

# 抽象化

- 一切皆關乎抽象。
- 設計、軟件、產品、物理、逆轉近視、初創企業、商業、用戶、策略、國家、編程語言、變量及概念，皆以抽象為核心。
- 抽象是一種元思維方式。
- 通過高層次的抽象，我們能概括事物並批量處理。這如同函數或方法，如同寫作中的總結，如同目錄，如同公司架構。
- 我們可以從不同角度對事物進行抽象。
- 抽象的標準是我們需要思考的，這就像我們如何提取或組織代碼。
- 我們希望分層次組織事物，每一層只與相鄰層次互動。
- 若出現問題或錯誤，我們應從高層次開始追查，直至找到根本原因。這就像國家的腐敗，通常發生在某些特定群體或個人中。
- 抽象的缺點在於涉及一些高層次概念。在這個世界中，個人擁有更多自由。高層次的規則可能並不適合所有包含在該層次中的對象。
- 事物在變化。我們可能需要重構代碼以獲得更好的抽象，可能需要重組公司架構以更好地反映業務並服務客戶。
- 思考每一層次之間的標準。它們是否有某些模式？如果有，可以將其編寫為 lambda 代碼，以構建高階函數。
- 戴著全矯正度數眼鏡看黑板（5米遠）和手機是一種糟糕的抽象，它未能更好地反映眼球的利益。
- 應用該抽象的功能、方法和程序應仔細考慮。我們是否需要對整批對象執行相同任務，還是應該分別對待？