

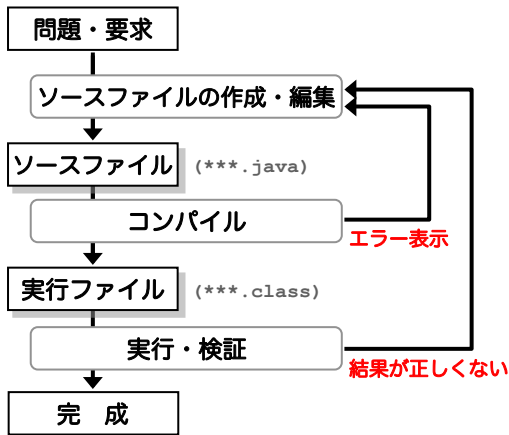
Java プログラミング入門

— Java プログラミングの基礎：コンパイル・実行 —

早稲田大学

プログラムの作成手順

Java 言語におけるプログラムの作成手順



ソースファイルの作成・編集

例題

画面上に“Hello.”というメッセージを表示させなさい。



ソースファイルの編集

- テキストエディタを使用し，ソースファイルを編集する
- エディタは，emacs, gedit など



Hello.java

```
public class Hello {
    public static void main(String[] args) {
        System.out.println("Hello.");          /* メッセージの表示 */
    }
}
```

コンパイル

`javac` : Java 言語のソースファイルをコンパイルする

[使い方] `javac` <Java 言語のソースファイル> ↵

ソースファイル

Hello.java



コンパイル

```
$ javac Hello.java ↵
```

```
$
```




クラスファイル

Hello.class

実行・検証

`java` : Java 言語のクラスファイルを実行する


[使い方] `java` `<クラスファイル (.class はつけない)>` 

クラスファイル

`Hello.class`



実行

```
$ java Hello   
$
```



検証・完成

- 正しく動作していない場合は、はじめに戻る

ソースファイルの形式

クラス名.java

```
import ライブラリ名; ← 省略可能
```

```
public class クラス名 {  
    public static void main(String[] args) {
```

ここにプログラムの本体を記述

```
    }  
}
```

注意点

- 大文字と小文字は区別される。
- Java ではクラス名の最初の文字を大文字にする習慣がある（例：Hello）
- ファイル名は“**クラス名.java**”とし、クラス名と一致させる

プログラムの記述 (1)

文は基本的に “;” (セミコロン) で終わる

```
System.out.println("Hello.");
```

```
int a = 1, b = 2, c;  
c = a + b * 3;  
c = (a + b) * 3;
```

プログラムの記述 (2)

```
/* コメント */      : コメント ( 注釈 ) の記入  
// コメント
```

```
System.out.println("Hello.");      /* メッセージの表示 */
```

```
/*  
    ここはコメントです。  
    ここもコメントです。  
*/
```

```
// ここはコメントです。  
    ここはコメントではありません。
```


プログラムの記述 (3)

改行 , インデント (字下げ) を行い読みやすくする

```
public class Hello {  
    public static void main(String[] args) {  
        System.out.println("Hello.");          /* メッセージの表示 */  
    }  
}
```

```
public class Hello { public static void main(String[] args) {  
System.out.println("Hello."); /* メッセージの表示 */  
}}
```

- 上の2つは同じプログラム . どちらも正しく動作する .
- 下の方が読みづらいことが分かる .

プログラムの記述（４）

エラーの原因となるよくある間違い

- ファイル名とクラス名が違っている
- 文の終わりにセミコロンがない
- メソッドや変数のスペルに間違いがある
- 括弧の対応が取れていない
- 全角スペースが入っている