

u582●パッケージ対応

パッケージ化

○パッケージとは

Javaで提供されているクラスや他の人の作成したクラスを利用しながらプログラムを開発していく場合クラス名やインターフェース名が自分の作成したものと重複する可能性があります。

こうした事態を防止するためにJavaにはパッケージの機能が用意されています。

パッケージは複数のクラスをグループ化しグループごとに名前空間を分離する働きをもっています。

○パッケージの作成

以下のように自分で作成したクラス群をパッケージにまとめるにはキーワードpackageを使用します。

packageキーワードを用いて自分の属する名前空間を". (ピリオド)" で区切って階層化して指定します。

```
【例】 package tokyopc.pinfo.gas
import java.awt.*
~~~~~
```

package指定をしたクラスはjavacコマンドの-dオプションを利用してコンパイルします。-dオプション付きのコンパイルで名前空間に対応したディレクトリにクラスが保存されます。

○パッケージのロード

パッケージの名前空間を指定したクラスは同じパッケージ内ではそのまま使用できます。

外部パッケージから使用するには以下のような2つの方法で明示的にパッケージをロードしなくてはなりません。

```
【例】 ①import tokyopc.pinfo.gas.*
~~~~~
~~~~~

②tokyo.pc.pinfo.gas.GasStation
aGs=new tokyopc.pinfo.gas.GasStation
~~~~~
```

アクセス制限

○変数とメソッドのアクセス制限

クラス内の変数とメソッドは ". (ピリオド)" 演算子を使ってクラス外のプログラムからも使用できます。

しかしオブジェクト指向ではクラス内の変数などオブジェクトの持っているデータを隠蔽し限られたメソッドや変数のみを提供するのが正しい考え方です。

こうしておくとおブジェクト内部でのデータ表現方法を変更しても利用する側には影響しないなどといったメリットがあるからです。

言語の構文上でオブジェクトの変数に外部から無条件にアクセスを許すことはこの概念を徹底できずオブジェクト指向プログラミングの良さが半減することを意味します。

そこでJava言語では変数やメソッドに3つのキーワードを指定することによりアクセス範囲を限定できるようになっています。

privateキーワード – class only

- それらが定義されたクラス内のメソッドからのみアクセス可能

default(アクセス制御の指定なし) – package only

- 同一パッケージ内からのみアクセス可能

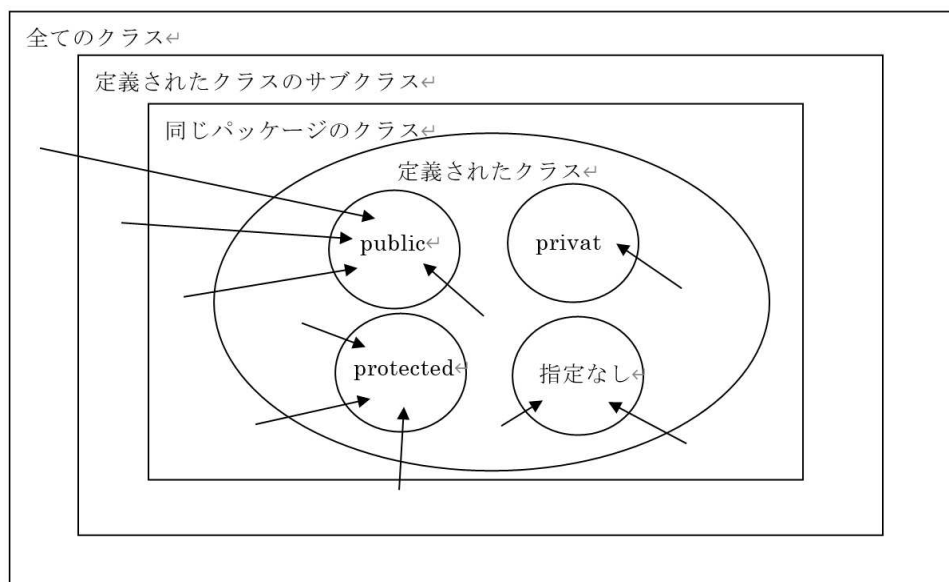
protectedキーワード – subclass & package only

- それらが定義されたクラス内のメソッドからアクセス可能
- そのクラスから派生したサブクラスからアクセス可能
- 同一パッケージ内のクラスからアクセス可能

publicキーワード – all ok

- どこ（どのオブジェクト、どのメソッド）からでもアクセス可能

private → default → protected → public の順でアクセス権は広がります



今後作成するJavaのプログラムはローカル実行形式であれサーバサイド実行形式であれpackage対応にします。

(パッケージ化されていないとServletは実行できません)

また、**コンパイル時のカレントディレクトリの位置も**
~/WEB-INF/classesとなります。

パッケージ化されたJavaプログラムを実装・起動

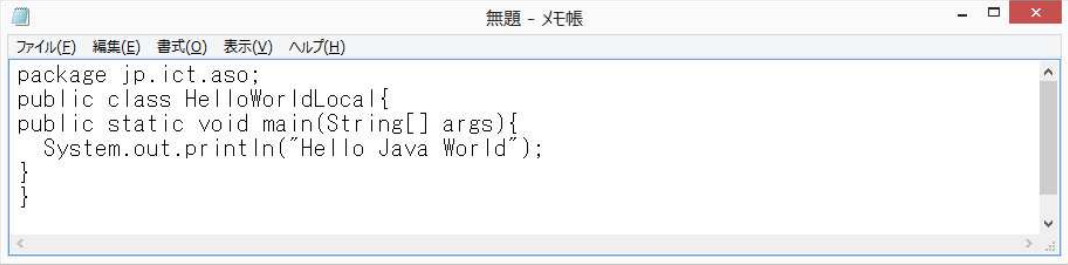
パッケージ対応にしたローカル実行形式の **HelloWroldLocal.java** を作成します。

以下のようにコンパイルして実行してください。

※パッケージ化されたクラスは **javac -d ファイル名.java** でコンパイルします。

実行するときは**完全修飾クラス名での実行**が必要です。

[root@aso classes]# **vi HelloWorldLocal.java**



```
package jp.ict.aso;
public class HelloWorldLocal{
public static void main(String[] args){
    System.out.println("Hello Java World");
}
}
```

[root@aso classes]# **javac -d . HelloWorldLocal.java**

[root@aso classes]# **java jp.ict.aso.HelloWorldLocal**

Hello Java World

[root@aso classes]#