

# 디자인이라는 도전!

호롤롤롤

# 디자인의 자연스러운 진화

- 좋은 디자인은 진화한다.
- 사용하면서 발생하는 문제점이 재수정되는 과정이 반복되어 기능적, 미적 모두 아름다운 물건이 만들어 진다.
- 장인이 잘못된 점을 기꺼이 고치려고 하면 자연스러운 진화가 이루어지는데 이를 HILL Climbing, 언덕오르기 라고 한다.

# 디자인의 자연스러운 진화

디자인의 진화를 가로막는 힘

- 1. 시간
- 2. 압박감
- 3. 개성



# 디자이너는 왜 올바른 길에서 벗어나게 되는가?

- 1. 디자인 사회의 보상체계가 미적인 것을 최고로 여김
- 2. 디자이너 자신은 대개 전형적인 사용자가 아님
- => 일상 생활의 심리학
- 3. 디자이너가 만족시킬 고객이 일반 사용자가 아닐 수 있다는 점. => 보통 구매자는 중간자 역할의 인물.

# 디자인 과정의 복잡성



# 디자인 과정의 복잡성

- 사용자가 수도꼭지를 잠그지 않으면?
- => 스프링으로 누르는 동안만 나오게
- 누르면서 씻기 힘들
- => 시계를 붙이자
- 그럼 너무 복잡함. 발로 제어할 수 있는 수도꼭지는?
- => 더 정밀하고, 비용이 많이 듦
- 센서로 손을 인식해 자동으로 켜지고 꺼지게
- => 1. 비용, 2. 비가시적
- .....
- 모든 수도꼭지가 공공장소에서 쓰이는 디자인은 아니지만, 동일한 요소를 고려해야 한다.

# 특정한 사람을 위한 디자인

- 인체 측정학에 따라 사람들은 모두 다르다.
- 100퍼센트에 맞는 공학적 설계는 없음.
- 같은 사람이라도 나이를 들면서 초점, 자세 등이 바뀌기 때문이다.



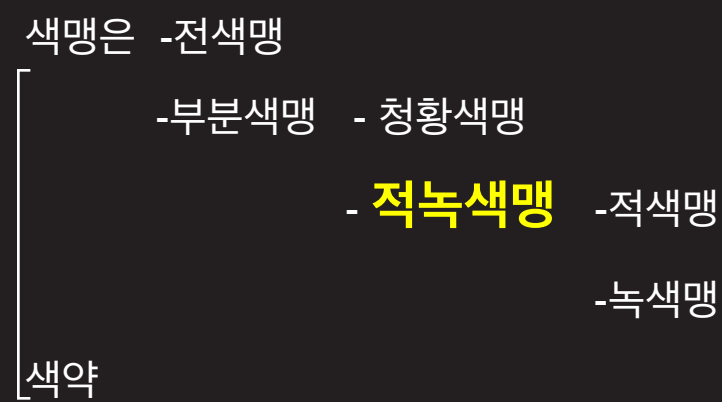
## 5.우리의 색상 지각은 제한적이다

### 5.4 색맹

디자인 원칙에 영향을 주는 요소 \_ 색맹을 가진 사람들이 구분할 수 있는 색상을 사용하였는가

색맹(색상 인식 결핍)\_색맹을 가졌다고 아예 볼 수 없음을 뜻하는 것이 아니라, 하나 혹은 그 이상의 대응색상 채널이 올바르게 작동하지 않아 특정 색상 조합을 구분하기 어려운 상태를 뜻한다. (남성의8%, 여성의 0.5%)

#### 색맹의 종류



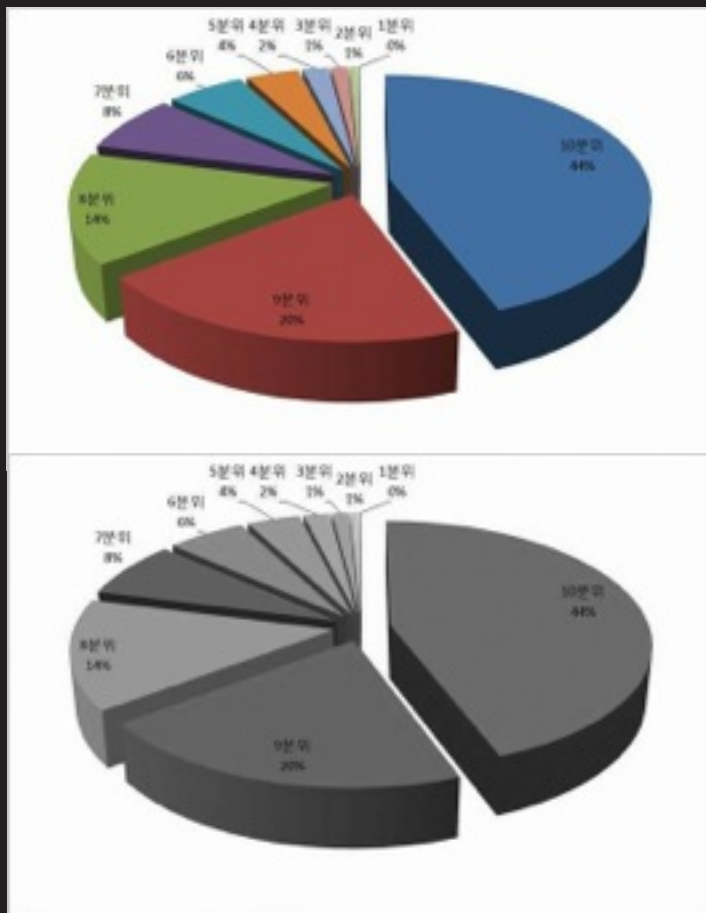
적녹색맹 \_어두운 빨강&검정 , 파란&보라, 밝은녹색&흰색 \_ 을 구별하지 못한다

## 5.우리의 색상 지각은 제한적이다

### 5.4 색맹

그래프에서 일부 사용자들이 구분하기 어려운 색상들을 사용하고 있다.

색맹을 가진 사람들이 구분하기 어려운 색상들로 그래프를 구성하고 있다.



## 5. 우리의 색상 지각은 제한적이다

### 5.6 색상에 대한 가이드라인



색상  
Hue



채도  
Saturation



명도  
Brightness

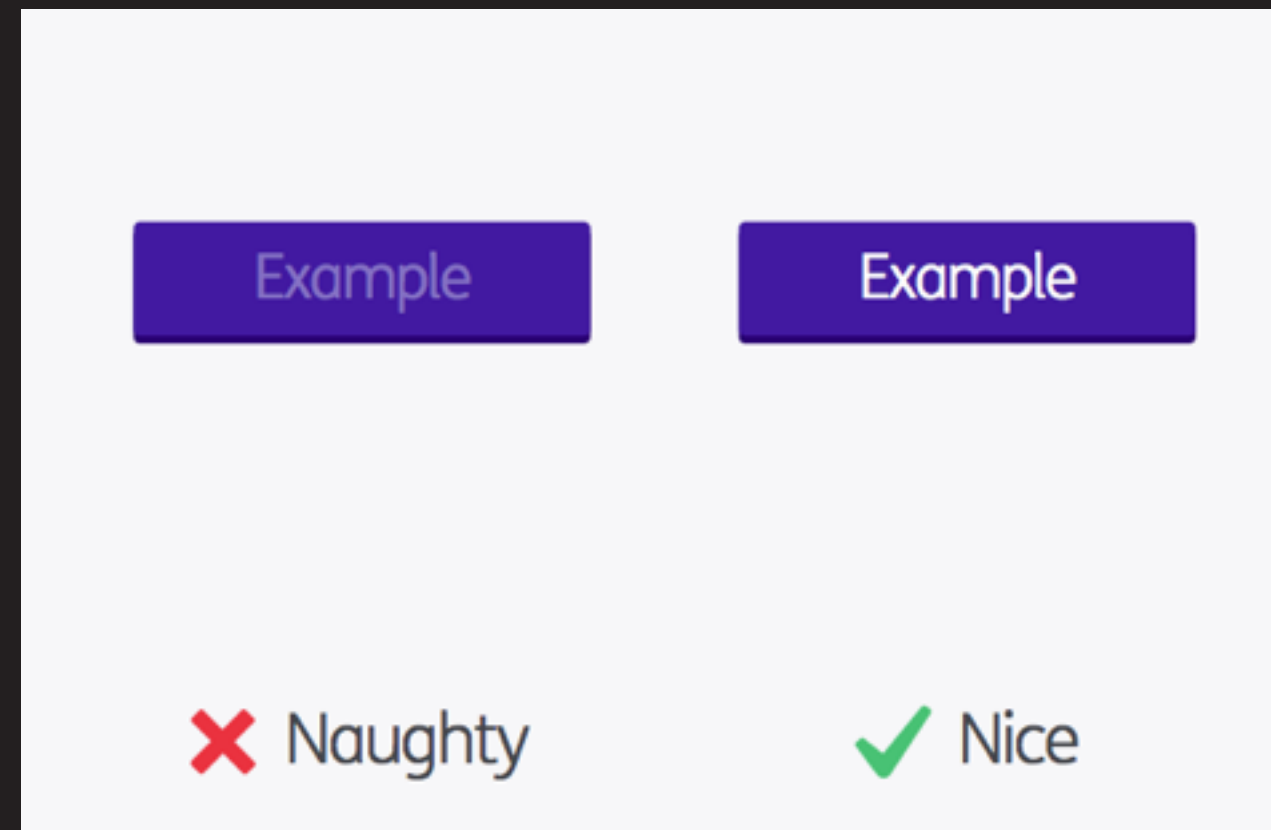
#### 1. 색상 뿐 아니라 채도와 명도에서도 구분하여 사용할 것

미묘한 색상 차이는 피하고 대비가 충분하도록.

회색조로 변환했을 때, 구분이 잘 되는가?

컬러 대비 체크 툴

[http://snook.ca/technical/colour\\_contrast/colour.html](http://snook.ca/technical/colour_contrast/colour.html)

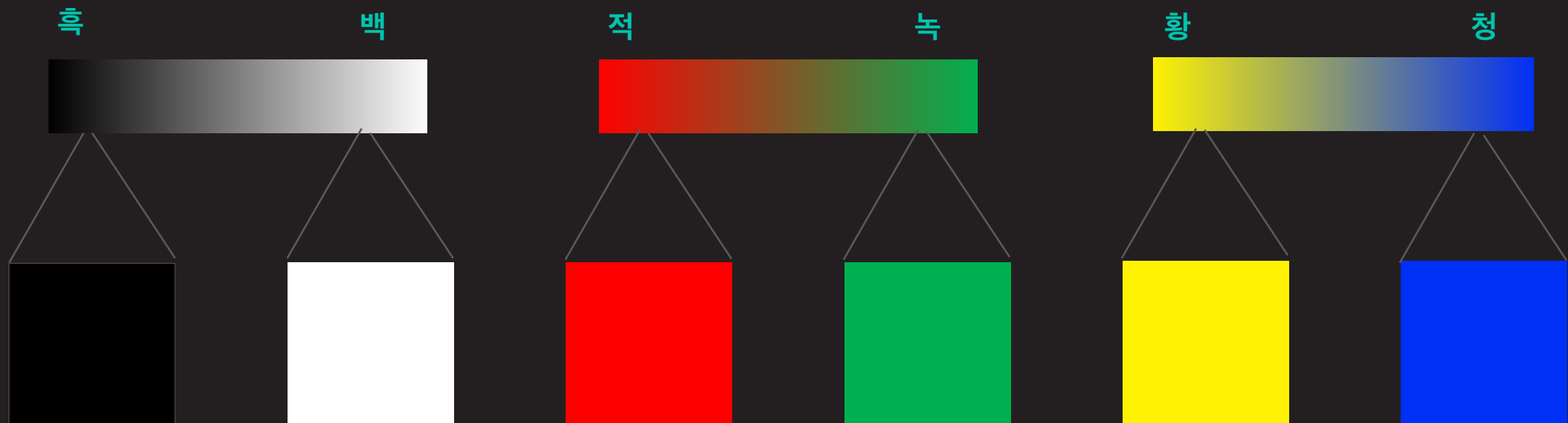


## 5. 우리의 색상 지각은 제한적이다

### 5.6 색상에 대한 가이드라인

#### 2. 쉽게 구분할 수 있는 색상 사용하기

우리 시각이 대응하는 색상 채널

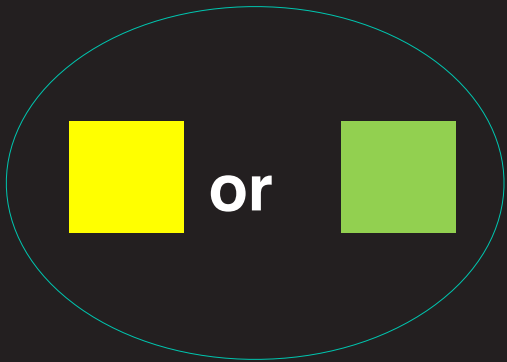
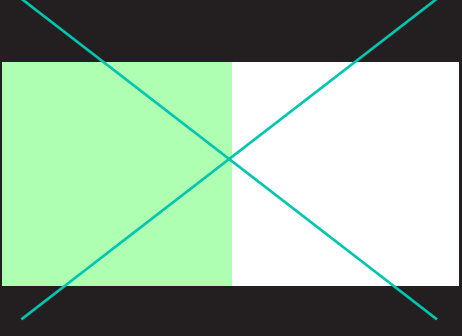
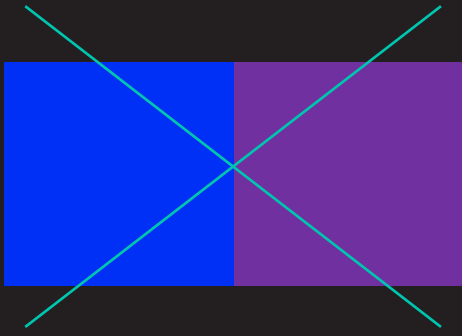
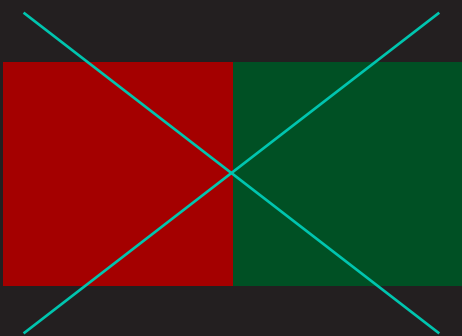
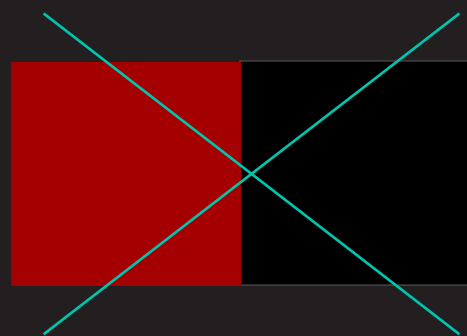


이 채널에서 가장 강한 신호가 만들어지는 색이 가장  
구분이 잘된다.

# 5. 우리의 색상 지각은 제한적이다

## 5.6 색상에 대한 가이드라인

### 3. 색맹을 가진 사람을 생각할 것



## 5. 우리의 색상 지각은 제한적이다

### 5.6 색상에 대한 가이드라인

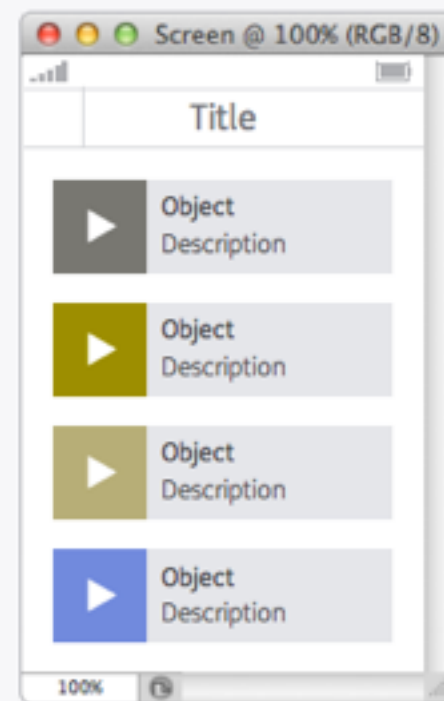
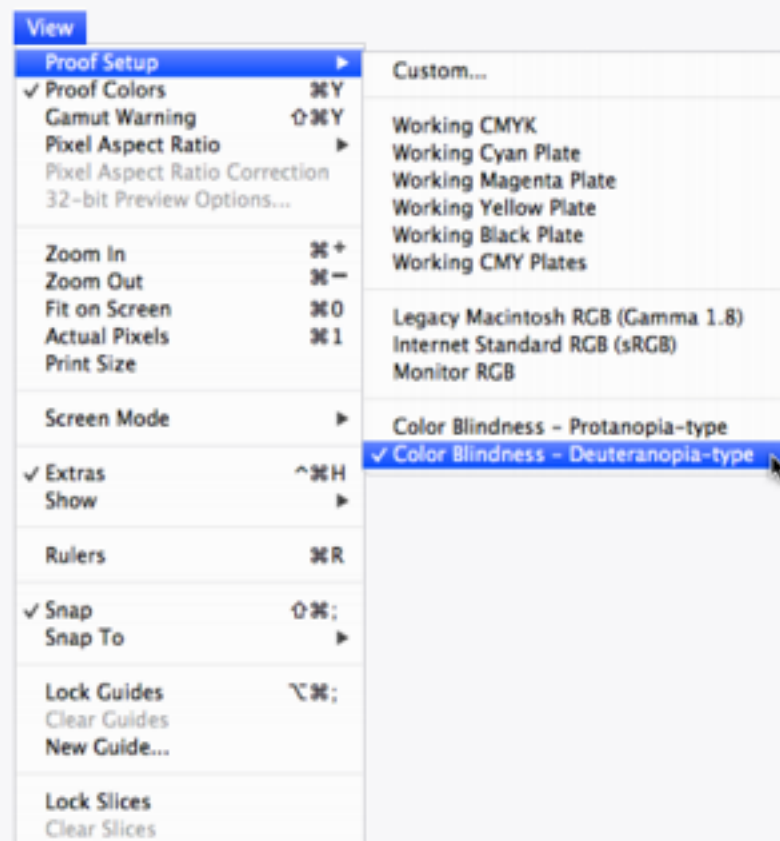
### 3. 색맹을 가진 사람을 생각할 것

Adobe 색맹 미리보기 모드

- View > Proof Setup > Color Blindness

Sim Daltonism 색맹 미리보기 툴

<http://michelf.ca/projects/sim-daltonism/>

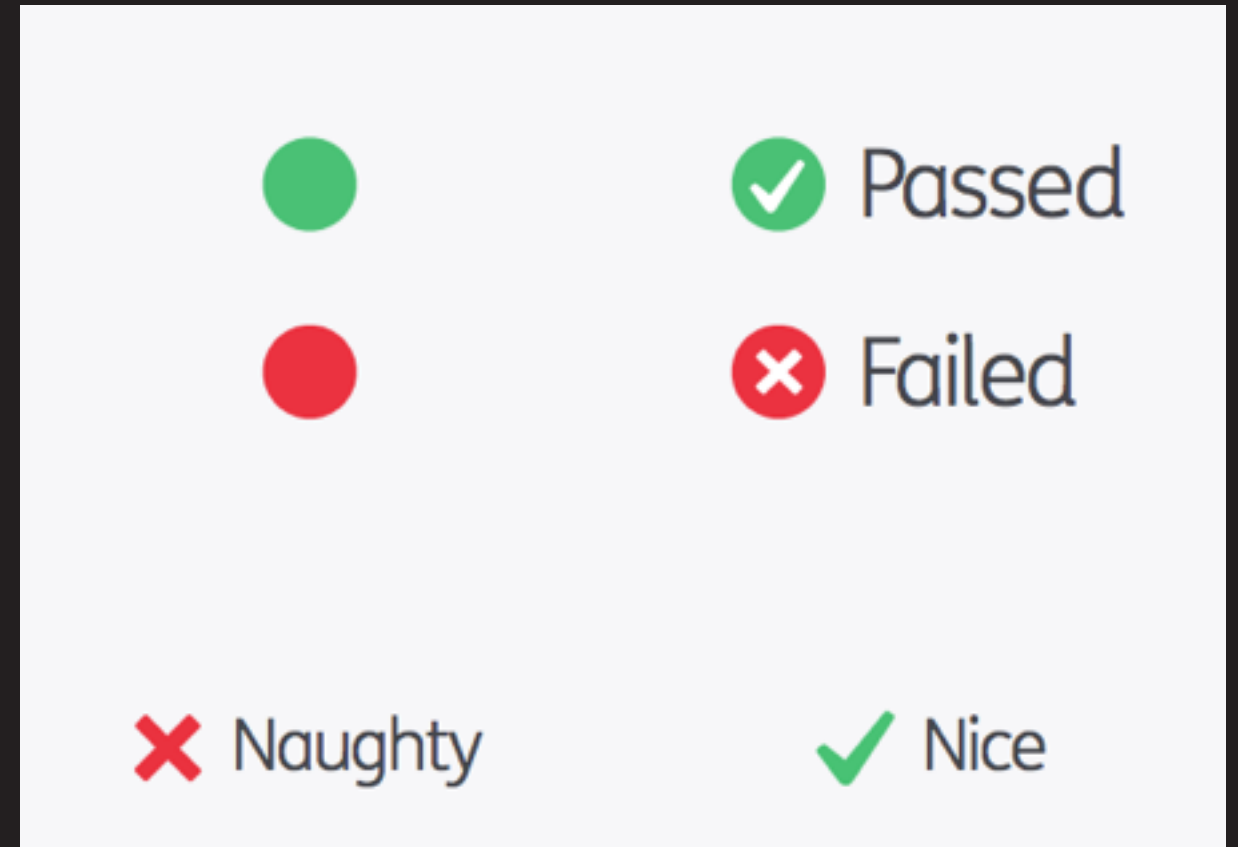
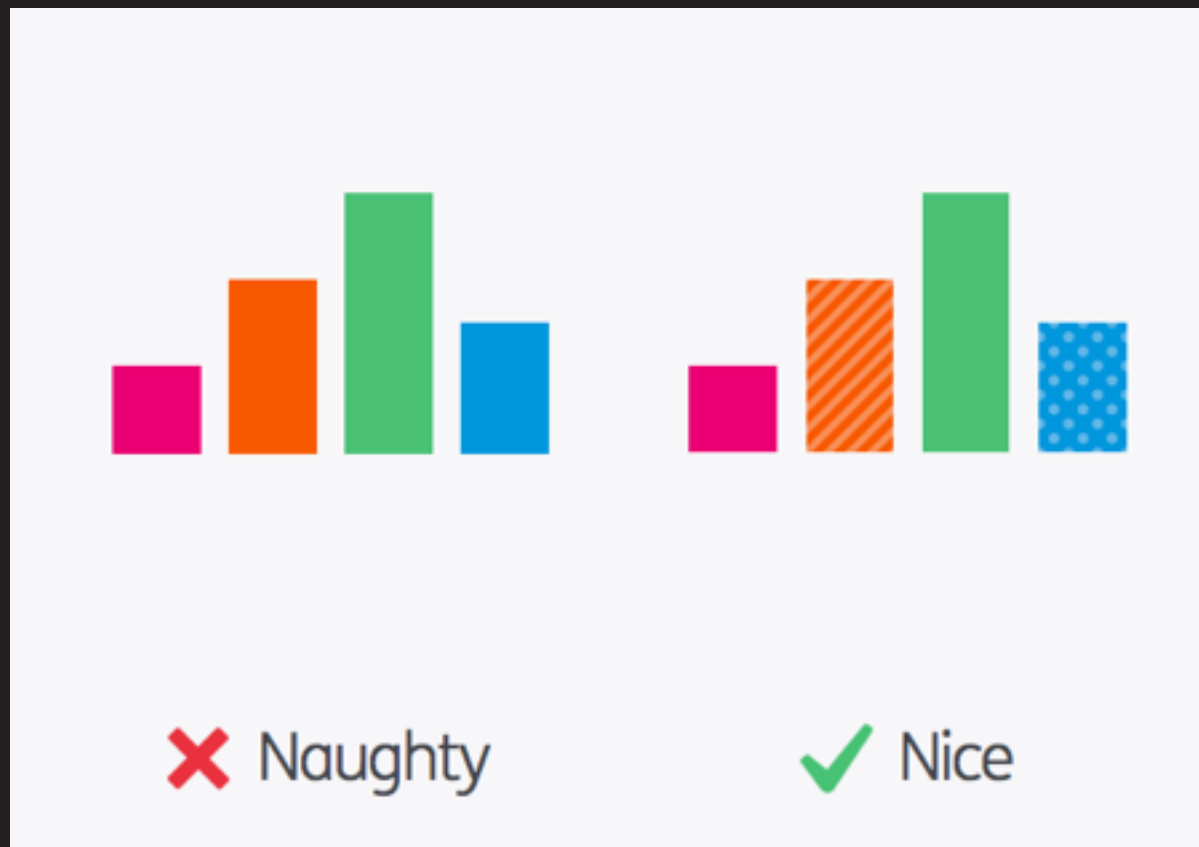


## 5. 우리의 색상 지각은 제한적이다

### 5.6 색상에 대한 가이드라인

### 4. 색상에만 의존하지 말 것

색상이 구분이 안될 때를 대비해서  
색상 이외의 다른 표식도 함께 사용하세요



# 특정한 사람을 위한 디자인

## 선택주의 : 초점의 문제

- 의식적 주의력에는 한계가 있다. (한 가지에 주의의 초점을 두면 다른 것에 대한 주의를 줄게 된다.)
- ex 1) 불이 나면 당황해 당겨야 할 문을 미는 행위.
- ex 2) 토스트기의 빵이 안나온다고 쇠붙이로 꺼내려함.
- 결국 디자인이란 이런 것 까지 신경을 써서 만들어야한다.
- (문을 밀고 당겨서 열수 있게, 토스트기 강제 전원 차단 버튼)



# 수도꼭지 :

## 디자인의 어려움을 보이는 한 사례

- 통제할 요소 : 물의 양, 온도
- 1. 의도
  - => 이 수도꼭지를 어떤 방향으로 돌리면 물의 세기가 달라지겠지?
- 2. 행위의 대응
  - => 이 수도꼭지를 왼쪽/오른쪽으로 돌렸더니 뜨거운 물이 나왔군.
- 3. 평가
  - => 물의 양, 온도가 적절한지 어떻게 알 수 있지?
  - => 수도꼭지가 하나일 경우, 또는 두개일 경우 어떻게 다르지?

# 디자이너가 겪게 되는 두 가지 치명적인 유혹

- 1. 은밀한 기능추가주의
- 2. 잘못된 이미지의 숭배

# 컴퓨터 시스템의 결점

1. 어떻게 잘못될 수 있는가
2. 아직 잘할 수 있는 기회가 있다.
3. 카멜레온 같은 컴퓨터

# 3. 카멜레온 같은 컴퓨터

## 1. 탐색 가능한 시스템 : 실험을 권장하기

- 가시적, 좋은 심성모형 사용하기, 되돌리기 쉬워야함

## 2. 컴퓨터 사용의 두 양식

- 명령어 양식(삼인칭 상호작용) : 실제 일할 사람에게 명령을 내리는 것
- 직접 조작 양식(일인칭 상호작용) : 자신이 직접 조작하는 것

## 3. 미래 컴퓨터는 비가시적이 된다

- 과업중심적 시스템
- 컴퓨터를 사용하고 있다(x), 캘린더를 보고 있다(o)



# 자동화 된 모든 행위에는 학습이 따른다!

우리는 어떤 조건에서 더 잘 학습할 수 있을까?

- 주어진 조작법이 과업 중심적이고 단순하며 일관적인 경우
- 주어진 어휘가 과업 중심적이고 친숙하며 일관적인 경우
- 위험성이 낮은 경우

## 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

어떤 과업을 수행하고자 도구를 사용할 때,

**해야할 일 - > 도구가 제공하는 조작법** (변환필요)

인지적 노력이 필요한 부분

Ex)

**목적: 회사의 조직도를 그려보자!**

그림그리기 소프트웨어를 사용해 회사의 조직도를 그린다: 선과 도형을 다 그려야 함

조직도 작성 전용 소프트웨어: 선과 도형을 다 그릴 필요 없음

## 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

# UX디자이너가 도구를 이해하는데 필요한 프로세스

1. 과업 분석을 수행한다
2. 과업 중심의 개념 모델을 만든다. 이는 주로 객체/행위 분석으로 구성된다.
3. 과업 분석과 개념모델에 기반하여 사용자 인터페이스를 디자인.



# 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

## 1. 과업분석

과업분석은 아래와 같은 것을 분석하는 것이다.

사용자의 목표, 과업의 빈도, 과업의 구성단계, 과업의 구성도, 어떤 사람이 어떤 과업을, 과업을 수행할 때 겪는 문제, 과업 사이의 관련도, 과업을 수행 할 때 쓰는 용어들.... 등등

### 과업분석 예시\_좋은사례



직관적으로 알 수 있는 디자인\_ 최소한의 시간과 경험만으로 자동적 처리모드에 도달 할 수 있는 인터페이스

# 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

## 1. 과업분석

### 과업분석 예시\_실패사례



고객, 상품, 고객의 상품평, 상품평에 대한 댓글 ▶ 적절한 객체 목록  
데이터베이스, 테이블, 영속적인 쿠키 ▶ 적절하지 않은 객체 목록

- 상품객체에 대한행위 \_ 조회하기, 상품평 쓰기
- 상품평 객체에 대한 행위\_상품평 읽기, 댓글달기
- 댓글 객체의 속성\_제목, 작성자 이름, 작성일

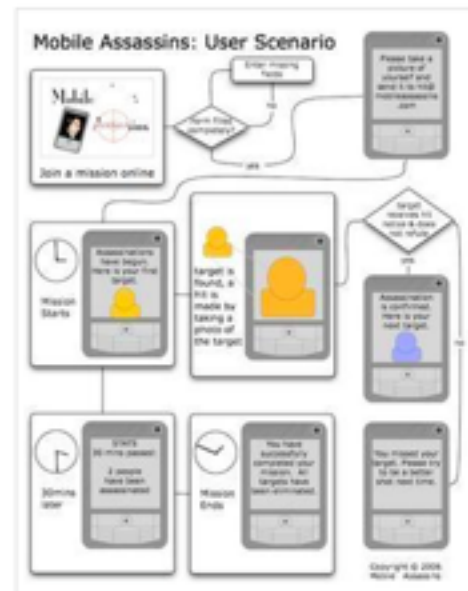
# 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

## 1. 과업분석

### 과업분석 예시

#### ③ 시나리오 분석법(scenario analysis)

시스템을 이용하는 사용자의 구체적 경험을 순차적으로 기술



#### 예시)

##### 다국적 석유회사 로열더치 셸(Royal Dutch Sell)의 에너지 위기 시나리오 분석

- 1968년 당시 유가가 안정되어 있어 그 누구도 유가상승을 예측하지 못했던 시절이었는데 미국의 석유 비축량이 바닥을 보이고 있고, 67년 6월 전쟁 후 석유산유국이 서방세계의 이스라엘 지원에 대한 반발로 정치적 징후를 보고 에너지 시나리오를 작성  
오일 쇼크 사태가 닥쳤을때 이를 사건에 예측하고 철저하게 준비 하였기 때문에, 급격한 환경 변화에 대응할 수 있었으며, 이를 계기로 7위에서 2위로 도약함.

시나리오 분석법 : 시스템을 이용하는 사용자의 구체적 경험을 순차적으로 기술

## 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

### 2. 객체/행위 분석

#### 객체/행위 분석

- 사용자에게 노출할 모든 개념적 객체들, 각 객체에 대해서 사용자들이 수행 할 수 있는 행위
  - 각 객체의 속성들
  - 각 객체간의 관계
- 들이 나열되어 있음.

#### 즉, 사용자에게 노출할 개념들의 선언

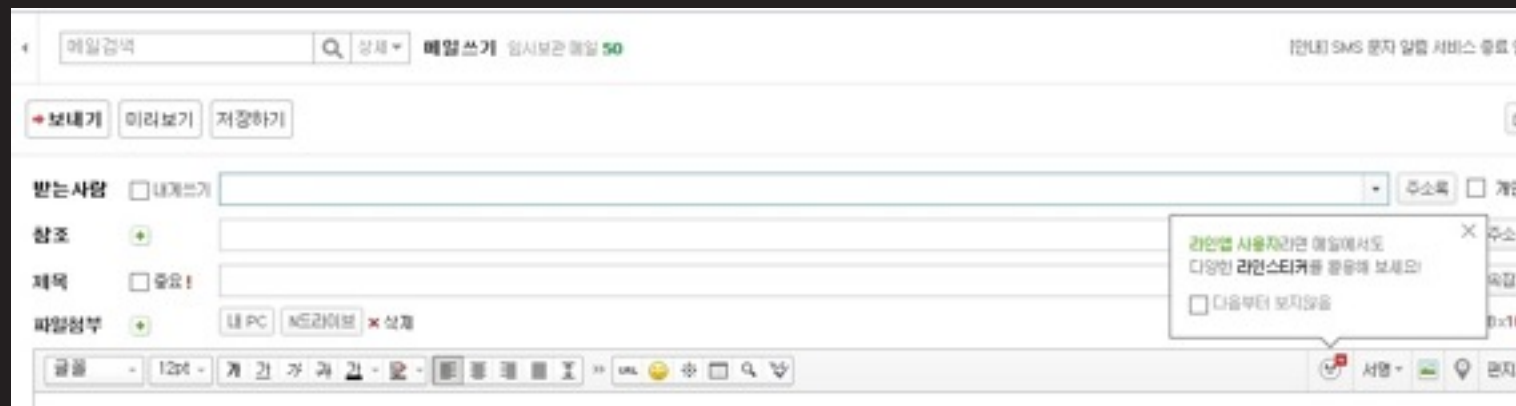
무엇이건 객체/행위 분석에 포함되어 있지 않다면,  
사용자도 알 필요가 없어야 한다.



# 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

## 2. 객체/행위 분석

### [메일을 쓰는 상황]



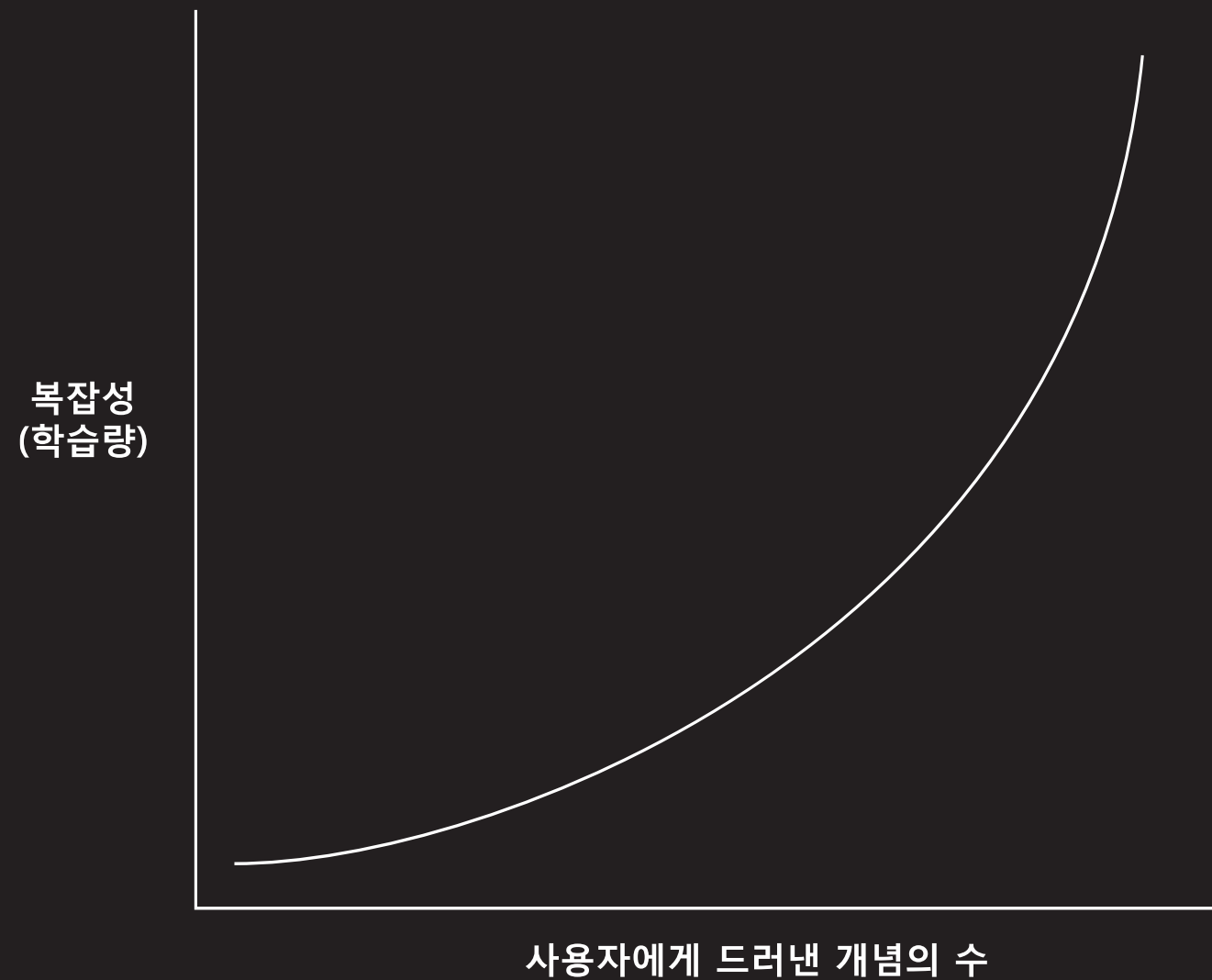
-메일쓰기에 대한행위 \_ 받는 사람쓰기, 제목쓰기, 내용쓰기

-미리 보기에 대한 행위\_ 수정하기, 지금 보내기, 한눈에 보기

1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

3. 가능하면 단순하게

모델이 필요한 기능을 만족하는 한,  
사용자가 익혀야 할 개념이 적으면 적을수록 좋다





## 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

### 3. 가능하면 단순하게

모델이 필요한 기능을 만족하는 한,  
사용자가 익혀야 할 개념이 적으면 적을수록 좋다



사용자의 욕시모를 필요성 때문에 너무 많은 기능을 탑재할 경우 복잡성이 더해지는 것이 아니라 곱해진다.

▶알람시계의 경우 알람의 기능만이 있는 것이 아닌 다른 부가적인 라디오, MP3등 많은 기능들이 더해질 경우 복잡성 증대



## 1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

### 3. 가능하면 단순하게

모델이 필요한 기능성을 만족하는 한,  
사용자가 익혀야 할 개념이 적으면 적을수록 좋다

#### 올해 스마트폰 트렌드는 군살빼기?

트랜스레이터앱·LG뷰톡 등 기능 줄이기 나서

2014.06.21, 토 10:00 입력



기존 탑재 애플리케이션(앱)을 줄이라는 정부 가이드라인도 있지만, 제조사들이 단순함을 스마트폰의 새로운 가치로 여기기 시작했기 때문이다. 올해부터 본격적으로 '군살빼기'가 시작됐다.

21일 업계에 따르면 제조사들이 스마트폰에 일상 생활에서 많이 쓰이는 필수 기능을 강화하고 이용이 낮은 앱이나 기능을 지속적으로 줄이고 있다.

대표적으로 [번역 애플리케이션]은 조만간 완전히 자취를 감추게 될 전망이다.

선탭재 앱이 너무 많아 메모리를 많이 차지 하는 데다, 앱들이 난립해 '공해'에 가깝다는 주장 때문이다. 이에 따라 삼성전자, LG전자, 팬택 등은 최대 60~70여개에 이르던 기본 탑재 앱을 20여개 수준으로 줄이고 나머지는 다운로드받을 수 있게 했다

1. 우리는 주어진 조작법이 과업 중심적이고 단순하며 일관적일 때 더 빠르게 배운다

3. 가능하면 단순하게

모델이 필요한 기능성을 만족하는 한,  
사용자가 익혀야 할 개념이 적으면 적을수록 좋다



사용자의 목적과 과업에 잘 맞기만 한다면 익혀야 할 개념은 적을수록 좋다. 뒤로가기, 취소, 홈으로가기, 지문 잠금장치 등등 익혀야할 개념을 줄임.

# 널리 알려진 인터페이스 디자인 규칙들

## Johnson (2007)

## 원칙 1.

기술이 아니라 사용자와 사용자의 과업에 집중할 것

- 사용자를 이해하라
- 과업을 이해하라
- 소프트웨어가 쓰일 맥락을 고려하라

# 원칙 2.

## 기능 먼저, 표현은 나중

- 개념 모델을 만들 것

# 원칙 3.

## 과업에 대한 사용자의 관점에 맞출 것

- 자연스러움을 추구하라
- 당신의 언어가 아닌 사용자의 언어를 사용하라
- 프로그램의 내부가 드러나지 않도록 하라
- 기능과 복잡성 사이의 적절한 타협점을 찾을 것

# 원칙 4.

## 일반적인 상황에 맞춰 디자인할 것

- 가장 일반적인 작업이 쉽게 수행될 수 있도록 할 것
- 두 종류의 “일반적인” 것이 존재한다:  
많은 사람이 쓰는 것, 자주 쓰는 것
- 특이한 사례보다는 핵심적인 사례에 맞춰  
디자인 할 것

## 원칙 5.

사용자의 과업을 복잡하게 만들지 말 것

- 사용자에게 추가적인 문제를 제시하지 말 것
- 사용자가 제거에 의한 추론을 하도록 하지 말 것



# 원칙 6.

## 학습을 유도할 것

- 안에서-밖으로(inside-out)가 아니라  
밖에서-안으로(outside-in) 생각하기
- 일관성, 일관성, 일관성
- 위험성이 낮은 환경을 제공 할 것

# 원칙 7.

그저 데이터가 아닌 정보를 제공할 것

- 전문가의 도움을 얻어서 정보를 잘 표현할 것
- 화면 영역은 사용자의 것이다
- 디스플레이 관성을 유지할 것

# 원칙 8.

## 반응성 있는 디자인

- 사용자의 명령에 즉시 반응할 것
- 소프트웨어가 다른 일을 수행하느라 바쁜 상황과 그렇지 않은 상황을 사용자가 알 수 있도록 할 것
- 사용자가 기다리고 있는 동안 다른 일을 할 수 있도록 허용할 것
- 움직임은 부드럽고 명확하게 표현할 것
- 오래 걸리는 작업이 있을 때 사용자가 언제든지 취소할 수 있도록 할 것
- 사용자가 작업에 걸리는 시간을 추정할 수 있도록 할 것
- 사용자가 자신의 작업 속도에 맞춰 작업할 수 있도록 할 것

# 원칙 9.

## 사용자를 대상으로 테스트하고 수정하기

- 테스트 결과를 보면 숙련된 디자이너도 놀랄 수 있음
- 테스트를 통해 밝혀진 문제를 고칠 수 있는 시간을  
배정하여 일정을 잡을 것
- 테스트의 두 가지 목적 : 정보성, 사회성
- 언제 어떤 목적에서든지, 적절한 테스트는 존재한다

# 어려운 과제를 쉽게 만드는 일곱 가지 원칙

1. 머리 속의 지식과 세상 속의 지식을 모두 이용하라.
2. 과제의 구조를 단순하게 하라.
3. 일이 가시적이게 만들어라. 실행의 간격과 평가의 간격을 좁혀라.
4. 대응관계가 올바르게 만들어라.
5. 자연스러운 제약 및 인공적 제약의 위력을 활용하라.
6. 만일의 오류에 대비한 디자인을 하라.
7. 이 모든 것이 잘 되지 않으면, 표준화하라.