package com.example.demo.controller

Javaのクラスファイル

“com.example.demo.controllerというパッケージに属している

パッケージ　関連するクラスをグループ化する方法で、クラス名が

重複するのを防ぐことができる

パッケージは逆順のドメイン名がよく使われている

例えば、com.example.demoはexample.comドメインでdemoアプリケーションを開発している事を示している

import org.springframework.stereotype Controller;

Spring MVCのコントローラクラスである事を示すアノテーション

org,springframework.ui.model

ビューにデータを渡すために使用するクラス

これらのクラスをインポートする事でspring MVCでwebアプリケーションを開発するために必要な機能を利用できる。

アノテーションとは、Java言語の機能の一つで、コンパイルや実行時に特定の情報を与えるためのもの

アノテーションを利用することで、開発者がコンパイルや実行時に必要な情報を明示的に記述することができる

アノテーションは＠マークで始まる特殊な構文で表現される

Spring Frameworkでは@Controller @RequestMappingといったアノテーションが使用される

アノテーションを自作することができる、自作アノテーションを利用することで

特定の処理を自動化したり、メタデータを記述することができる

import org.springframework.web.bind.annotation.GetMapping;

SpringFrameworkのアノテーションである@GetMappingを使用するために必要なimport文

@GetMappringはHTTP GETリクエストを処理するために使われる

つまり、指定されたURLにGETリクエストが送信された場合、関連するメソッドが呼び出される

@Controller

Springframewaorkのアノテーション

このクラスがコントローラであることを示している

public class WebController {

WebControllerっていうクラス

このクラスの名前であり、publicはアクセス修飾子

Javaのアクセス修飾子

1 public どこからでもアクセス可能

2 protected 同じパッケージ内、または異なるパッケージであっても、サブクラスからのみアクセス可能

3 default　同じパッケージ内からのみアクセス可能、修飾子が指定されていない場合、自動的にデフォルトのアクセス修飾子となる

4 private 同じクラスからのみアクセス可能

特殊な修飾子

1 final 変数、メソッド、クラスに対して使用できる、変数にfinal修飾子を付けると

変数の値を変更できなくなる

メソッドにfinal修飾子を付けると、サブクラスでオーバーライドできなくなる

クラスにfinal修飾子を付けると、そのクラスを継承できなくなる

2 abstract　クラス、メソッドに対して使用できる

クラスにabstract修飾子を付けると、そのクラスは抽象クラスになる

抽象クラスは具象クラスの基底として用いられるが、直接インスタンス化することはできない

メソッドにabstractを付けると、そのメソッドは抽象メソッドになる

抽象メソッドは実装を持たず、具象クラスでオーバーライドする必要がある

3 static 変数、メソッド、ブロックに対して使用できる、変数にstatic修飾子を付けると、その変数はクラス変数となり、クラス全体で共有される

メソッドにstatic修飾子を付けると、そのメソッドはクラスメソッドとなり

インスタンス化せずに呼び出すことができる

ブロックにstatic修飾子を付けると、クラスが初期化されるときに実行される