

# Javaフレームワーク

## ・Spring Framework

データベースアクセスやトランザクション処理などをサポートしているフレームワークなど多数のフレームワークの集合体となっている。javaのWebフレームワークで、webアプリを作ることができる。

使う理由:

変更に強い(DI(Dependency Injection)と呼ばれる仕組みが導入されていて、依存性のあるプログラムなため、外から取り入れることができることから一つ一つのプログラムを独立しやすい。また、**最低限の変更で済むので開発が楽。**)

テストが簡単(専用のテストプログラムがある。(Spring MVC Test)無事に一連の処理を終えるかどうかをテスト(結合テスト)することができるため**安全。**)

特徴:

**拡張性**(基本的な機能の大部分が**(インターフェース)**として提供されていて、必要な機能だけを外部から使うことができるから無駄がなく、必要な機能はあとから簡単に追加することができる。オブジェクト指向の基本原則を徹底して**Javaを最大限に生かすことができる**ように設計されている。)

**保守性が高い**(**AOP(Aspect Oriented Programming)**=クラスには本質的な処理のみを書いて、本質的ではない処理(共通の処理)を別を書く)というプログラミングを用いることができる。主に必要となる処理とそうではない処理を分けることで、**コードをわかりやすくする**効果があります。それによって、何かバグや不具合があった場合にも**原因を特定しやすくなる**のです。)

**再利用性が高い**(**AOP**の仕組みにより、共通のプログラムをまとめやすくなるので、**再利用することも簡単**になりる。再利用することで、プログラムを書く量が減り、修正する場合も最低限の修正で済むようになりる。とくに長期のアプリ開発でものすごく**効率がよくなる**。再利用性の高いプログラムを、**プログラマーの腕ではなく仕組みで解決している**。)

## ・Play Framework

プログラミング言語の「Java」と「Scala」を使って作られたWEBフレームワーク。Javaの基礎的な知識があれば使い始められる。他のフレームワークよりも**軽量なため、利用しやすく学習コストも低め**。**webアプリ、webアプリのバックエンド、スマートフォンアプリと通信するバックエンド**などに使われる。(バックエンド＝ユーザーの目に直接触れない縁の下の力持ち的な部分のような。)

使うメリット:

**高速に開発できる**(高速で開発できる点が最大のメリットです。Ruby on RailsやDjangoと同じく「MVC」という設計に基づいて、WEBアプリ開発を高速で行えます。費用の制限がある企業向けのシステムや、スピードある開発を求められるWEBサービスなどでは高速で開発できることが何よりのメリットで、**速いことは最大の武器**にもなります。Javaのコンパイルに時間がかかるというデメリットも最大限に抑え、メリットである**堅牢性や高速な動作を最大限活かせる**ようになっている。)

**リソース消費が少ない**(少ない資源で動けるので、無駄がなく速い、さらに、CPUやメモリーに負担が少なく、**少ないリソースでアプリを動作させること**ができる。規模がどんどん変化していくWEBアプリでは重要。開発が進み、大規模なシステムに成長したとしても問題なく動作するWEBアプリを作れるので、将来のことは心配しなくても良い。

## ・Java EE(Java Platform, Enterprise Edition)

ServletやJSP、EJBやJavaの各種APIなど、企業システムの構築に必要な機能が標準化したフレームワークがまとめて提供されていて。具体的に標準化しているのは企業システムの機能として実装される **コンポーネント**を作るための約束事(仕様)とコンポーネントからデータベースなどの **インフラ・サービス**にアクセスするための **API(Application Programming Interface)**。

・**移植性(ポータビリティ)の高さ**(上のふたつが標準化されてるため、一度開発したアプリケーションを、Java EEに対応したさまざまな **実行環境(アプリケーション・サーバ)の上で動かすことが可能** となるのです。開発したアプリケーションを、サーバ・マシンやアプリケーション・サーバを変えても引き続き使い続けられるため、企業は長期間にわたってソフトウェア資産を保護し続けることができる。)