

# Java-フレームワークを学ぼう！

## 【フレームワークの選び方のポイント】

### ―― 1. 目的で選ぶ ～必要な機能は備わっているかな？～

フレームワークによって得意とする部分が違うため、開発者の知識や開発を担当する機能に応じて考えると良い。

例えば・・・ ①JSFは見た目に関する機能を提供している

②最低限を知っていればすぐ使い始められるような軽量さがある

③学習量は必要だが多機能である

など

### ―― 2. オススメやトレンド、将来性から選ぶ ～トレンドになるにはそれなりの理由がある！～

先輩方のオススメや、世間のトレンド（検索数が高いなど）から決めるのも良い。

例えば・・・ ①コードが書きやすい、テストがしやすい

②開発が活発であるため参考にできる情報やコミュニティが充実している など

### ―― 3. 学習環境で選ぶ ～自分や開発メンバーに適しているかな？～

そのフレームワークを学ぶためのスキルアップの環境が用意できるかで決めるのも良い。

例えば・・・ ①いざとなった際に身近に相談できる人がいる

②研修やスクールで身につけられる環境がある

など

## 【フレームワークの大枠】

### ●Webアプリケーションフレームワーク

Webアプリケーション作成のために用いられるフレームワーク。

一般的にフレームワークという場合、Webアプリケーションフレームワークを指す。

### ●ユーティリティ系フレームワーク

データのソートや文字列操作、数学関数、テストなど、アプリケーションの機能や性能、操作性を向上させるためのフレームワーク。

## 【主なWebアプリケーションフレームワーク】

### ■JSF (JavaServer Faces)

ーJavaベースのWebアプリケーションフレームワーク MVCモデル採用

ーJavaプラットフォームJava EE (Java Platform, Enterprise Edition) に含まれている

ースタンダードな技術で採用している企業も多い

ーユーザーインターフェイス（ユーザーが操作する見た目の部分）を簡単に作ることができる

ー開発チームとデザインチームで分業できる特徴があり、Strutsに比べて分業が行いやすい

### ■Spring Framework

ーJavaプラットフォーム向けのオープンソースアプリケーションフレームワーク

ーカバーしている範囲が広く、拡張性、保守性、再利用性を合わせ持つ万能型

ーDI (Dependency Injection: 依存性注入) 機能を備え、単体テストに優れている

### ■Play Framework

ーJavaおよび関数型言語のScalaというプログラミング言語をサポートしている

Webアプリケーションフレームワーク

ーJava EEの中核機能は必要としないため比較的シンプルで開発しやすいと言われる