

フレームワークとは

フレームワーク(framework: 枠組み)は、特定のプログラミング用語で書かれた、半完成品のソフトウェアのこと。フレームワークのなかには、

- 様々な開発で使える、再利用可能な機能(クラス、ライブラリ、モジュール、APIなど)
- そのまま利用できる主要部分のひな形(テンプレート)
- 開発者が機能を追加するための仕様や規約

が含まれています。これら既存の機能に加えて、開発者がさらに機能を追加することにより、目的のソフトウェアをより早く完成させることができます。

機能の集まりといえばライブラリやモジュールを連想するかもしれませんが、違いは

- ライブラリ: 全体を制御するコードは開発者が記述、開発者側がライブラリの機能を呼び出す
- フレームワーク: 全体を制御するのはフレームワーク側のコード、個別に追加された機能を呼び出す

というように、制御のためのコードが**フレームワーク内にある点**です。

* * 各種フレームワークには得意・不得意がある。

MVCモデル(Model View Controller) :

①Model

>>コントローラの指示を受け、自身が担当するデータの関連処理を行います。DBから該当データを引き出したり、データを加工する事を担当します。

②View

>>各ページのデザイン情報が書かれた、テンプレートクラス。このクラスにモデルなどで処理したデータを埋め込み、ユーザーに表示します。

③ Controller

>> 利用者からのWebアクセスを仲介し、モデルやビューに指示を出します。

フレームワークを用いるメリット・デメリット

メリット

- 開発期間を短縮化
- チーム内で共有、品質向上
- バグが少ない

デメリット

- コーディングスキルが比較的向上しづらい
- Javaの深い理解が得られづらい
- フレームワークによる脆弱性が避けられない

Javaフレームワークの種類(一例)

Play Framework
Spring Framework
Apache Struts
JSF
Wicket