

## 1. オブジェクト指向とは何か a.特徴3つ、またその説明、b.具体例

オブジェクト指向プログラミングとは、コンピュータプログラムの構造、構成法の一つで、関連するデータの集合体と、それを操作する手続きを「オブジェクト」(object)と呼ばれるひとまとまりの単位として一体化し、オブジェクトの組み合わせとしてプログラムを記述する手法。

また、オブジェクトの定義と、オブジェクト間の関係、相互作用を記述することによりプログラムを構築していく。オブジェクトにはそれぞれ固有のデータ(属性/プロパティ)と手続き(メソッド)があり、外部からのメッセージを受けてメソッドを実行し、データを操作する。オブジェクトに付随するデータの操作は原則としてすべてオブジェクト中のメソッドによって行われる。

### 1) カプセル化

オブジェクトの安全性を高めるために、オブジェクトのメンバーを保護すること

※メンバ = オブジェクトのメソッド及び変数

この性質を理解し、影響範囲の小さく疎なプログラミングを心がけることが、大規模で安全なシステム構築に繋がる。

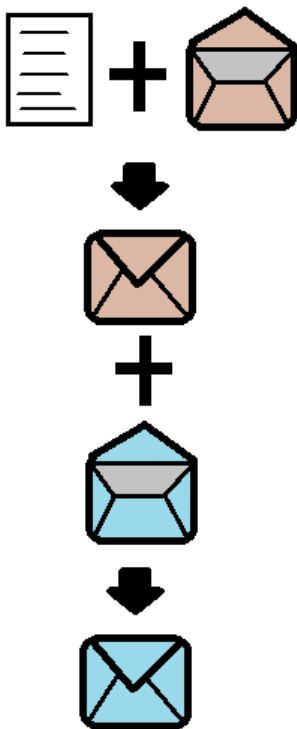
→ 封筒に入ったお手紙を、さらに新しい封筒の中に入れること。

もう少し具体的に書くと、

パケットを包み込んだパケットを作ること

であり

新しいヘッダ部を付与して、元々のパケットをデータ部と見なしたパケットにすること。



### 2) 継承 (簡単に言うと、クラス定義の共通部分を別クラスにまとめる仕組みのことをいう。)

- ・コードの重複を減らしてコードの再利用性を高める
- ・メソッドを追加したり、オーバーライドすることで自由に拡張する。

この特徴があるおかげで、共通部分をまとめて、再利用することができる。

3) オーバライドやオーバーロードによって、メソッドを状況によって使い分けられること

オーバライド: 親クラスのメソッドを子クラスで書き換えること

オーバーロード: 同一クラス内で引数の違いによってメソッドを呼び分けること

「オーバーライド」って単語が出てきたら「子クラスで親クラスのメソッドを再定義することなんだな。」と、考える。

また、「動物」で表すと、

【生産】動物が生まれる

【動作】動物が前に歩く

【変化】動物が歳を取っていく

【破壊】動物が死ぬ

変動の本質は似ているけど、振る舞いが全く違います。

これらの振る舞いをモノの性質毎に分けて処理する枠組みを「オブジェクト指向」で構築する。

参考文献：おうちで学べるプログラミングのきほん 単行本 2013/4/23 河村 進

## 2. Github flow とは何か

GitHub Flow（ギットハブ フロー）とは、ブランチ戦略の1つであり、その名の通りGitHubで利用されているとてもシンプルナイト開発フローです。

GitHub Flowではメインのmasterブランチと機能開発のためのfeatureブランチの2つのみのシンプルな構成になっている。

## 3. サーバサイドエンジニア、フロントエンジニアとはどのような違いがあるか

- ・サーバサイドエンジニアとはサーバサイドの処理を実装するエンジニアのこと。

フロントエンジニアがWebのユーザーから見える部分の開発を行うのとは正反対に、サーバ側のプログラムの実装やデータ処理などを行うのがサーバサイドエンジニアである。

そのため、サーバサイドエンジニアはフロントエンジニアと対比するエンジニア職として解説されるケースが多い。

- ・フロントエンドエンジニアとは、フロントエンド側を担当するエンジニアのこと。

フロントエンド（front-end）とは、ユーザーと直接データのやり取りを行う要素のことで、Web制作ではWebブラウザ側（クライアント側）を指す。

また、フロントエンジニアは主にHTMLやCSS、JavaScriptなどを使用し、クライアントの要望を反映させたWebサイト開発を行う。

<仕事内容について>

- ・フロントサイドエンジニアはHTMLやCSS、JavaScriptなどを使用してWeb開発を行うのが仕事で、Web開発のフロントサイド側の開発を行う人材で、ユーザーの目に見える部分を実装する。

そのため、フロントサイドエンジニアにはHTML、CSS、JavaScriptの知識やWordPressを使用した実装スキルなど、Webサイト構築に必要なスキルが求められます。

簡単に言うと・・・

フロントエンドエンジニア = ユーザーが見ている画面のデザイン

サーバサイドエンジニア = サーバーでプログラムの実行・管理

## フロントエンド



マイページ情報ちょうだい！  
(リクエスト)



マイページ情報だよ！  
(レスポンス)

## サーバーサイド



フロントエンド「〇〇の情報送って下さい！」

サーバーサイド「これが〇〇の情報です！」・・・というような感じ。

参考文献：フロントエンドエンジニアのための現在とこれからの必須知識 単行本 2016/1/28

斉藤 祐也 (著, 編集), 水野 隼登 (著), 谷 拓樹 (著), 菅原 のびすけ (著), 林 優一 (著), 古沢 宏太 (著)

## 4.AWSとは何か

AWSとは、クラウドコンピューティングを使ったサービスのこと。

AWSとはAmazon Web Servicesの略で、Amazonが提供している100以上のクラウドコンピューティングサービスの総称。

クラウドコンピューティング：インターネットを介してサーバー・ストレージ・データベース・ソフトウェアといったコンピューターを使った様々なサービスを利用することを指す。クラウドコンピューティングでは、手元に1台のPCとインターネットに接続できる環境さえあれば、サーバーや大容量のストレージ、高速なデータベースなどを必要な分だけ利用できる。

5つのポイント

- ・豊富なサービス
- ・世界最上級の運用専門知識
- ・最もセキュアなセキュリティ基準
- ・安心できる情報網（コミュニティやパートナーの存在）
- ・まったく新しいテクノロジーを最速で提供し続けるサービスコンセプト

コスト削減や俊敏性を高めビジネスの効率化、ビジネススピードの加速化しイノベーションの加速を実現。

参考文献：<https://www.skyarch.net/column/whataws01/>

## 5.Dockerとは何か

dockerとは

### ①システム導入の高速化

OSは既に共有して利用しているため、それらの設定は省き、システムを構築するために必要となる最低限のプログラムしかインストールする必要がない。

そのため、すぐにプログラムを適用し、導入までのスピードを高速化することができます。

### ②起動時間の短縮化と多くの処理実装が可能である。

完全仮想化であれば、サーバを起動する際にOSレベルから立ち上げていく必要があり時間が掛かるが、一方DockerではOSは既に立ち上がっているため、その分サーバの起動時間を短縮化することができる。

また、リソースの使用量が少なく済むためサーバへの負荷が低く、1度に多くのプログラム処理を実装することが可能である。

### ③コンテナ設定の再利用が可能

1度作成したコンテナは、Dockerイメージを作成し、他のコンテナへ適用することにより再利用が可能となる。検証してみたい時や、リソースを拡大したい時など、すぐに同じ環境設定が適用されたコンテナを準備することができる。