

Javaのフレームワーク

1. JSF(java server faces)

○概要

- ・ javaベースのwebアプリケーションフレームワーク
- ・ javaの標準フレームワーク

○特徴

- ・ MVCモデルを採用
 - ・ 表示する画面に対応するサーバーサイドのクラス（バックングbean）によって処理が行われる「コンポーネントフレームワーク」に属する
- ・ MVCのView(表示)では多くのフレームワークがJSPで実装されるのに対し、XHTMLで実装される。前者はサーブレットなしに正しい処理は行われない可能性があるが、後者はブラウザ上で確認することができデザインの確認が行いやすい

○導入方法

- ・ netbeansにて

新規プロジェクト>java web/webアプリケーション>[フレームワークの設定]java server faces

ファイル>新規ファイル-ファイル>java server faces-JSFページを選択する

2. javaEE(java Platform,Enterprise Edition)

○概要

- ・ 大規模システム向けのフレームワーク
- ・ javaの標準フレームワーク

○特徴

- ・ ServletやJSP、WJBやjavaの各種APIなど企業システム開発に必要な機能を一通り備えられている
- ・ oracle社支援のもとノウハウが集まっている

○導入

新規プロジェクト>java web/webアプリケーション>java EEアプリケーションサーバを設定>[フレームワークの設定]java server faces

ファイル>新規ファイル-ファイル>java server faces-JSFページを選択する

3. Spark Framework

○概要

- ・ webマイクロフレームワーク

○特徴

- ・ シンプルなプログラミングや迅速なシステム開発を目的にしたフレームワーク
- ・ 膨大なアノテーション記述や設定ファイルが不要となっており開発者の負担を軽減

減する

- ・ 必要最低限の機能しかないので中小規模のシステム開発に向いている

4. Play Framework

○概要

- ・ Java以外にもscalaというJavaに親和性の高いプログラミング言語を使用できる

○特徴

- ・ 軽量さと高い生産性
- ・ Java EEに準拠しておらず柔軟なアプローチが可能になる
- ・ 動作時にCPUリソースやメモリ使用量の負荷が小さいためハードウェアの負担が小さい

○導入

5. Spring Framework

○概要

- ・ DI(dependency injection)コンテナという環境に依存する機能を外部から提供可能にする方法をもつ

○特徴

- ・ 7種類のモジュールからなる

オブジェクトの関連付け、生成、登録などを管理する「coreモジュール」

javabeansへのアクセスを可能とする「contextモジュール」

MVCモデルによる開発を支援する「webMVCモジュール」

○導入

新規プロジェクト>java web/webアプリケーション>サーバー : glass fish server/java EE7>[フレームワークの設定]spring web MVC>spring framework3.0を選択
以下、コントローラーモデルビューを実装する