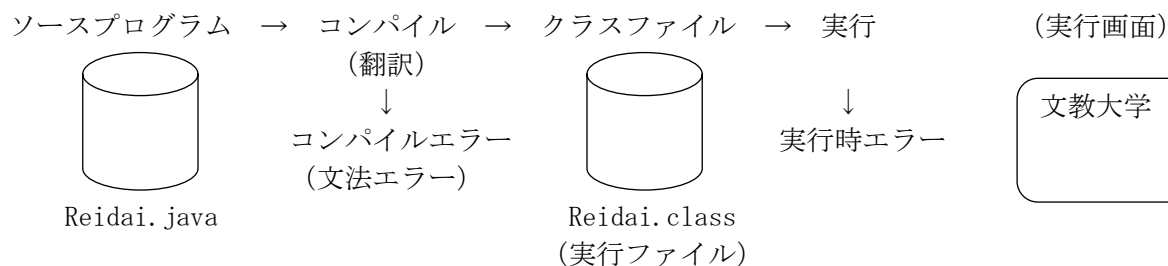
## ■プログラムの作成から実行まで

- (1)プログラムの入力      フリーソフト「CPad for Java」      (テキストエディタでも可能)

```
class Reidai {
    public static void main(String[] args){
        System.out.println("文教大学");
    }
}
```

← ソースプログラム  
(大文字と小文字に注意!)

- (2)プログラム実行の流れ



## ■画面に文字を表示する文(print)

System.out.println("メッセージ"); → 表示後に改行あり  
System.out.print("メッセージ"); → 改行なし

(任意の位置で改行する場合)

System.out.print("¥n");      (¥n は改行の意味)

## ■定数と変数

整数 int 10進数 -2,147,483,648 ~ +2,147,483,647  
           8進数 数字の先頭が0で始まる (010 077 など)  
           16進数 数字の先頭が0xで始まる (0x10 0xFFFF など)  
 実数 float 単精度  $-3.4 \times 10^{38} \sim +3.4 \times 10^{38}$   
       double 倍精度  $-1.7 \times 10^{308} \sim +1.7 \times 10^{308}$

文字 char シングルコーテーションで囲んだ単一文字 ('a' '5' '文' など)  
 文字列 String ダブルコーテーションで囲んだ文字列 ("bunkyo" "文教大学" など)

## ■変数の利用

型名 変数名 ;  
 (定義) int kazu;  
       double jissu;  
       char moji;  
       String mojiretu;

(代入) kazu=100;  
       mojiretu="bunkyo";

(注) 右辺の値が左辺の変数に代入される

(変数の表示例)

変数のみを表示 System.out.println(kazu);  
 メッセージ付きで表示 System.out.println("数値="+kazu);  
 8進数で表示 System.out.println(Integer.toOctalString(kazu));  
 16進数で表示 System.out.println(Integer.toHexString(kazu));

+記号で文字列を連結できる

## ■キーボードからの入力

(画面例)

文字を入力してください-->bunkyo

(注) キーボードから入力する文字は、すべて「文字列」である

(プログラム例)

```
public static void main(String[] args) throws Exception {
    BufferedReader inp=new BufferedReader(new InputStreamReader(System.in));
    String keybd;

    System.out.print("文字を入力してください-->");
    keybd=inp.readLine();
    _____①_____
}
```

(波線は変数名なので名称は任意)

文字列以外の値を処理したい場合は①の部分に次の文を入れる

(整数を入力する場合 int kazu )  
 kazu=Integer.parseInt(keybd);

(実数を入力する場合 double jissu )  
 jissu=Double.parseDouble(keybd);

(1文字を入力する場合 char moji )  
 moji=keybd.charAt(0)

(注) キーボードから入力を行う場合は、main メソッドの行末に throws Exception を付ける。入力エラー(例外)に対応するために必要。