

해커와 화가

01. 공부벌레는 왜 인기가 없을까

- 공부벌레는 낙오자가 아니다.

02. 해커와 화가

- 해킹과 그림은 공통점이 많음
- 무언가 아름다운 것을 만드는 방법은 이미 존재하는 것을 살짝 뒤틀거나 알려진 아이디어 몇개를 새로운 방식으로 결합하는 것
- 아름다운 소프트웨어인지 아닌지 측정할 수 있는 유일한 외적인 방법은 시간이다.
- 해커는 자신의 평판을 스스로 어쩔 수 없다는 것을 받아들여야 한다.
- 프로그래밍 언어는 존재하지 않는 프로그램을 생각해내기 위한 도구이다.
 - 볼펜이 아니라 연필이다.
- 회사는 해커가 하고싶은 일을 하도록 내버려두지 않는다.
- 언젠가 큰 돈을 벌고싶다면 산출물을 내는 개발자를 여럿 두지 말고 한명을 뒤라
- 훌륭한 소프트웨어를 만드는 방법은 자기 자신의 스타트업 회사를 만드는 것이다.
- 훌륭한 해커들은 좋은 프로그램을 들여다보며 배운다. (소스코드)
- 그림은 점진적으로 완성된다
 - 프로그램을 위한 스펙 요구사항이 완벽할 것이라고 기대하지말고 스펙이 바뀔 수 있다는 것을 수용할 수 있는 방식으로 프로그램을 짜는 것이 더 현명하다.
- 성능 최적화를 지나치게 일찍 시도하는 것은 위험하다.
- 열정이 식을만큼 잔잔한 업무를 할 때를 위해 할만한 단순한 일들을 남겨두는 것도 좋은 방법이다.
- 프로젝트를 아주 정밀하게 정의된 모듈로 구분하고, 각 모듈에 주인을 할당하고, 모듈 사이에 존재하는 인터페이스는 마치 프로그래밍언어 자체처럼 명확한 수준으로 정의하는 것이 좋다.
- 사물을 다른사람의 입장에서 바라보는 것이 성공의 비결이다.
 - 소스코드와 소프트웨어는 스스로를 잘 설명해야한다.

03. 우리가 말할 수 없는 것

체제 순응자 테스트

- 다른 동료들 앞에서 선뜻 꺼내기 곤란한 의견을 마음속에 품고 있는가?
 - 아니오 라면 다시 생각해보자
- 입 밖에 내지 못할 생각을 하지도 않는다는 것은 무언가 잘못되었다.

문제

- 말할 수 없는 것
 - 말을 함으로써 사람들이 곤경에 빠지는 것이 무엇인가 있을 때

이단

- 참과 거짓을 구분하기 금기되어있는 말들은 하면 안된다.

시간과 공간

- 지금 잘못하고 있는지 확인하는 방법중에는 과거에는 아무런지 않게 용납되었지만 현재 우리가 보기에 말도되지않는 방법이 무엇인지 살펴보는 방법이 있다.

잘난 체하는 사람

- 터부시되는 생각은 사색의 실험을 통해 찾을 수 있다.

메커니즘

- 터부가 생산되는 방식을 살펴보면 말할 수 없는 것을 알 수 있다
 - 도덕적 유행

왜?

- 왜 말할 수 없는 것을 찾아보아야할까
 1. 순수한 호기심
 2. 잘못 이해되는 관념이 존재
 3. 두뇌활동에 도움이 된다

- 쉼게 생각할 수 없는 것을 생각할 수 있도록 자신을 훈련시키는 일은 평상시의 생각을 뛰어넘도록 도와준다.

닫힌 사고

- 말할 수 없는 것을 발견했을 때 그것을 입밖에 내지 말거나 그 말에 어울리는 싸움터를 골라라

열린 얼굴

- 닫힌 사고만큼이나 열린 얼굴이 필요하지는 않는다.
- 은유, 유머, 솔직함 등으로 반격할 수 있다.

항상 질문하라.