

抽象化

- ・すべては抽象化に関するものです。
- ・デザイン、ソフトウェア、製品、物理学、近視の逆転、スタートアップ、ビジネス、ユーザー、戦略、国々、プログラミング言語、変数、そして概念は、すべて抽象化に関するものです。
- ・抽象化とは、メタ思考の一形態です。
- ・物事を高いレベルで抽象化することで、それらを要約し、一括して扱うことができます。それは関数やメソッドのようなものです。それは文章の要約のようなものです。それは目次のようなものです。それは会社の組織図のようなものです。
- ・異なる側面から物事を抽象化することができます。
- ・抽象化の基準は、私たちが考えるべきことです。それは、コードをどのように抽出または整理するかというようなものです。
- ・レベルごとに物事を整理したいと考えています。各レベルは隣接するレベルとのみ相互作用するべきです。
- ・問題やバグが発生した場合、高いレベルから始めて問題を見つけることができます。根本原因まで追跡する必要があります。これは国の腐敗に似ています。腐敗は通常、特定のグループや人々の中で起こります。
- ・抽象化の欠点は、いくつかの高レベルの概念を含むことです。この世界では個人がより多くの自由を持っています。高レベルに対するルールは、そのレベルに含まれる全てのオブジェクトに適しているとは限りません。
- ・物事は変化します。より良い抽象化のためにコードをリファクタリングする必要があるかもしれません。ビジネスをより良く反映し、顧客にサービスを提供するために会社の構造を再編成する必要があるかもしれません。
- ・各レベル間の標準について考えてみましょう。それらには何らかのパターンがありますか？もあるなら、高次の関数を構築するためのラムダコードとして記述することができます。
- ・黒板（5m離れた場所）と携帯電話を見るために完全な処方の眼鏡をかける人々は、抽象化が不十分です。それは眼球の利益をより良く反映していません。
- ・その抽象化を適用するための関数、メソッド、手順は慎重に検討する必要があります。オブジェクトのバッチ全体に対して同じタスクを行う必要があるのか、それとも個別に扱うべきなのかを考えるべきです。