

PlayFramework メモ

何度目かの PlayFramework 入門．とりあえず sbt を用意した状態からスタート．

プロジェクト作成

```
sbt new playframework/play-scala-seed.g8
```

プロジェクト名は適当に決める．

プロジェクト名と同名のディレクトリが作られるので，そこに移動．

HelloWorld する

conf/routes に URL と，その URL にアクセスしたときに実行したいメソッドの対応を書く．
たとえば，

```
GET    /my-hello                controllers.MyHello.hello
```

とか書く．

で，メソッドを用意．クラスは app の下に用意．今回は app/controllers/MyHello.scala ．

中身は，

```
package controllers

import javax.inject._
import play.api.mvc._

import play.api.data._
import play.api.data.Forms._

class MyHello @Inject()(mcc: MessagesControllerComponents)
  extends MessagesAbstractController(mcc){

  def hello() = Action { implicit request: MessagesRequest[AnyContent] =>
    Ok(views.html.myhello("My Hello World", "miyo"))
  }
}
```

とか．ここで，views.html.myhello は，app/views/myhello.scala.html に定義する．

中身は，

```
@(mesg1: String, mesg2: String)
<html>
<head><title>MyHello</title></head>
<body>
@mesg1 from @mesg2 <br>
<ul>
<li>mesg1.lenth = @mesg1.length</li>
<li>mesg2.lenth = @mesg2.length</li>
<li>mesg1.length + mesg2.length = @(mesg1.length + mesg2.length)</li>
</ul>
</body>
</html>
```

とか．HTML に Scala を埋め込んだもの．

必要があれば，@import で，Scala の import 文が書ける．

@() の中で定義した変数が Scala から引き渡されて，html 中で使用できる．

```
<li>mesg1.length + mesg2.length = @(mesg1.length + mesg2.length)</li>
```

は、

```
<li>mesg1.length + mesg2.length = @(@mesg1.length + @mesg2.length)</li>
```

や

```
<li>mesg1.length + mesg2.length = @mesg1.length + @mesg2.length</li>
```

ではないので注意。

実行する

```
sbt run
```

で実行して、ブラウザで

```
http://localhost:9000/my-hello
```

にアクセス。エラーがでたら適宜修正。

ブラウザでリロードすると裏でコンパイルプロセスが走って、再実行される。

本来の開発の順番は

逆がよい。

ビューとアクションを用意してから route を書くみたい。あたり前か。