L +

SHARE. SPECIAL.

AIRFORCE INFORMATIONAL SYSTEMS MANAGEMENT GROUP
SEMINAR MAGAZINE
ISSUE VOL.1

#1234 SEMINAR MAGAZINE :)

Photoshop+illustrator Basic Seminar
Typography for Non Design Major
Good & Modern Programming
Foreign Company Job Seminar
for Computer Science Major

월간 <공군> 11월호 부록

SHARE.
SPECIAL.

너의 재능, 샘이나는 공군 체계단 장병들의 동아리 활동을 촉진하고,
병사들의 자발적인 자기계발 문화 형성을 위해 만들어진 프로젝트입니다.

정보체계관리단 병사들이 세미나 형식으로 다양한 주제로 자신들의 재능을 공유하고,
더불어 병사 간 소통을 통해 건강한 병영문화를 만들고자 합니다.

체계단 병사라면 누구나 세미나의 주인공이 될 수 있습니다.
자신들의 평소 관심사를 다른 사람들과 서로 공유하면서 함께 성장하는 기회가 될 것입니다.

샘이나는 격식있고 무거운 자리가 아닌 자유로운 분위기를 추구합니다.
하지만 우리가 함께하는 시간은 결코 가볍지 않습니다.
앞으로의 진로, 나아가 인생에 있어 도움이 될 수 있는 소중한 시간이 될 것입니다.

너의 재능,
샘이나

EDITOR'S LETTER

그렇게 뜨겁던 여름이 이제는 아주 옛날이 되어버린 것처럼, 달라진 공기가 계절의 변화를 느끼게 해줍니다. 간만의 연휴였던 어제, 가을 근무를 위해 동약복을 꺼내어 계급장을 바꿔달고 세탁기를 돌렸습니다. 옷은 깨끗해졌지만 주름이 많이져서 마르고 나면 다림질이 필요해보였습니다. 근데 그 옷을 보면서 왠지 모르게 지난 한 달하고도 보름동안 매거진으로 끙끙 앓던 제 모습이 겹쳐보이더군요.

이 매거진을 처음으로 제작하기로 결정했을 때 정말 많은 고민이 있었습니다. 거의 맨땅에 헤딩하는 수준으로 잡지 구성부터 디자인, 콘텐츠까지 많은 참고자료들을 찾아보며 공부하고 만들어보았습니다. 몇 번이고 다시 엎고 제작하면서 포기하고 싶었던 마음이 들었던 것도 사실입니다. 하지만 이렇게 마침내 완성하고 보니 역시 하길 잘했다는 생각이 듭니다.

아직은 쌀쌀했던 3월의 봄부터 시작해서 여름까지 진행되었던 세미나들을 이 매거진에 모아보았습니다. (지금 이 매거진이 제작되는 와중에도 이미 두 주제의 세미나가 더 진행되었습니다.) 이렇게 모아놓고 보니 그래도 '군생활에 뭐 하나는 남겼구나'라는 생각이 듭니다. 어쩌면 흘러가버렸을 우리들의 말 한마디, 생각 한마디가 이 종이에 녹아있습니다. 그리고 이 세미나를 통해 함께 나누었던 여러분들의 재능을 보다 보기 좋게 정리해보려고 노력을 많이 했습니다. 부족한 점이 물론 많겠지만, 서투른 점도 분명 있겠지만, 앞으로의 '샘이나'와 매거진이 더 발전할 수 있도록 관심과 격려로 봐주시면 감사하겠습니다.

이 매거진의 다림질은 여러분들이 함께해야 완성될 수 있습니다. 따순 열기가 구김을 다릴 수 있듯, 여러분들의 따끔한 충고나 따뜻한 응원이 이 매거진을 더 멋있게 만들 수 있습니다. 저희는 언제나 가판대에서 여러분들을 기다리고 있겠습니다. 여러분들의 손에서, 책장에서 멋있게 다려질 날을 상상하면서요.

편집장 이 창 훈

너와 나를 잇다

돌이켜보면 우리는 수많은 '나'와 '너'로 이루어져있다. 결코 오롯이 혼자서 '나'이기는 쉽지 않다. 가족, 친구, 선생님 등 주변의 많은 사람들을 거쳐 지금의 내가 있다. 하지만 어쩐 일인지 나이가 들고 어른이 되면 이러한 연결고리들은 하나 둘씩 나에게서 떨어져 나간다. 어쩌면 스스로 '다 컸다'고 단정지어버린게 아닐까. 아직도 세상은 넓고 이렇게 너와 내가 있는데.

'너의 재능, 샘이나'는 너와 나를 잇고, 함께 재능과 관심사를 공유하는 연결고리가 되고 싶습니다. 가진 것도, 배울 것도 많은 우리. 조금만 더 마음을 열어보세요. 주변에 수많은 멋진 만남들이 기다리고 있습니다. 흘러가는 오늘이, 다가오는 내일이 당신에게 가슴 두근거리는 시간들이 되기를 바라며 이번 매거진을 준비했습니다. 자, 준비되셨나요?

CONTENTS

5p 너와 나를 잇다

8p 공유의 가치

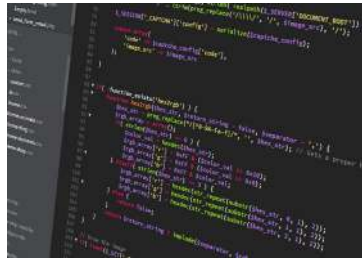
11p #1. Ph'il (포토샵/일러스트 기초)

39p #2. 글자 다루기 세미나

53p #3. 굿&모던 프로그래밍

64p 첫과 끝의 이야기

69p #4. 외국계 취업 세미나



AIRFORCE INFORMATIONAL SYSTEMS MANAGEMENT GROUP *SEMINAR* MAGAZINE ISSUE VOL. 1

EDITORIAL

Editor in Chief 이창훈 LEE CHANGHOON
Editor 최문규 CHOI MOONGYU
서희강 SUH HEEKANG
주호영 JOO HOYOUNG
양민호 RYANG MINHO
Asst. Editor 곽신우 KWAK SHINWOO

ART

Designer 이창훈 LEE CHANGHOON

PHOTOGRAPHY

PEXELS.COM

MANAGEMENT&ADMINISTRATION

이창훈 LEE CHANGHOON
임정근 LIM JEONGKEUN

PUBLISHER

국군인쇄창 (F18100418)
(910-4504, 042-553-4504)

TREND STORY

공유의 가치

『샘이나』의 핵심은 자신의 재능을 다른 사람들과 '공유' 하는 데 있습니다.
혼자 살아가기 너무나 복잡하고 힘든 현대사회에서, 공유는 큰 힘이 됩니다.

"점호 10분 전"

오늘도 하루가 시작됩니다. 점호를 마치고 분주히 출근 준비를 합니다. 이제는 제법 쌀쌀해진 공기가 몸을 감쌉니다. 출근을 하면서 여러분들은 어떤 생각을 하나요? 혹시 이미 평범해져버린 자신의 군생활에 지루해 하고 있는 자신을 발견하지는 않으셨는지요. 우리가 군에 있든 있지 않든, 우리의 삶은 계속 현재 진행형입니다. 매 순간 순간의 생각과 행동이 쌓여서 여러분들의 삶을 만듭니다. 많은 제약과 어쩔 수 없는 상황이 때로는 몸과 마음을 괴롭히죠. 하지만 그 파도에 휩쓸리기만 한다면 결국은 제풀에 죽어 자꾸만 움츠려들거에요.

우리는 모두 특별한 무언가를 지니고 있습니다. 없다 없다 덮어놓고 부정만 하지 말고. 분명 당신은 당신 자체로 빛나고 있으니까요. 내가 모르고 있었던 나의 특별한 재능, 사소하지만 소중한 나의 보석을 생각해봐요. 그리고 그 빛나는 보석을 지금 우리와 함께 나눠주세요.

각자가 갖고 있는 보물이 모이면 우리는 더 다채롭고 멋있는 사람으로 성장할 수 있습니다.
샘이나는 여러분들의 소중한 재능을 함께 나누고 토론할 수 있는, 언제나 열려 있는 방이 되고 싶습니다.
언제든지 오세요. 꼭 가벼운 발걸음으로 찾아오세요.
하지만 우리가 함께 하는 시간은 결코 가볍지 않을 것입니다.
끝으로, 아직도 망설이고 있을 누군가에게 드리는 글입니다.

오늘이 끝이라 해도 우리가 할 수 있고, 하고 싶은 일이란 게 특별한 건 아닌 것 같아요.
그저 평소에도 얼마든지 할 수 있었던 건데 용기가 부족해서 못하거나 망설이고 게을러서 놓쳤던 것들이지 싶어요, 기껏 해 봐야.

지금 시작해요 우리, 함께, 여기서.

#1
포토샵&일러스트 기초
Ph'il 세미나

2018년 3월 7일 - 4월 11일
최문규

Photoshop & illustrator
Ph'il Seminar

MARCH 7 - APRIL 11, 2018
CHOI MOONGYU

한 번쯤은 다뤄보고 싶다 생각했던 디자인 툴.
제대로 배워두면 필요할 때 써먹을 수 있는
6주 커리큘럼의 포토샵과 일러스트 기초 세미나입니다.

Seminar Instructor & Editor	최문규 CHOI MOONGYU
Management & Administration	임정근 LIM JEONGKEUN
Seminar Assist.	장우진 JANG WOOJIN
Photography	김경문 KIM KYOUNGMOON
	강세호 KANG SEHO
	이선용 LEE SEONYONG
	배명환 BAE MYEONGHWAN

한 번쯤은 다뤄보고 싶다 생각했던 디자인툴, 비전공자를 위한 디자인 전공자의 Ps+Ai Seminar

포토샵은 다양한 것을 가능하게 합니다. 인물사진 색 보정, 편집 등 우리가 알고 있는 '뽀샵'부터 시작해서 행사용 포스터 및 배너 디자인, 실사진 및 캐릭터 일러스트, 간단한 애니메이션, 브랜드/탬플릿/명함 디자인 등 다 설명하기 힘들 정도로 많은 일을 할 수 있습니다. 사실 포토샵이나 일러스트를 배우는데 이런 지루한 기초상식은 별로 필요 없습니다. 그냥 되는대로 하다 보면 어느새 이론은 필요없고 갖고 놀다 보면 실력이 늘어나는 걸 깨닫게 됩니다. 이 수업은 여러분이 적어도 필요한 때에 다른 사람의 도움없이도 원하는 기능을 쓸 수 있도록 함에 목적을 두고 있습니다. 내가 원하는 작업을 스스로 할 수 있다는 것, 이것이 우리 세미나가 줄 수 있는 가장 큰 선물입니다.

Our Instructor

최문규 CHOI MOONGYU

- School of Visual Arts(SVA) 그래픽디자인학과 학사 졸업
- Rippleshot 여름 단기 인턴
- Taesan 반도체부품 · 자동차부품 · 크라이오펌프 가공회사 웹사이트 제작

Our Seminar

Photoshop 4주 / Illustrator 2주

1주차. 색의 3원색 기반 (14p)
포토샵의 기초 (16p)

2주차. 사진 편집 및 색 보정 (19p)

3주차. 필터와 스타일 (24p)

4주차. 레이어 스타일 (29p)

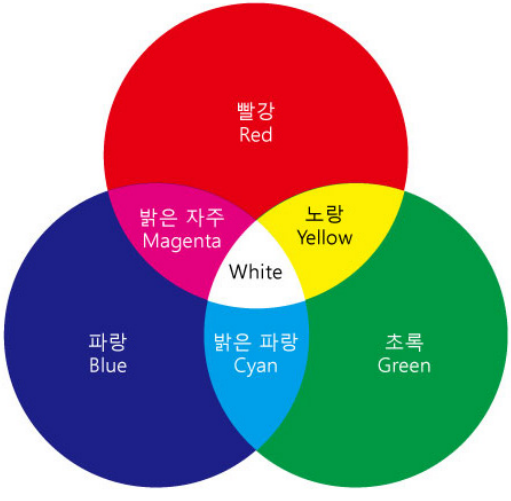
5주차. 일러스트레이터의 상식 및 기초 (30p)

6주차. 단축키와 패스파인더 (34p)

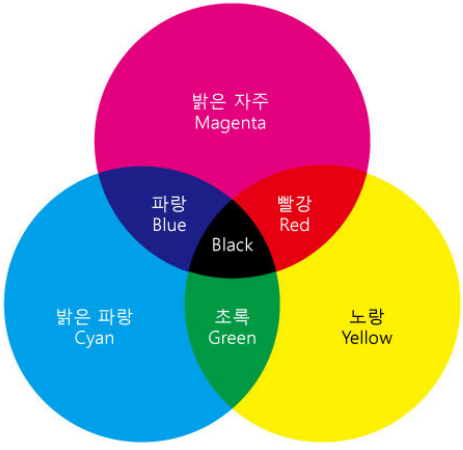
Ph’il + 1

ABOUT COLOR

포토샵은 기본적으로 RGB 형식을 사용합니다.
RGB란 빛의 3원색인 Red, Green, Blue 3색을 조합하여 색을 만들어내는 방식입니다. RGB형식은 주로 프리젠테이션 환경에서 채용하는데, 이는 모니터와 스크린상에서 선명한 3원색을 표현할 수 있기 때문입니다. RGB 모든 색이 0이 되면 검은색이 되고, 모든 색이 255가 되면 흰색이 됩니다. (RGB의 값은 0부터 255까지 있습니다.)



빛의 삼원색(가산혼합/RGB)



색의 삼원색(감산혼합/CMYK)

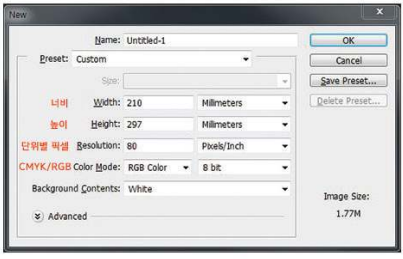
빛의 3원색인 RGB가 있다면 색의 3원색도 존재하겠죠? 여러분들도 아시는 빨강, 파랑, 노랑을 사용하는 CMYK 형식입니다. CMYK란 색의 원색인 Cyan, Magenta, Yellow에 K(채도; 탁한 정도)를 조합하여 색을 만들어내는 방식입니다. CMYK는 포토샵에서도 설정을 통해 사용할 수 있지만, 주로 일러스트레이터에서 사용하는 방식이며 출력 환경에서 사용합니다. 모니터는 빛을 조합하는 방식이므로 RGB가 가능하지만, 프린트된 출력물은 빛을 낼 수 없으니 CMYK를 사용하는 것이죠. 따라서 모든 프린터는 CMYK로만 출력이 가능합니다. RGB와 달리 CMYK는 모든 색이 0%가 되면 흰색이 되고, 모든 색이 100%가 되면 검은색이 됩니다.

RGB/ CMYK

Part 1.
포토샵의 기초

프로그램은 일단 깔았는데 뭐가 뭔지 잘 모르시겠다고요?
캔버스 열기, 레이어, 브러쉬 등 포토샵에서 가장 기본이 되는 기능들을 살펴봅시다.

1) 새 파일 열기

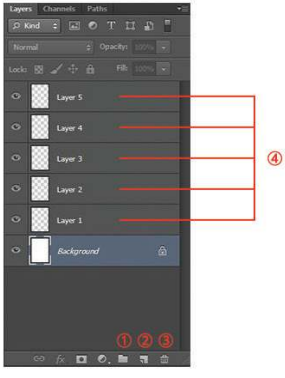


(210mmx297mm은 A4용지 표준 크기)

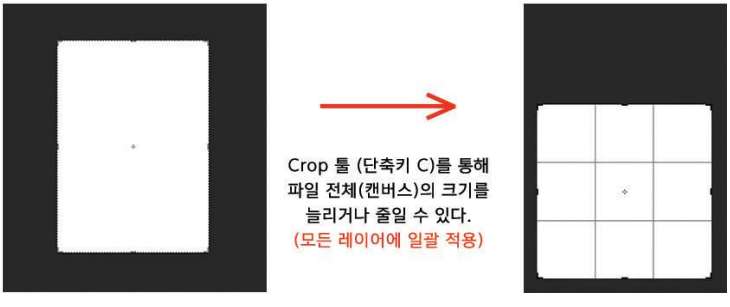
2) 레이어

① 새 그룹 생성
② 새 레이어 생성
③ 레이어 삭제
④ 생성된 레이어들

레이어란 일종의 투명종이와 같다.
가장 위쪽부터 아래로 보여지는 방식이며
각 레이어는 제각각의 그림, 사진, 필터 등
여러가지 효과를 넣을 수 있다.



3) Crop 툴



Crop 툴 (단축키 C)를 통해
파일 전체(캔버스)의 크기를
늘리거나 줄일 수 있다.
(모든 레이어에 일괄 적용)

4) 브러쉬 툴

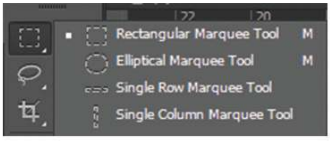
① 브러쉬 크기
② 브러쉬 경도 (높을수록 강함)

5) Blur, Sharpen, smudge 툴

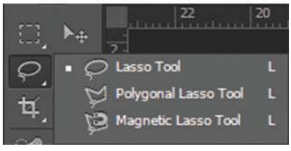
① 적용한 부분을 흐릿하게 만듦
② 적용한 부분을 선명하게 만듦
③ 적용한 부분을 밀어냄

사용방법은 브러쉬와 동일하다.

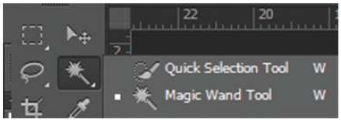
6) Marquee, Lasso, Magic Wand 툴



Marquee: 공간 지정 툴 (사각형, 원 등 특정 모양)
특정 공간만 이동/편집/복사/삭제하고 싶을 때 사용



Lasso: 공간 지정 툴 (자유 모양)
Marquee와 동일하되, 수정하고 싶은 공간을
자유자재로 지정

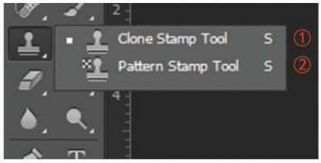


Magic Wand: 공간 지정 툴 (자동 지정)
Quick Selection: 브러쉬 방식으로 자동 지정
Marquee와 동일하되, 수정하고 싶은 공간을
자동으로 지정 (주로 색과 톤으로 구분)
Tolerance값을 조정해 범위를 좁히거나 넓힐 수 있다

Part 2.
사진 편집 및 색 보정

기초 기능을 둘러봤다면 이제 포토샵의 꽃, 사진 편집을 해봅시다.
스탬프, 퀵마스크, 색조/채도/명암 조절을 통해 밋밋한 사진을 맛깔나게 바꿀 수 있습니다.

1) Stamp와 힐링 툴



① Clone Stamp : 지정한 크기와 범위의 내용을 현 커서 위치에 복사
② Pattern Stamp : 특정한 질감 및 패턴을 현 커서 위치에 적용

Alt를 눌러 복사할 위치를 지정 → 왼쪽 클릭으로 적용

사진 협조해주신 배명환 상병에게 감사드립니다.

Spot Healing : 지정 복구
지정한 범위의 색을 주변의 색과 섞는다.

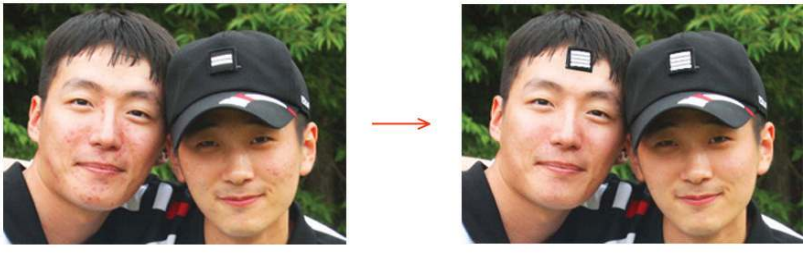


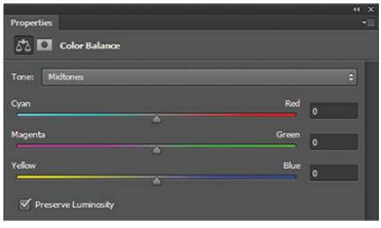
사진 협조해주신 김경문, 강세호 병장에게 감사드립니다.

2) 퀵마스크

Quick mask mode: 퀵마스크
선택지정(marquee, lasso, magic wand 등)된 부분을
색깔로 표시해 주는 모드

이 모드로 진입한 후 브러쉬툴 혹은 지우개툴로
선택영역을 자유자재로 변경할 수 있다.
매직완드나 marquee로 잡을 수 없는 세세한 부분들까지
잡아주기 위해 사용한다.

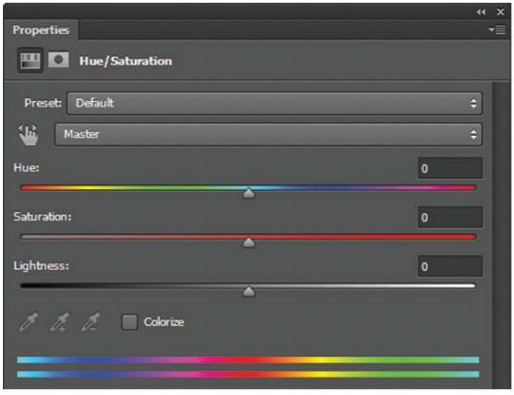
3) Color Balance



Color Balance : 색 균형 조정
선택한 레이어 또는 이미지의 전체적인 색의 톤을 조절할 수 있다.
균형은 RGB와 CMY 기준. (이는 출력시의 CMYK와 크게 연관이 없다)
Shadow, Midtone, Highlight 의 3가지 레벨로 톤을 조절할 수 있으며
보통 자연스럽고 현실적인 색 조절을 위해서는 이 3가지를 모두 조절해야 한다.

조절시 주의할 점:
이미지를 붉게 만들기 위해 Red로 이동한다기보다,
이미지의 푸른 기운을 없애기 위해 Cyan에서 멀어진다는 느낌으로 사용해야 옳다.

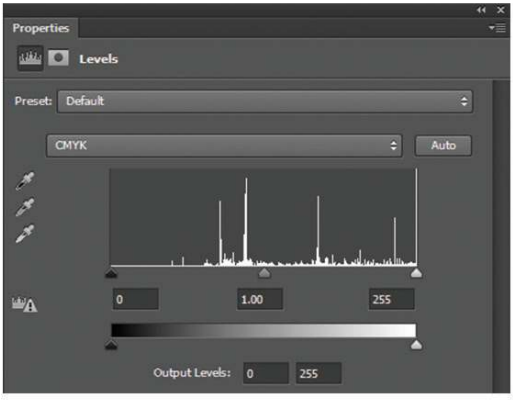
4) Hue / Saturation



Hue / Saturation : 색조 및 채도 조절
선택한 레이어 또는 이미지의 전체적인 색조를
조절할 수 있다.
색조를 변경할 때는 Hue를 사용하고
채도를 변경할 때는 Saturation을 사용하며
빛의 세기를 조절할 때는 Lightness를 사용한다.
(Lightness는 사용 빈도가 적음)

조절시 주의할 점:
Hue / Saturation으로는 원하는 색조가
나오지 않을 가능성이 높기 때문에
원하는 색이 나오지 않을 경우
Color Balance로 직접 조절하는게 좋다.

5) Level

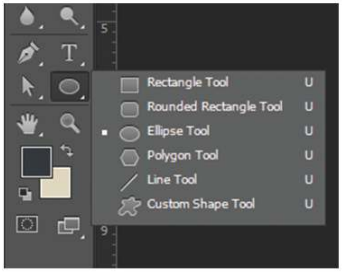


Levels : 명암 조정
선택한 레이어 또는 이미지의 전체적인 명암을
조절할 수 있다.
가장 밝은 부분과 가장 어두운 부분을 기준으로
명암의 분포를 그래프로 표시한 것이며
가장 어두운 0부터 가장 밝은 255값 사이의
톤을 조절함으로서 이미지/레이어의 명암을
조절할 수 있다.
보통 흐릿하게 찍힌 사진을 보정하는 데 사용.

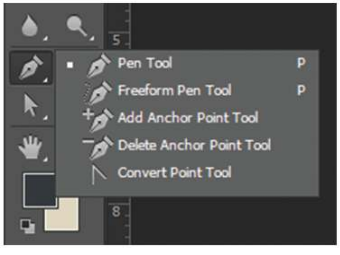
조절시 주의할 점:
과도한 사용은 눈의 피로를 유발할 수 있으니 주의. (예: 태양권)

사진 협조해주신 이선용 병장에게 감사드립니다.

6) 도형과 펜툴



① 사각형
② 둥근 모서리의 사각형 (모서리 지름 조절)
③ 원형
④ 다각형 (각 수 조절)
⑤ 선
⑥ 자유 도형



① 기본 펜툴
② 자유모양 펜툴
③ 고정 점 추가
④ 고정 점 삭제
⑤ 고정 점 속성 변환

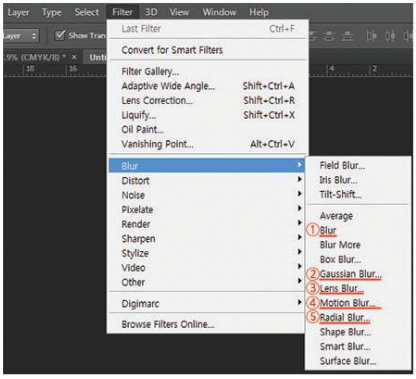
펜툴은 직선과 곡선을 자유자재로 정할 수 있으며,
처음에 생성했던 점으로 돌아가면 모양이 완성된다.
이미 완성된 모양도 고정 점 추가나 삭제를 통해 변형할 수 있다.
③과 ④를 굳이 사용하지 않더라도 고정 점에 커서를 올리면 자동으로 변환된다.
⑤는 고정 점의 각선이나 곡선을 반대로 바꿀 수 있다. (각선->곡선, 곡선->각선)

Part 3.
필터와 스타일

기본적인 편집은 이제 식상하다! 그렇다면 필터 옵션은 어떠세요?
다이내믹한 효과를 넣어주는 필터를 통해 사진, 글씨 등을 멋있게 만들 수 있습니다.

1) 필터 옵션

a) Blur 효과

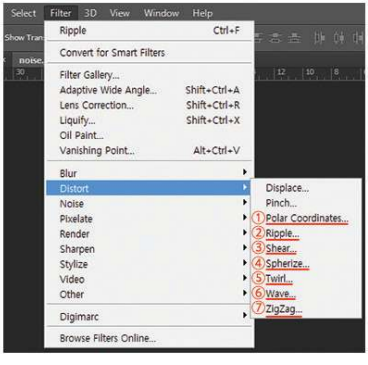


전체 / 선택 영역을 흐릿하게 만드는 필터

① Blur : 흐릿하게 만듬 (약함, 조절값 없음)
② Gaussian Blur : 흐릿하게 만듬 (조절값 있음)
③ Lens Blur : 카메라 렌즈와 같은 흐림 효과
④ Motion Blur : 속도감 있는 잔상 흐림 효과
⑤ Radial Blur : 원형 잔상 흐림 효과

① ② ③ ④ ⑤

b) Distort 효과

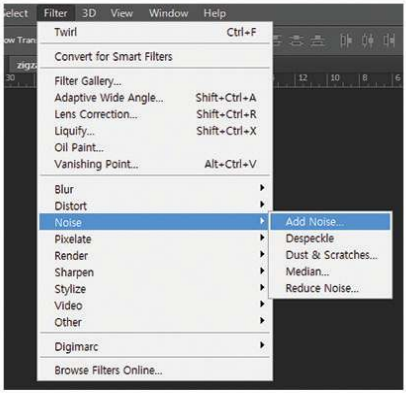


전체 / 선택 영역의 모양을 변경하는 필터

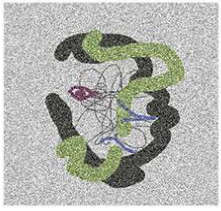
① Polar ~~ : 특정 부분을 기점으로 구(球)형으로 변경
② Ripple : 물결치는 효과 부여
③ Shear : 지정 그래프 모양으로 변경
④ Spherize : 원형으로 부풀리는 효과 부여
⑤ Twirl : 중앙으로 꼬여드는 모양으로 변경
⑥ Wave : 파도 효과 부여
⑦ ZigZag : 원형 지그재그, 파문 효과 부여

① ② ③ ④ ⑤ ⑥ ⑦

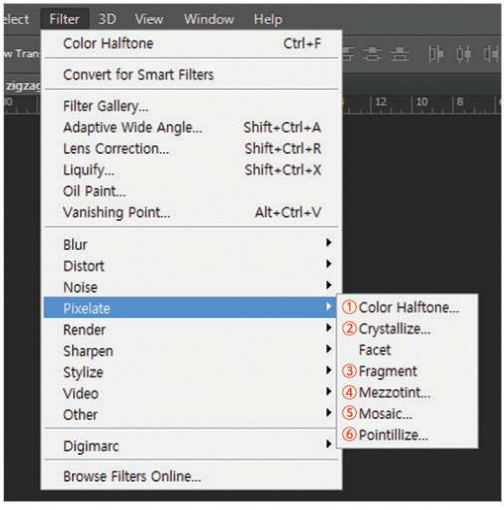
c) Noise 효과



노이즈 효과를 추가



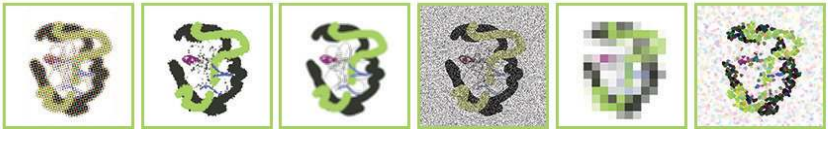
d) Pixelate 효과



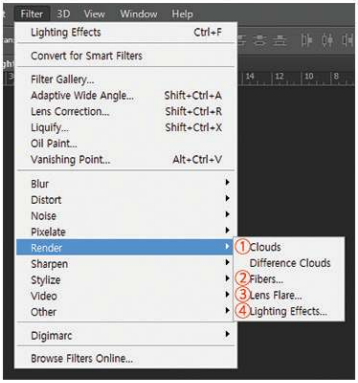
모자이크 효과 부가

① Color Halftone : 점 모양으로 사라지게 만듬
② Crystallize : 크리스탈 모양으로 모자이크
③ Fragment : 투명도 조정 반복으로 잔상 효과
④ Mezzotint : 노이즈와 유사한 효과
⑤ Mosaic : 모자이크
⑥ Pointillize : 점묘화처럼 만듬

① ② ③ ④ ⑤ ⑥

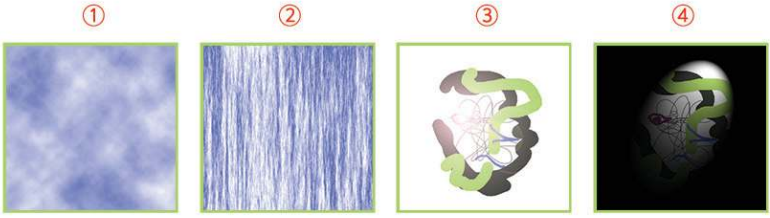


e) Render

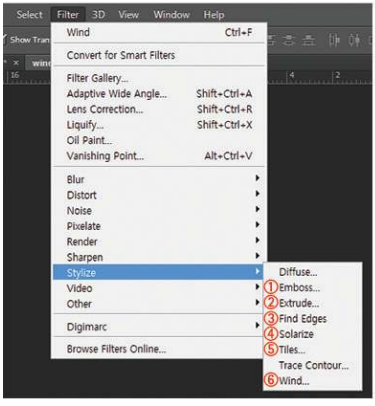


다이내믹함을 더하기 위한 렌더링 효과

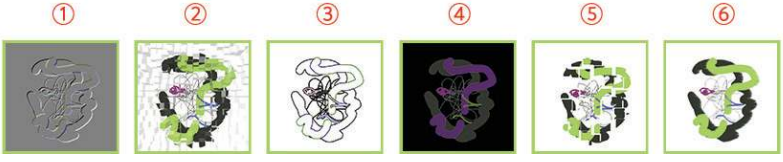
① Clouds : 구름 효과 추가
② Fibers : 섬유 효과 추가
③ Lens Flare : 역광 효과 추가
④ Lighting Effects : 조명 효과 추가



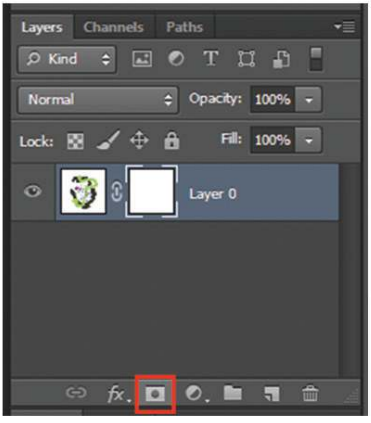
f) Stylize



① Emboss : 음각 및 양각 효과
② Extrude : 튀어나오는 효과
③ Find Edges : 테두리선 이미지로 변경
④ Solarize : 화면 반전 효과
⑤ Tiles : 타일 효과
⑥ Wind : 바람 잔상 효과



2) 마스크



레이어 위에 덧씌우는 편집용 공간을 생성한다.

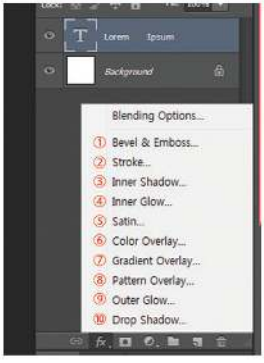
fade in, fade out 효과 및 특정 부분 숨김
원본 그림의 손상 없이 작업 가능

Gradient 툴을 이용해 자연스러운 fade out 효과

Brush 툴을 이용해 특정 공간만 해당 레이어에서 숨김
하위 레이어가 붉은색일 경우, 하위 내용을 표시하기 때문에 붉은색이 표시됨

Part 4.
레이어 스타일

1) 각종 효과들



특정 레이어에 효과들을 맛깔나게 적용하는 툴

① Bevel & Emboss : 3d 엠보싱 효과 적용
② Stroke : 오브젝트에 테두리선 적용
③ Inner Shadow : 오브젝트 안쪽으로 그림자 적용
④ Inner Glow : 오브젝트 안쪽으로 빛 효과 적용
⑤ Satin : 새틴 효과 적용
⑥ Color Overlay : 오브젝트에 색 적용
⑦ Gradient Overlay : 오브젝트에 색(그라데이션) 적용
⑧ Pattern Overlay : 오브젝트에 질감 적용
⑨ Outer Glow : 오브젝트 바깥쪽으로 빛 효과 적용
⑩ Drop Shadow : 오브젝트 바깥쪽으로 그림자 적용

원본

Bevel & Emboss를 적용한 사진

각 메뉴마다 세부적인 설정은 다르지만, 보통 값을 조절하는 방식은 비슷하다.

① 해당 효과를 얼마나 세게 적용시키는가
② 적용시킬 방향을 어디로 하는가
③ 적용 범위와 크기는 어느 정도인가
④ 빛과 그림자의 세기와 방향을 어떻게 하는가

정도의 공통사항이 있으며, 각 효과별 세부설정을 해 보면 어떻게 다른지 알 수 있다.

Part 5.
일러스트레이터의 상식 및 기초

포토샵은 대충 알겠는데 일러스트레이터는 대체 뭔가요? 비슷한 거 아닌가요?
비슷하게 만들거면 일러라는 프로그램을 만들 이유가 없죠. 우리 함께 알아봅시다.

1) 일러스트레이터란?

일러스트레이터, 줄여서 일러스트, 더 줄여서 일러!
사진 편집 및 효과 적용하는 것이 메인인 포토샵과 달리, 일러의 역할은 단순하다.

오브젝트를 만들어라!

이미 있는 사진이나 도형, 그림을 이용하는게 아니라
직접 그 도형 및 그림을 제작하는 쪽에 가까운 것이 일러이다.

Q: 포토샵도 도형 만들 수 있었잖아요?
A: 그렇습니다. 하지만 편의성 및 자율성 면에선 일러가 더 좋습니다.

Q: 그럼 일러 장점이 그게 끝인가요?
A: 그럴 리가요!

포토샵은 기본적으로 모든 작업을 픽셀(pixel)화하고 저장하는 반면,
일러스트레이터는 기본적으로 벡터(vector)로 작업한다.
벡터란, 픽셀단위 하나하나를 편집하는 것이 아니라
한 점에서 다른 점까지의 지정범위를 편집하는 방식이다.
그렇기 때문에 일러는 안 깨진다!

똑같이 작은 크기를 가진 도형의 크기를 늘렸을 때 이미지가 계단식으로 깨지는 포토샵과 달리
일러스트레이터는 깔끔한 외관을 유지할 수 있다는 것이다.

그렇기 때문에 일러스트레이터는 보통 필요한 이미지를 처음부터 만들어 써야 할 때 사용한다.
만화 캐릭터같은 것을 그릴 때 일러를 쓰는 사람도 종종 있을 정도이다. (추천하진 않는다)

2) 일러는 무조건...

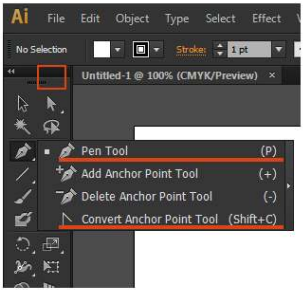
사진 편집과 효과 적용을 주로 하는 포토샵은 그 특성상 기능이 너무 많고 복잡하고 헷갈린다.
이 일을 전문적으로 하는 사람들도 포토샵을 100% 활용하고 있다고 확신하기는 힘들 정도이다.

일러스트레이터도 기능이 적은 건 아니다.
하지만 일러의 존재이유이자 가장 많이 쓰이고 일러가 있는 한 영원히 쓰게 될 툴은 딱 하나이다.

바로 **펜툴** 이다.

그러니 펜툴만 자유자재로 쓸 수 있다면 일러가 가진 활용성의 80%는 쓰고 있다고 봐도 된다.

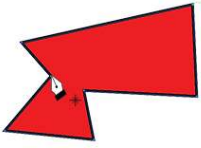
3) 펜 툴



포토샵과 달리, 일러는 펜툴을 활용하기 쉽도록 단축키도 전부 지정되어 있다.
포토샵 시간에 배운 펜툴처럼, Add와 Delete Point항목은 커서를 올려다 놓기만해도 변하므로
단축키 P키와 Shift+C키, 그리고 A키(흰색마우스)만 외워 두면 펜툴을 사용하는데 전혀 지장이 없다.

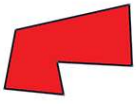
a) 사용법

펜툴을 클릭하거나 P키를 클릭하면 이렇게 생긴 포인터가 나타난다.
이것으로 문서에 클릭하면 점이 표시되는데, 여기부터 도형을 생성할 수 있다.
다른 지점에 점을 클릭하면 선이 되고, 원하는 만큼 점을 찍어 도형을 만들고,
처음에 찍었던 점을 찍으면 도형이 완성된다.

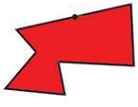


완성된 도형의 모서리에 펜을 올려놓으면
펜툴의 별모양이 -로 변하는데, 이는 해당 포인트를 도형에서
빼내겠다는 뜻이다. (중요 : 삭제가 아님!)

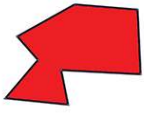
모서리가 아닌 선에 펜을 올려놓으면 반대로 +로 변하는데,
이는 해당 면에 포인트를 추가하겠다는 뜻이다.



① 포인트를 뺄 경우



② 포인트를 더할 경우(편집 전)

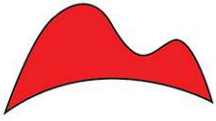


② 포인트를 더할 경우(편집 후)

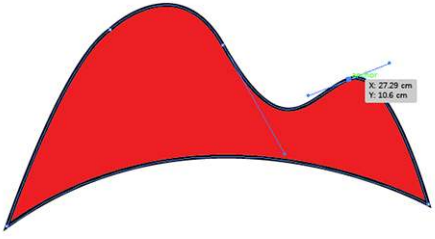
직선이 아닌 곡선을 만드는 방법도 포토샵과 동일하다.
한 점을 찍은 후 두번째 점을 찍을 때 마우스를 끌고 당기면
곡선은 핸들(handle)이라는 것이 생기는데 (그림의 파란 선)
이 핸들의 각도와 길이에 따라 원하는 만큼의 곡률과 길이를
지정할 수 있다.

즉, 찍으면 각을 줄 수 있고 끌면 곡선을 만들 수 있다.

b) 만들어진 도형의 편집



포토샵과 같이, 단축키 A키를 누르면 하얀 화살표가 나온다.
이를 이용해 이미 만들어진 도형의 모양을 편집할 수 있다.
사용 방법은 펜툴과 같이 모서리를 클릭하여 옮기거나
클릭 후 나타나는 핸들을 옮기면 된다.



점(포인트)을 클릭할 때 핸들이 표시되는 것을 볼 수 있다.

적절한 도형과 펜툴 및 A화살표(정식명칭은 셀렉션 툴)
사용으로 직접 만들어 본 카카오 라이언 캐릭터.

Part 6.
단축키와 패스파인더

어떤 프로그램을 쓰든지 단축키 사용은 작업 속도를 올리는 아주 좋은 수단이죠.
포토샵 및 일러스트레이터의 단축키에 대한 상세를 둘러봅시다.
마지막으로 일러스트레이터의 꽃, 패스파인더를 알아봅시다!

1) 미묘하게 포토샵과 다른 단축키들

어떤 프로그램을 쓰던지 단축키는 작업 속도를 올리는데 어마무지하게 영향을 미침.
포토샵도 그렇고 일러스트도 그렇지만,
같은 어도비 프로그램이고 상호 호환도 많이 하는 프로그램인데 이 둘의 단축키는 미묘하게 다르다.
다음은 자주 쓰게 되는 포토샵/일러스트의 단축키들이다.

보라색 : 포토샵/일러스트레이터 단축키가 동일함
파란색 : 포토샵 단축키
주황색 : 일러스트레이터 단축키

같은/다른 단축키, 같은 기능

a) V키 (Move tool)
가장 기본적인 마우스의 형태로 검은색 마우스 포인터 모양이다.
특정 레이어 및 오브젝트들을 클릭 후 드래그하여 움직이거나 크기를 조절하거나 하는
기본적인 편집이 가능.

b) A키 (Direct selection tool)
편집을 위한 마우스의 형태로 하얀색 마우스 포인터 모양이다.
벡터/펜툴로 만들어진 도형 및 이미지들의 세부적인 편집이 가능하다.
그룹으로 묶여진 레이어 및 도형들의 그룹을 해제하지 않고도 개별적인 편집이 가능.
왼쪽의 툴바에서 포토샵과 일러의 A마우스 위치가 다르지만 기능은 같다.

c) U키 / L,M키 (Shape tool)
도형을 만들 수 있는 툴이다.
포토샵의 경우 모든 모양이 U키 하나에 모여있어
해당 버튼을 길게 눌러 나오는 세부 모양을 정해야 한다.
일러스트의 경우 원 모양과 사각형 모양의 툴이 개별적으로 존재하며
단축키도 L(원)과 M(사각형)으로 다르다.

d) P키 / Shift+C키 (Pen tool / Convert point tool)
도형을 만들거나 부분적인 선택 지정을 할 때 쓰이는 펜툴.
포토샵에도 Convert point tool(약칭 꺽새)은 있지만 단축키가 지정되어 있지 않다.
일러스트에서 펜툴로 창작활동을 할 때 P키와 Shift+C키만 외운다면 작업속도 상승.

e) B키 (Brush tool)
브러쉬 툴. 더 이상의 자세한 설명은 생략한다.

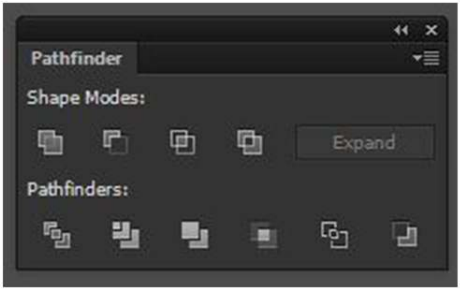
같은 단축키, 다른 기능

a) S키 (Stamp tool / Scale tool)
포토샵에선 주변 색을 복사 · 적용하는 Stamp 툴이지만,
일러에선 오브젝트의 크기를 조절하는 Scale 툴이다. (굳이 툴로 쓰지 않아도 V마우스로 가능)

b) J키 (Spot healing tool / Graph tool)
포토샵에선 주변 색과 해당 부분의 색을 섞는 힐링 툴이지만,
일러에선 각종 그래프를 적용할때 쓰는 Graph 툴이다. (강사도 쓴 적 없음)

2) 패스파인더

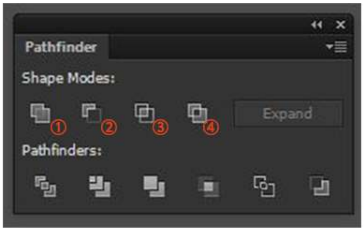
패스파인더는 아래와 같이 생긴 툴박스다.
만약 이 창을 찾을 수 없다면,
상단 메뉴 바에서 Window – Pathfinder를 클릭하거나 Shift+Ctrl+F9를 누르면 된다.
(단축키는 컴퓨터 환경에 따라 달라질 수 있음)



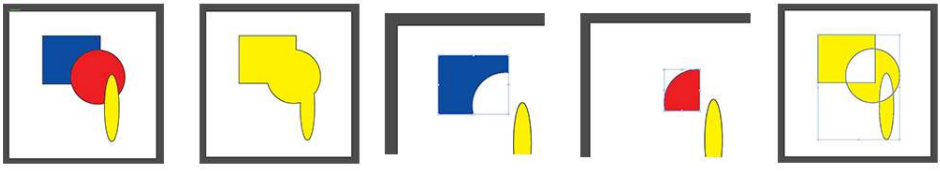
패스파인더란?
Path + finder = 펜툴의 고정점과 연결선 역할을 하는 Path와 찾는다는 의미의 find를 더한 기능.
즉 각종 도형들의 기준 외곽선을 찾아준다는 의미로,
그 역할은 **여러 도형의 합성 / 분리 / 교집합 / 빼기 등의 작업 수행**이다.
여러 도형을 많이 만들고 다루게 되는 일러스트레이터의 특성상 사용하면 매우 편리한 기능이다.

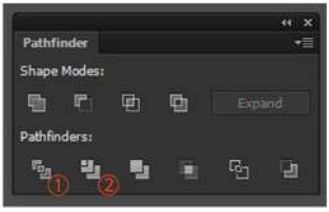
여기서는 가장 많이 쓰이는 몇가지만 설명하지만,
다른 기능들도 심심치 않게 쓰이므로 많이 가지고 놀다보면 무엇인지 알 수 있을 것이다.

사용법



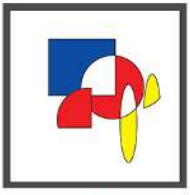
① 2개 이상 도형의 합체.
가장 위에 위치한 도형의 색으로 통일된다.
② 2개의 도형 중 상대적 차지 면적이 적은 쪽과
그 겹치는 영역을 삭제
③ 2개의 도형이 겹치는 영역만 남기고 삭제
④ 2개 이상 도형이 겹치는 영역만 삭제
가장 위에 위치한 도형의 색으로 통일된다.





① 전체 도형 분리

② 전체 도형 분리 (겹치는 아랫면 삭제)



①

②

#2
비전공자를 위한 글자 다루기 세미나

2018년 4월 18일
서희강

Basic Typography Seminar for Non Design Major

APR 18, 2018
SUH HEEKANG

똑같이 글자로만 되어 있을 뿐인데
내가 만든 피피티보다 다른 사람의 피피티가 더 좋아보이고
내가 만든 보고서보다 다른 사람의 보고서가 더 좋아보인다면
이 세미나에 그 비밀이 숨어 있습니다.

Seminar Instructor & Editor	서희강 SUH HEEKANG
Management & Administration	임정근 LIM JEONGKEUN
Illustration Designer	서희강 SUH HEEKANG
Online Development	이창훈 LEE CHANGHOON

글자만 적혀 있을 뿐인데 예쁘다고 느낀 적, 제목의 크기를 어떻게 해야 적절한지 고민해본 적 있다면 '생활 타이포그라피'로 함께해요

문자가 발명된 이래로 '글자를 어떤 형태로 담아내느냐'는 노력과 고민은 그 문자를 사용하는 공동체의 지적 지표 중 하나였습니다. 동양문화권과 서양문화권을 불문하고, '글자를 잘 쓰는 법'은 학문을 다루는 사람의 중요한 능력 중 하나였고 활자가 발명된 후에는 활자를 다루는 전문가들이 양성되었습니다. 하지만 컴퓨터가 보급되면서 손글씨를 디지털 활자가 대신하게 되었고 '글자를 잘 쓰는 법'이 '활자를 잘 다루는 법'으로 전이되지 못하고 소멸하게 되었습니다. 과거에 비해 개인 한 명이 생산해내는 활자 디자인물들이 압도적으로 증가하게 되었으나 역설적으로 이에 필요한 교육은 축소된 것입니다. 이 세미나는 학교에서 가르쳤어야 하지만 그러지 못한 것을 다루고 있습니다.

Our Instructor

서희강 SUH HEEKANG

- 홍익대학교 시각디자인 학사 졸업
- 하나은행 명동 플래그쉽 스토어 미디어 디자인 담당
- 2011 International Design Excellence Awards 동상
- 국카스텐, 김완선, 노을 앨범 자켓 디자인
- 현재 공군본부 홍보과에서 그래픽/굿즈 디자인 담당

Our Seminar

01. 개요

02. 무엇을 목표로 글자를 다루어야 하는가

03. 글자다루기-폰트

04. 이어질 내용

Typography + 2

생활 타이포그라피

글자 다루기

01 글자 다루기의 목표와 폰트

공군본부 정훈공보실 홍보과 미디어영상팀
서희강 **상병**

디자인 주요 이력

시각디자인 전공 학부 졸업
하나은행 명동 플래그쉽 스토어 미디어 디자인
2011 International Design Exellence Awards 동상
국카스텐, 김완선, 노을 앨범 자켓 디자인
콘텐츠제작병으로 입대하여
현재 공군본부 홍보과에서 그래픽/굿즈 디자인 담당

01. 개요

문자가 발명된 이래로, '글자를 어떤 형태로 담아내느냐'는 노력과 고민은 그 문자를 사용하는 공동체의 지적 지표 중 하나였습니다. 동양문화권과 서양문화권을 불문하고, '글자를 잘 쓰는 법'은 학문을 다루는 사람의 중요한 능력 중 하나였고 활자가 발명된 후에는 활자를 연구하고 활자를 다루는 전문가들이 양성되었습니다.

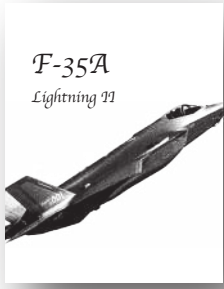
활자가 발명된 이후에도 컴퓨터와 프린터가 보급되기 이전까지는 '글자를 잘 쓰는 법'이 각급 학교와 가정에서 다뤄져야 하는 중요한 배움 중 하나로 취급되었으나 컴퓨터와 프린터가 보급되면서 '글자를 잘 쓰는 법'은 중요하지 않은 것으로 다뤄졌습니다. 이제 더 이상 '글자를 잘 쓰는 법'은 중요하지 않을 수 있으나 '글자를 잘 쓰는 법' 보다 상위 개념인 '글자를 잘 다루는 법'은 여전히 중요합니다.

하지만, 한국사회에서는 '글자를 잘 쓰는 법'이 '글자를 잘 다루는 법'으로 전이되지 못하고 소멸하게 되면서, 글자에 대한 조형교육은 디자인계열로 대학을 진학하게 되거나 개인적으로 관심을 가지는 경우, 혹 직무교육을 하게 되면서 받게 되는 교육으로, 피교육 대상 범위가 축소되었습니다. 각종 보고서, 발표 슬라이드 등, 과거에 비해 개인 한 명이 생산해내는 '활자가 들어간 디자인물'이 압도적으로 증가하였으나, 역설적이게도 이에 필요한 교육은 되려 축소된 것입니다.

글자를 다루는 학문인 '타이포그라피'는 전문적인 영역이고 그 범주와 깊이는 넓고 깊으나, 실생활에서 글자가 들어간 것들을 만드는 수준의 기초적인 글자 조형 능력은 누구나 익힐 수 있습니다. 또한 이러한 교육은 누구나 배워야할 영역으로 다뤄져야합니다. 그리고 이러한 것을 '생활 타이포그라피'라 부르고자 하며, 비록 제가 타이포그라피 전문가는 아니지만 군입대 전 1년간 크고 작은 강의를 하면서 구축된 내용과 2018년 7월 공군 정보체계관리단에서 진행한 세미나 내용을 바탕으로 본 교안을 제작하게 되었습니다.

02. 무엇을 목표로 글자를 다루어야 하는가

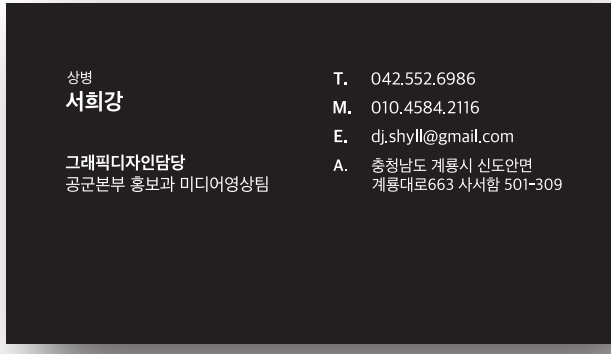
글자가 들어간 디자인의 경우, 좋아보이는 것과 그렇지 못한 것 간의 차이는 '글자가 어떻게 들어가 있냐'가 크게 좌우를 하게 됩니다. 가령 사진이나 그림이 들어간 포스터를 만든다고 했을 때, 뒤에 들어간 사진이나 그림은 상당히 별로지만 잘 디자인된 글자 한 줄이 들어갔을 뿐인데 굉장히 멋진 포스터가 된다거나 뒤에 들어간 사진이나 그림은 상당히 괜찮은데 그 위에 올라간 글자의 디자인이 엉망이어서 별로인 포스터가 되는 경우가 있습니다.



윗열의 항목들은 같은 사진에 글자 디자인만 변형한 것이고, 아랫열의 항목들은 같은 글자디자인에 서로 다른 사진들로 채운 것입니다. 윗열은 같은 사진을 썼음에도 각 항목에서 느껴지는 인상이 서로 달라서 일관성이 약한 반면, 아랫열의 경우에는 서로 관계 없는 사진을 썼어도 같은 글자 디자인을 통해 '어떠한 일관성'을 느낄 수 있습니다. 이를 통해, 그 어떤 요소보다도 '글자 디자인'이 디자인된 것의 표정과 일관성에 있어 큰 영향을 미침을 알 수 있습니다.

'글자 다루기'를 잘 해야하는 이유는 **시각적으로 문자 정보의 위계를 나눠서, 글자가 담고 있는 내용을 최대한 효율적으로 읽히게 함**에 있습니다. 대개의 경우, 정보의 위계가 잘 나눠진 것이 잘 읽히고, 그러한 것을 아름답다고 느낍니다.

위와 아래에 제시된 두 명함 디자인은 같은 내용과 같은 글씨체를 하고 있지만, 대부분의 경우 아래의 명함을 더 선호할 것입니다. 그 이유는 아래의 것이 더 잘 정돈되어 있다 느끼며 그로 인해 더 빠른 시간 안에 담고 있는 내용을 이해할 수 있고 더 아름답다 느끼기 때문일 것입니다.



'글자 다루기'는, **1.배치해야할 글자 정보**들을 **2.디자이너의 판단**으로 **3.정보의 목적**에 맞게 **4.잘 정돈**하여 **5.읽는이로 하여금 편안함 혹은 재미**를 느끼게 하는 행위를 말합니다.

물론, 위에서 서술된 목적에서 벗어난 목적 **실험적인 배치를 하거나 일부러 가독성을 떨어트려서 나름의 멋을 추구하는 경우 등**을 가진 작업물들도 존재하나, 여기서는 통상적인 실생활에서 필요한 '글자 다루기'를 다루고자 합니다.

03. 글자 다루기_폰트

글자의 표정을 가장 크게 좌우하는 것은 글자에 사용된 폰트입니다. 폰트는 용도 측면에서는 크게 제목용 폰트와 본문용 폰트로 나뉘며, 형태 측면에서는 크게 명조체와 고딕체, 그리고 손글씨체 등으로 나뉩니다.

제목용 폰트

안녕하세요

HG / HG꼬딕씨_Pro / 00g
제작사 / 폰트명 / 폰트 스타일

안녕하세요

Sandoll / 구운몽

안녕하세요

Sandoll / 격동고딕

안녕하세요

Yoon / 봄날 / L

제목용 폰트는 글의 본문보다는 제목에 어울리는 폰트로, 대체로 디자인의 컨셉이 강하거나 굵기가 굵어서 눈에 띄는 표정을 가지고 있는 것이 일반적입니다.

본문용 폰트

안녕하세요

Sandoll / 산돌고딕 Neo1 / Medium

안녕하세요

Sandoll / 산돌명조 Neo1 / Medium

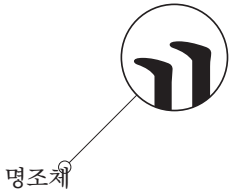
안녕하세요

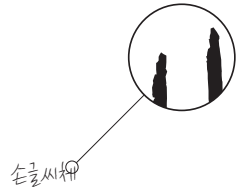
Yoon / 윤고딕 300 / 30

안녕하세요

Yoon / 윤명조 300 / 30

본문용 폰트는 본문에 사용될 것을 목적으로 만들어진 폰트로, 대개는 한 폰트 내에서 여러 굵기가 개발되며 가독성에 초점을 맞추어서 개발되는 것이 일반적입니다. 본문용 서체는 가장 기본이 되는 서체이기에 디자이너들은 자신이 주로 사용하는 본문용 서체를 두는 편이며, 제목용 폰트를 본문에 사용하는 것은 일반적이지 않으나, 본문용 폰트를 제목에 사용하는 것은 흔히 있는 일입니다.





명조체는 글자의 끝에 돌기가 있는 서체를 말하며 고딕체는 글자의 끝에 돌기가 없는 서체를 말합니다. 간혹 돌기가 있다고 볼 수도 있고 돌기가 없다고 볼 수도 있는 중간 형태의 서체들도 존재하나, 통상적으로 서체는 명조체와 고딕체로 나뉩니다. 글자의 끝에 있는 돌기를 serif라 부르는데, 20세기 이전만 하더라도 모든 서체는 돌기가 있는 서체였으나 20세기 초반들어서면서부터 돌기가 없는 서체들이 개발되자 이러한 서체를 돌기가 없다는 의미로 'sans serif' sans는 프랑스어로 없다는 의미라 불렀고 serif는 돌기를 의미하기도 하지만 돌기가 있는 서체를 지칭하기도 하는 말이 되었습니다. 손글씨체는 디지털 서체 개발의 도입으로 실제 손글씨를 스캔하여 활자로 만드는 것이 가능해지면서 탄생하게 된 서체입니다.

명조체

안녕하세요

Sandoll / 산돌명조 Neo1 / Regular

안녕하세요

안삼열 / 310 안삼열 / M

안녕하세요

Yoon / 윤명조 400 / 40

안녕하세요

Asia / a뉴궁서 / M

명조체는 고딕체에 비하여 정갈하며 유려하고 옛 것의 질감을 가지고 있습니다. 글자 크기가 작고 그 수가 많을 경우에는 명조체가 고딕체에 비해 가독성이 높아 신문 본문과 책 본문에는 주로 명조체가 쓰이나 지향하는 컨셉에 따라서 본문에 고딕체를 채택하는 잡지나 책도 많습니다. 과거에는 '제목에는 고딕체, 본문에는 명조체'와 같은 공식이 있긴 하였으나 이는 허물어진지 오래되었고 명조체의 정갈하고 유려한 느낌을 살려서 단독적인 텍스트나 글의 타이틀에 사용하는 경우도 많습니다.

고딕체

Yoon / 화이트핏 / 44

Rix / Kopub 돋움체 / Medium

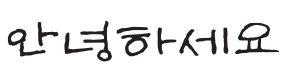
Sandoll / 산돌고딕 Neo2 / SemiBold

Asia / a옛날목욕탕 / B

고딕체의 경우 20세기 초반 대량생산 시대의 요구에 의해 만들어졌고 장식적인 요소가 명조체에 비해 적기에 효율적이고 합리적인 표정을 가지고 있습니다. 주로 제목이나 단독적인 텍스트에는 고딕체가 사용되는 경우가 많으나 위에서 언급했듯 고딕체와 명조체 간의 목적에 따른 엄격한 용법은 상당 수 허물어졌습니다. 다만, 웹 환경이나 모바일 환경에서는 고딕체의 가독성이 높고 고딕체의 감성이 더 어울리는 경우가 많아서 고딕체의 사용 비중이 높습니다.

손글씨체

Sandoll / 공병각펜 / Medium



Yoon / DS아스팔트스캔들 / L

덧붙여, 특수한 목적에 따라 개발되는 서체들이 있습니다. 기업 전용 서체나 공공기관 전용 서체, 그리고 그 외의 목적을 가진 폰트들이 바로 그것입니다.

기업 전용 서체

안녕하세요

Sandoll / 배달의민족 도현

안녕하세요

Rix / 야놀자 야체

안녕하세요

Rix / Tmon티몬소리 / Black

안녕하세요

AG / 아리따 돋움 / Bold

안녕하세요

Sandoll / 나눔스퀘어 / Light

안녕하세요

알 수 없음 / tvN 즐거운이야기 / Light

기업에서는 다른 기업이나 제품과 구별되기 위해서 '브랜드'라는 개념을 적극 도입합니다.
브랜드 이름을 짓고 브랜드 로고를 만들고 브랜드 색상을 정하고 브랜드 스토리를 만드는 식으로 말이죠.
'브랜드'라는 개념이 처음 생겨났을 때는 단순히 다른 것과 구별되기 위해서 생겨났는데 시간이 지남에 따라 브랜드는 단순히 다른 제품과 구별되기 위해서 사용되는 것이 아니라 다른 소비자와 구별되기 위해서 사용되기 시작하였습니다. 이에 따라 '브랜드'가 소비자에게 어떤 경험과 인상을 남기느냐를 연구했고 그 결과 브랜드에게는 '컨셉'과 '일관성'이 중요하다는 결론에 도달하게 됩니다.

21세기에 이르러서는 브랜드에 있어 로고를 만들 때만 폰트를 고민하는 것이 아니라 소비자가 브랜드를 이용할 때 접하는 글자까지도 일관된 컨셉에서의 경험을 봉합하기 위해 기업 전용 서체를 개발하기에 이릅니다. 그리고 현재에는 상당수의 기업들이 브랜드 컨셉에 맞는 전용서체들을 보유하고 있으며 여러 지자체 및 공공기관에서도 전용서체들을 보유하고 있습니다.

공공기관 전용 서체

Sandoll / 포천막걸리체

Sandoll / 제주한라산

안녕하세요

Yoon / 서울남산 / 장체L

안녕하세요

한그리다 / 이순신돋움체 / M

공공기관 및 지자체 역시도 그 기관을 이용하는 사람들에게 일관된 인상을 주기 위해 전용 서체를 개발하는 기관이 점차 늘고 있습니다. 또한 디지털환경에서 쓰이기 위해서, 신문에 사용되기 위해서, 도로표지판에 사용되기 위해서 등 다양한 목적을 가진 서체들이 있습니다.

여러 목적을 가진 전용 서체

안녕하세요

Sandoll / Apple SD고딕 Neo / Regular
애플 제품의 디스플레이에
들어가는 용도로 개발

안녕하세요

Sandoll / Noto Serif CJK KR / Light
구글에서 한중일 통합 웹용서체의
필요에 의해 개발

안녕하세요

Sandoll / 도로표지판 / M
도로표지판에 들어가는 용도로 개발

안녕하세요

Rix / KoPub 바탕체 / Light
한국출판인회의에서 종이책과 e-book을
포함한 서적 매체에 들어가는 용도로 개발

다음 편에서는 크기/굵기/장평/행간/자간 등,
글자를 다루는데 있어 본격적인 사항들을 다루겠습니다.

공감

대한민국공군
REPUBLIC OF KOREA AIR FORCE

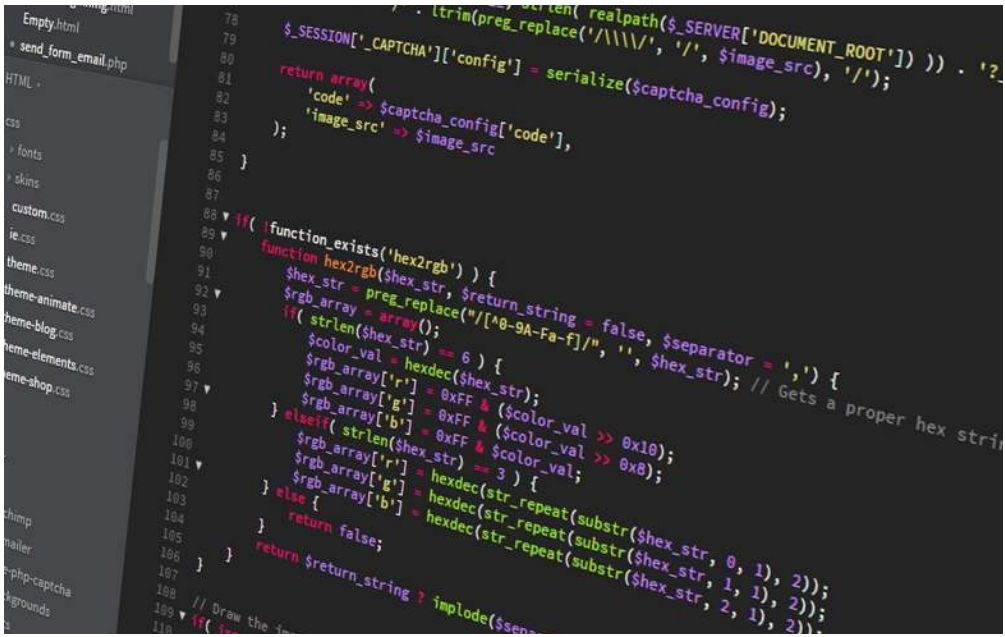
#3
굿 프로그래밍 &
모던 프로그래밍

2018년 5월 16일
주호영

Good & Modern Programming with Vue, TypeScript & Functions

MAY 16, 2018
JOO HOYOUNG

여러분은 객체지향에 대해 잘못 알고 있습니다.
여러분은 함수형에 대해 잘못 알고 있습니다.
그런 당신을 위한 모-던한 프로그래밍



Seminar Instructor & Editor	주호영 JOO HOYOUNG
Asst. Editor	곽신우 KWAK SHIN WOO
Management & Administration	임정근 LIM JEONGKEUN
Illustration Designer	이창훈 LEE CHANG HOON

※ 본 매거진 섹션은 1주차 내용을 다루고 있습니다. 2주차부터 실습 주제가 체계단 내부 프로젝트로 되어 있어 군사보안상 부득이하게 삭제되었습니다. 이 점 양해바랍니다.

함수형 프로그래밍이 어렵다는 오해를 풀어보고 진정으로 좋은 코드가 무엇인지 알아보자.

체계단에서 개발하는 거의 모든 병사들은 Java나 C++ 등의 객체지향적 프로그래밍 언어로 코딩을 할 수 있습니다. 하지만 '객체지향적' 프로그래밍을 하는 방법에 대해서는 잘 모르고 있습니다. 본 세미나에서는 객체지향 프로그래밍 입문서에서 대충 넘어가거나, 가장 중요한 부분 임에도 불구하고 문법을 익히기 전이라 중요성을 인식하지 못했던 몇 가지 질문들. '도대체 객체란 무엇일까?', '함수란 무엇일까?', '프로그램이란 대체 무엇일까?'에 대한 답을 고민해봅니다. 함수형 프로그래밍 방법론과 웹 기술의 최신 트렌드인 Vue, TypeScript 실습과 함께 궁극적으로 어떤 코드가 '잘 짠'코드인지 스스로 판단할 수 있을 만한 기준을 제시해줍니다.

Our Instructor

주호영 JOO HOYOUNG

- 서울과학고등학교 ('10 ~ '12)
- 서울대학교 ('13 ~ '15.08)
- LandComplete Team RideZero ('15.09 ~ '16.08)

Our Seminar

1주차. 전체적인 세미나의 방향

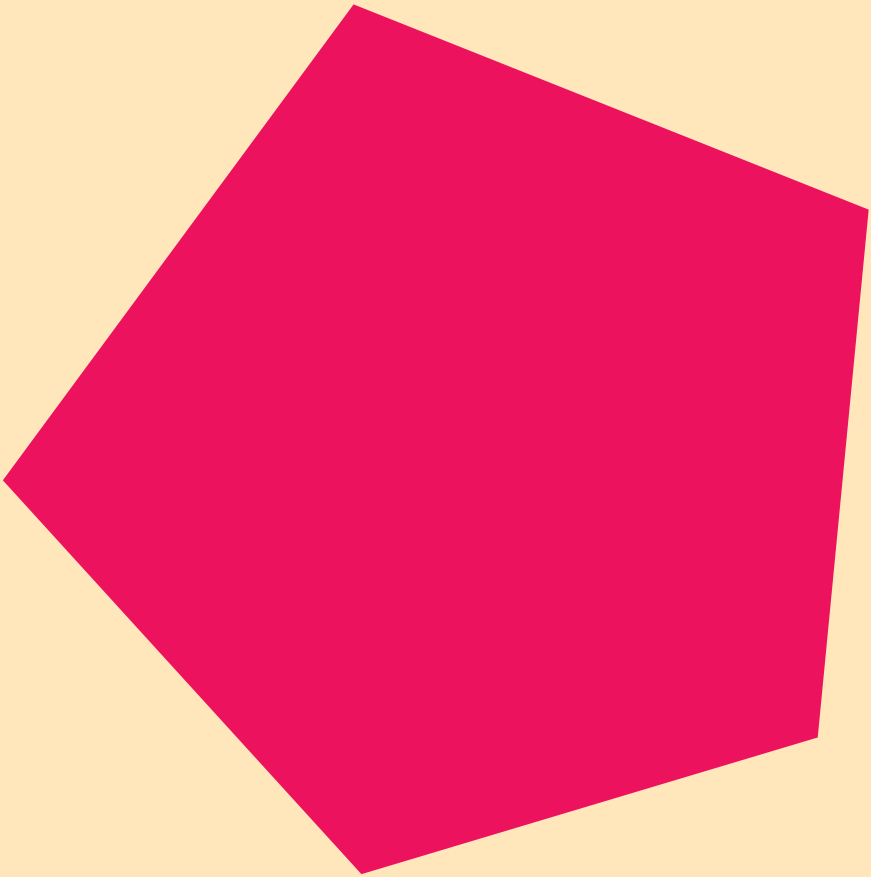
- 객체지향 프로그래밍, 함수형 프로그래밍에 대한 소개 및 인식 개조
  ; OOP, FP의 진정한 목표
- 좋은 코드란 어떤 코드를 말할까? ; 간단한 클린코드 원칙 소개
- 프로그래밍 패턴은 대안이 아니다 ; OOP의 코드 결합도 문제,
  프로그래밍 패턴 (Java) 간단 소개
- 함수형을 쓰면 어떤 코드가 나올까?

2주차. 실습을 중심으로 한 TypeScript, Vue.js 소개 및 개론

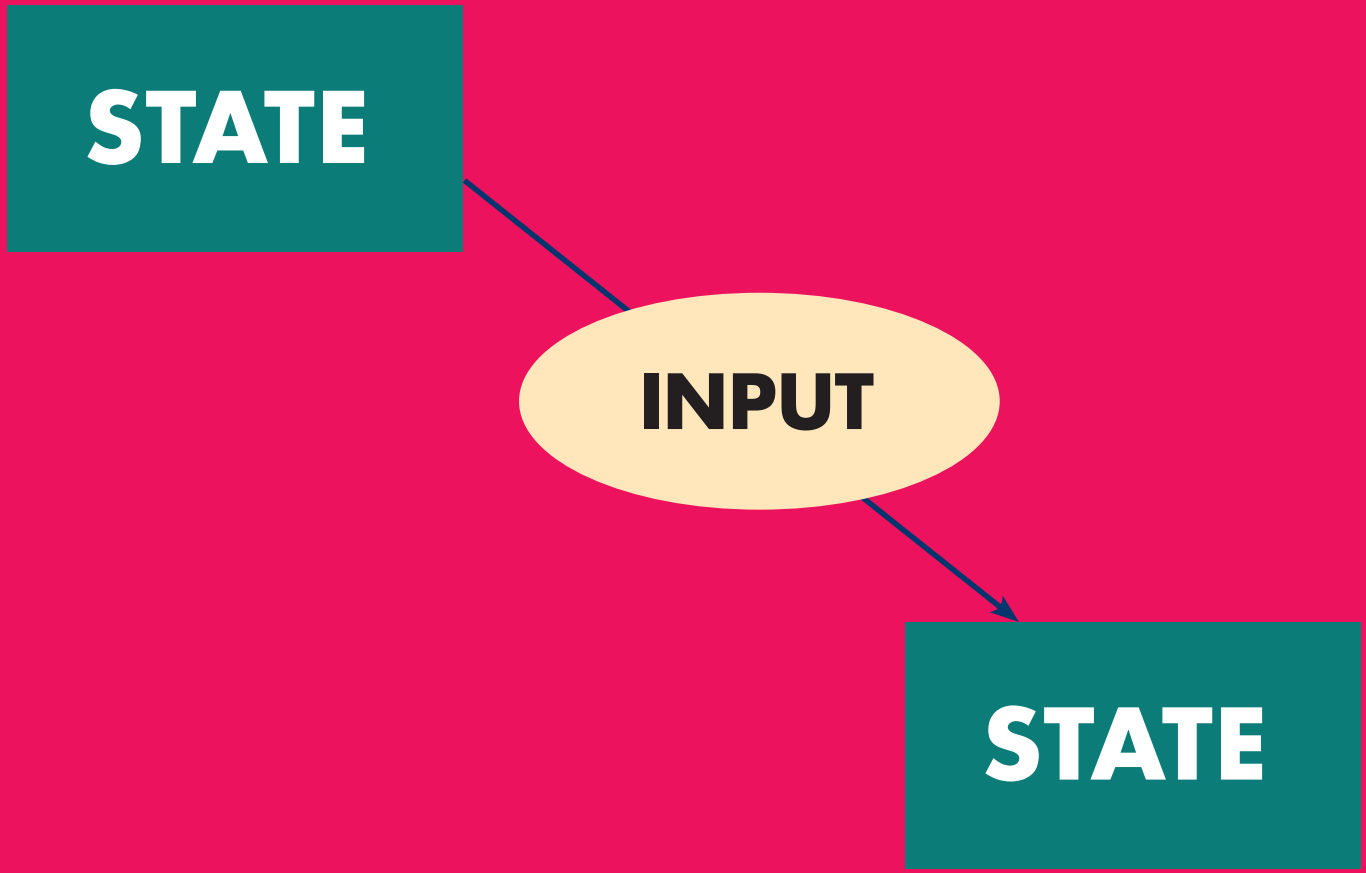
3주차. 테스팅과 함수형 실습

4주차. Vue 심화, Vuex 사용

- Vue의 컴포넌트 소개 및 Vuex 사용을 통한 클린한 데이터 흐름 만들기



Good/Modern + 3


STATE
INPUT
STATE

WHAT IS STATE?

프로그램이 접근 가능한 모든 것
변수 (클래스 내부, 전역 변수, 함수 내부)
눈에 보이는 것
눈에 보이지 않는 것
사용하는 컴퓨터 환경

WHAT IS INPUT?

키보드 입력
마우스 입력
함수의 인수
기계의 변화
시간의 변화

WHAT IS PROGRAM?

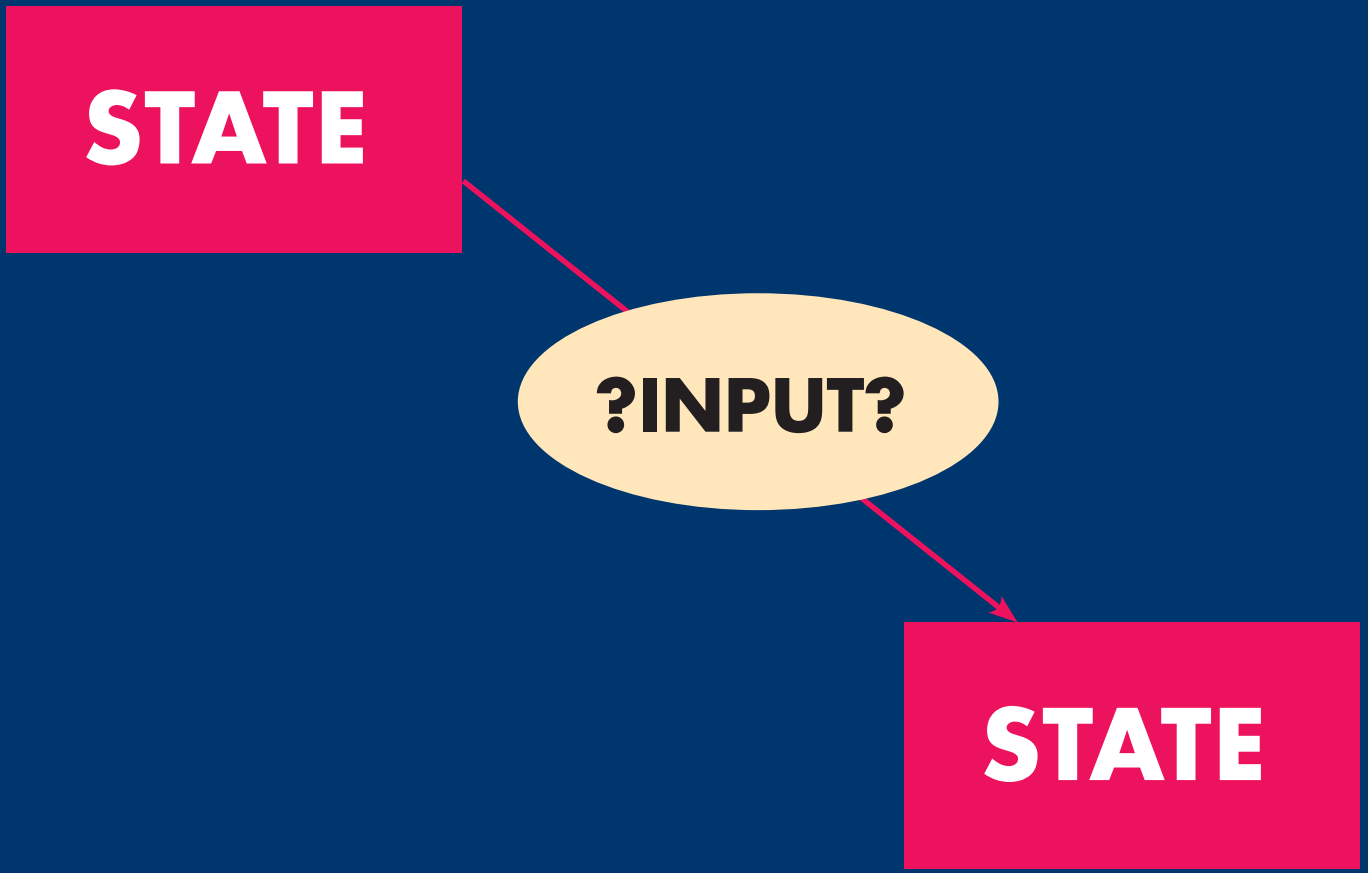
임의의 상태에 있는 어떤 것에 입력을 함으로써 다른 상태로 전이시키는 것

시스템을 하나의 상태로 보면 입력 또한 상태가 된다.

상태도 입력의 일종으로 볼 수 있다.

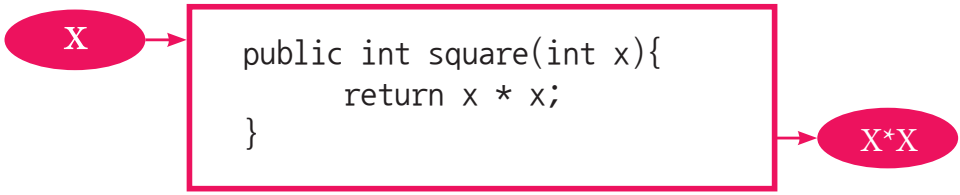
프로그램이란 상태의 연속이다.

TURING MACHINE


STATE
?INPUT?
STATE

IT'S THE SIDE-EFFECT, STUPID!

상태가 보이지 않는 INPUT을 만든다. 이는 Side-Effect(부작용)를 발생시켜 문제가 된다.




```
public void processNext() {
      Message message = messageQueue.popMessage();

      if (message != null) {
          process(message);
      }
}
```

messageQueue
process()

WHAT IS SIDE-EFFECT?

함수의 동작에 필요한 함수 외부의 무언가
- 인수 이외의 입력
- 시간, standard I/O, 파일, 전역 변수, 외부 변수

함수 실행 전후로 바뀌는 함수 외부의 무언가
- 전역변수 수정, 파일 수정, 입출력...
- 함수 인수로 들어온 객체 레퍼런스에 변형을 가해도 Side-Effect

부작용은 복잡성을 유발한다!

함수 외부 상태, 즉 world(계)에 대한 가정이 존재해야만 한다
- 객체를 제대로 생성, 초기화 했는가?
- 다른 클래스의 전역변수 값을 제대로 조정했는가?
- 현재 시간이 혹시 문제가 되지는 않는가?
- 함수들의 실행 순서를 제대로 조절했는가?
- 내가 퇴근한 후 누가 내가 쓰고 있는 함수를 변경하진 않았을까?

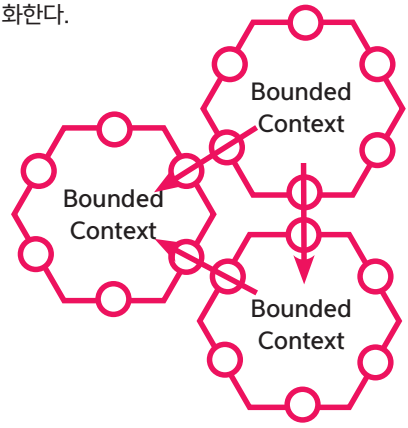
부작용은 버그를 유발한다!

사용자가 예상하지 못하는 입력과 출력 ==
사용자가 예상하지 못하는 행동 ==

BUG!

객체 지향 언어도 이런 문제가 있나요?

객체지향은 부작용을 객체라는 틀에 가둠으로써 상태 변경을 지역화한다.



캡슐화 : 코드의 안을 그저 보이지 못하게 하는 것
코드의 내부에는 외부 상태에 대한 숨겨진 가정이
존재할 수 있음

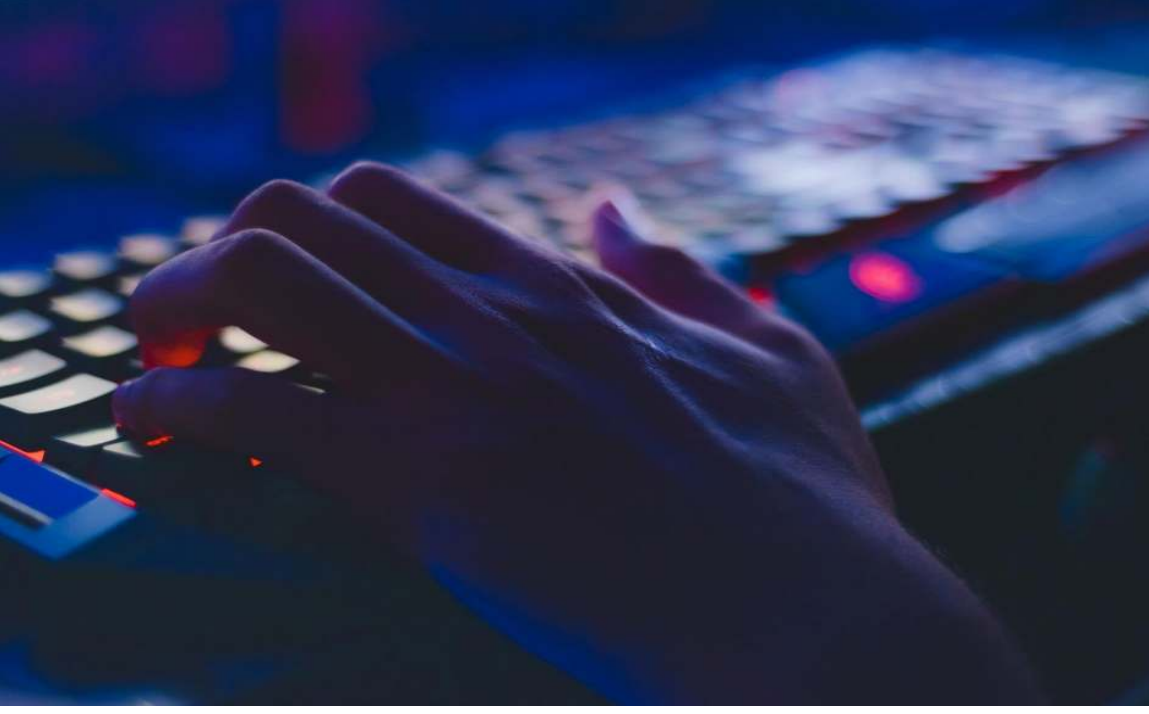
```
class Person {
    protected void walk() {
        System.out.println( " let ' s walk! " );
    }
}

class Villain extends Person {
    @Override
    protected void walk() {
        launchNuclearBombAfter(5000); // 5 seconds
        System.out.println( " let ' s walk! " );
    }
}

Person ordinaryCitizen = new Villian();
```

캡슐화의 문제

캡슐화의 문제는 사용자 측이 캡슐화가 제대로 되었다는 것을 믿을 수단이 없다는 것이다. 문서화 되지 않은 함수 로직이나 상태 변경이 존재 시 모든 것이 틀어질 수 있다. 고급 사용이나 버그 발생 시 코드를 뜯고 캡슐을 열어야 한다. 결합도 제거 또한 자동으로 해주지 않는다. 코드가 결합되면 수정이 어려워진다. 즉, 디버깅이 어려워진다는 뜻이다.

상속의 문제

상속은 결합도를 극단적으로 올리는 최악의 시스템이다. '잘 쓰면 되지 않을까?'하는 의문이 들 수 있으나 상속을 전문적으로 사용한 케이스가 극히 드물고 대학에서 전공을 공부하는 대부분 사람들에게는 어려운 문제이다. 특히 후임들은 이러한 코드를 알아보기 어려운데, 상속은 최초 작성자 이외에는 코드의 숨겨진 가정이나 숨겨진 입력을 모른다. 숨겨진 입력이 있으면 코드를 처음부터 끝까지 뜯어봐야한다.



부작용을 사용하지 않으면 될까?

같은 Input이라도 기존 상태에 따라 결과가 달라지는 것 자체가 부작용이다. 즉, 내부에 상태를 가지면 부작용은 불가피하다는 것. 하지만 억제할 수 있는 방법은 존재한다.

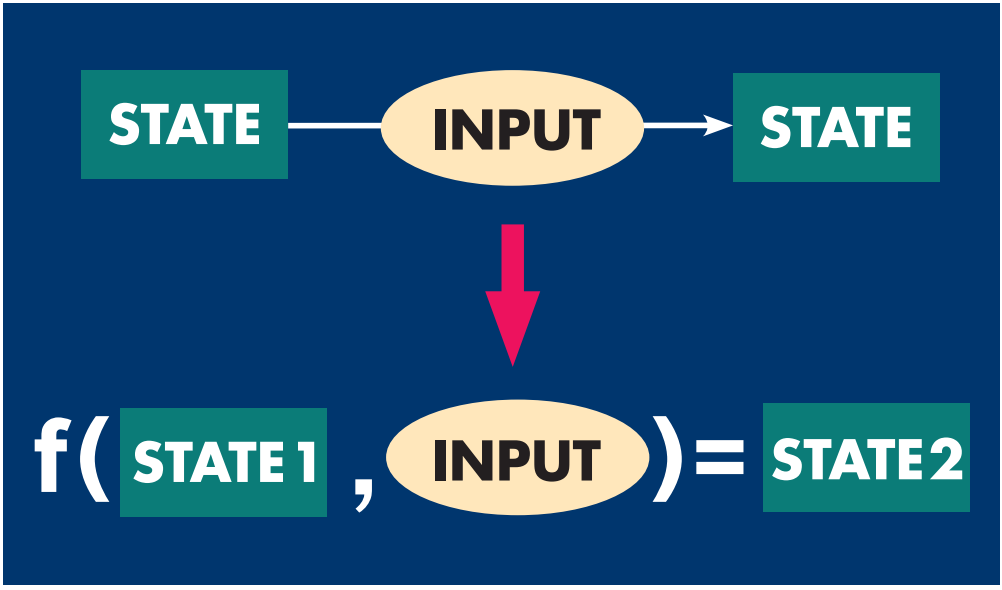
답은 함수형: 함수형의 목적은 부작용의 최소화이다

그 누구도 가르쳐주지 않았던 함수형의 목적, 바로 '부작용을 없애자'이다. 고차 함수, 재귀 함수, 람다 함수 등, 우리가 알고 있는 이러한 함수들은 함수형의 목적이 아니라 결과이다. 함수형의 유일한 목적은 부작용을 없애는 것이다. 어떻게 부작용을 없앨까? 답은 바로 순수함수에 있다.

순수함수란?

- 숨겨진 입력이나 출력이 존재하지 않는 함수
- 수학의 함수와 같이 한 입력에 대해 유일한 고정된 출력이 존재한다.
- 실행시킨다고 전체에 영향을 주지 않는다.
- 함수형 패러다임에서 (거의) 모든 함수는 순수성을 지켜야 한다.

순수하지 않은 함수 vs. 순수한 함수



1. 순수하지 않은 함수

```
public Program getCurrentProgram(TVGuide guide, int channel) {
   Schedule schedule = guide.getSchedule(channel);
   Program current = schedule.programAt(new Date());
   return current;
```

2. 순수한 함수

```
public Program getProgramAt(TVGuide guide, int channel, Date when) {
   Schedule schedule = guide.getSchedule(channel);
   Program program = schedule.programAt(when);
   return program;
}
```

3. 순수함수는...

1) 의존성이 존재하지 않는다. 즉, 변경이 쉽다.
2) 환경설정이 필요 없으므로 테스트 하기 쉽다.
3) 테스트를 통해 검증한 코드는 정상 작동이 증명된다.
4) 함수 정의만을 보고 행동 유추가 가능하다.
5) 예상하지 못한 동작을 전혀 하지 않는다.
6) 순서 의존성이 사라진다. 즉, 멀티스레딩에 효과적이다.

BEHIND STORY

첫과 끝의 이야기

모든 것에는 첫과 끝이 존재한다. 우리는 이 '첫'과 '끝'을 매우 중요하게 생각한다. 이를 간직하기 위해 조그마한 이벤트를 열기도 하고 날짜를 새긴 기념품을 나누기도 한다. 이렇게 무수한 첫과 끝을 거쳐서 계절이 흐르고 세월이 흘러, 마침내 우리는 점점 더 성장해나간다.
샘이나에도 이러한 첫과 끝이 존재한다. 지난 2월 말부터 시작된 세미나, 이제는 꽤 쌀쌀해진 아침이 가을이 왔음을 알린다. 첫 세미나를 맡았던 최문규 병장(775기), 그리고 세미나를 끝으로 군생활을 마치는 주호영 예비역(768기). 이번 매거진에서는 이들을 만나 그들의 첫과 끝에 대해 들어보았다.

첫

775 최문규

- 첫 세미나의 주인공! (짝짝짝) 먼저 본인 소개 부탁드립니다.
- 안녕하세요. 775기 최문규라고 합니다. 샘이나에서 '포토샵과 일러스트레이터의 기초'에 대해서 6주동안 세미나를 진행했었습니다.

- 벌써 시간이 흘러서 가을이 되었습니다. 첫 세미나가 3월이었었는데, 기억이 나시는지?
- 네, 처음에 임정근 병장이 '우리가 서로 재능을 나눌 수 있는 프로젝트를 기획하고 있는데 포토샵을 혹시 가르쳐 줄 수 있겠냐'하고 물었던게 기억납니다. 디자인이라는게 코딩을 하는 프로그래머들에게도 관심사가 되다보니 이외에도 꽤 많은 분들의 수업 요청이 있었어요. 처음 수업에 딱 문을 열었더니 사람들이 생각보다 많이 와서 놀랐었습니다.(웃음)

- 세미나를 준비하시면서 어려운 점이 있으셨다면?
- 사실 저는 이렇게 남들앞에서 세미나 같은 걸 해본 적이 전혀 없었습니다. 가르쳐 본 경험도 전무하고요. 그래서 처음 하는 경험에 긴장을 꽤 했었습니다. 포토샵을 처음 보는 분들에게 단축키 설명도 하지 않고 넘겼다가 수강생 질문에 다시 설명하기도 했었죠. 하지만 누군가가 제 이야기에 오랫동안 관심을 가지고 열심히 들어준다는 것은 정말 뿌듯한 경험이었어요. 그런 분들과 함께하니 수업할 맛이 나더라구요.(웃음) 점점 더 적응을 하면서 마지막 수업 때는 수강생들과도 돈독한 사이가 되었습니다.

- 이 세미나를 처음 부탁받았을때, 맡게 된 이유가 궁금합니다.
- 사실 포토샵이라는 툴은 몇가지 기본적인 기능만 익혀두면 원하는 대부분의 효과를 얻어낼 수 있습니다. 보통 많은 분들이 하고 싶어하시는 사진 보정, 증명사진, 합성 등은 몇 가지 기능으로 뚝딱 만들어낼 수 있어요. 그래서 이 세미나를 듣는 분들이 스스로 원하는 작업을 할 수 있는 힘을 길러드리고 싶었습니다. 더 이상 디자인 전공생들을 괴롭히지 말아주세요(?)

- '샘이나'에게도 첫 세미나였지만 본인 스스로에게도 첫 세미나 경험이었는데요. 소감 한마디?
수강생들과의 호흡이 가장 기억에 남습니다. 적극적인 참여와 질문이 수업을 더 풍부하게 만들었던 것 같아요. 제가 혼자서 진행했으면 놓쳤을 것들도 잡아주시더라구요. 잊지 못할 경험이었습니다!

언제나 처음은 어렵고, 누구나 처음은 서툴다.

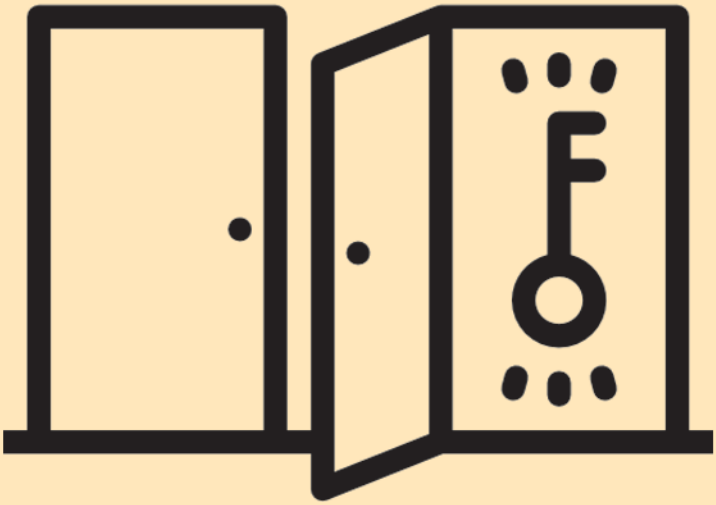
우리는 그것을, 그때 그 순간들을 종종 '시절'이라 부른다.

시절을 지나 다시 여기.

지금보다 더 높은 곳에서,

더 오랫동안, 더 자주 만날 인연이 되기를.

@dazzi_mozzi



끝

768 주호영

– 정말 이제 전역을 하시네요.
먼저 본인 소개부터 부탁드릴게요.

– 안녕하세요. 768기 주호영이라고 합니다. 저는 올해 9월에 전역을 하고요. (웃음) 샘이나에서 '좋은 프로그래밍이란 무엇인가'에 대해서 이야기를 나눈 적이 있습니다.

– 전역이라니. 정말 부럽습니다. 사실 이번에 맡으셨던 세미나가 '너의 재능, 샘이나'에서 진행된 첫 프로그래밍 세미나입니다. 처음 세미나를 맡게 된 계기는 무엇인가요?

– 사실 저는 프로그래밍 관련 글을 공군 카페 '휴머니스트'에서 연재를 하고 있었습니다. 몇 개의 글을 올렸는데 수많은 댓글이 달리고 사람들의 관심을 갖게 되는 색다른 경험을 하게 되었어요. 사실 많은 사람들에게 저의 관심사에 대해 이야기할 수 있는 경험은 흔치 않았습니다. 대학교 다닐 때 했던 발표 수업이 전부였죠. 근데 군대에서 이런 꽤나 임팩트 있는 경험을 하고나서 '나의 관심사를 이렇게 소개하는 일'에 재미를 느끼기 시작했습니다. 그래서 이렇게 세미나에 강사로 참여하게 되었어요.

– 당시 세미나 계획 공지를 올렸을 때도 게시판에서 댓글이 많이 달리면서 관심을 받았던 기억이 나네요. 세미나를 진행하면서 어떠셨는지 궁금해요

– 준비하면서 스스로에게 공부가 될 수 있어서 좋았던 것 같아요. 내가 내 주제에 대해 명확히 말할 수 있도록 다시 한번 '좋은 코딩'과 '나쁜 코딩'의 기준을 생각해 보았어요. 그리고 이런 세미나를 통해 사람들에게 좋은 코딩이 무엇인지 다시 한번 생각해볼 수 있는 계기를 마련한 것 같아 나름대로 뿌듯함도 느꼈습니다. 사실 지금 현업에 있는 프로그래머들도 20년, 30년동안 나쁜 코딩 습관을 가져가는 경우가 있거든요. 군생활 막바지에 이렇게 좋은 경험을 할 수 있어서 정말 진심으로 좋았습니다.

– 말씀하신대로 이제 곧 군생활이 끝나시게 되는데요. 마지막으로 체계단에서 지금 프로그래밍을 하고 있는 장병들에게 바라는 점이 있다면요?

– 체계단에 있는 대부분 사람들이 대학교를 다니다가 입대한 친구들입니다. 따라서 사실 현업에 대한 경험이 많이 부족하고, 대부분 군대에서 하는 업무가 첫 현업 경험일 수 있어요. 그럴수록 남들의 코드를 더 익히고 나만 이해할 수 있는 코드를 지양해야 합니다. 우리가 맡고 있는 업무는 결코 우리가 평생 지고 나갈 일이 아니기 때문입니다. 우리 후임들이 알아볼 수 있는 코드가 결국은 '좋은' 코드 아닐까요.

#4
전공자를 위한
외국계 취업 세미나

2018년 8월 10일
양민호

Foreign Company Job Seminar
For Computer Science Major

AUG 10, 2018
RYANG MINHO

회사는 다 똑같은거 아닌가? 외국계는 무엇이 다르길래?
나와 잘 맞을까? 내가 들어갈 수 있을까?
이러한 궁금증을 갖고 있는 분들을 위한 세미나입니다.

Seminar Instructor & Editor	양민호 RYANG MINHO
Management & Administration	이창훈 LEE CHANGHOON
Reference	GOOGLE CAREER
Online Development	이창훈 LEE CHANGHOON

해외 취업에 대한 막연한 궁금증 경력자의 경험과 조언으로 풀어보자 구글 소프트웨어 엔지니어 양민호

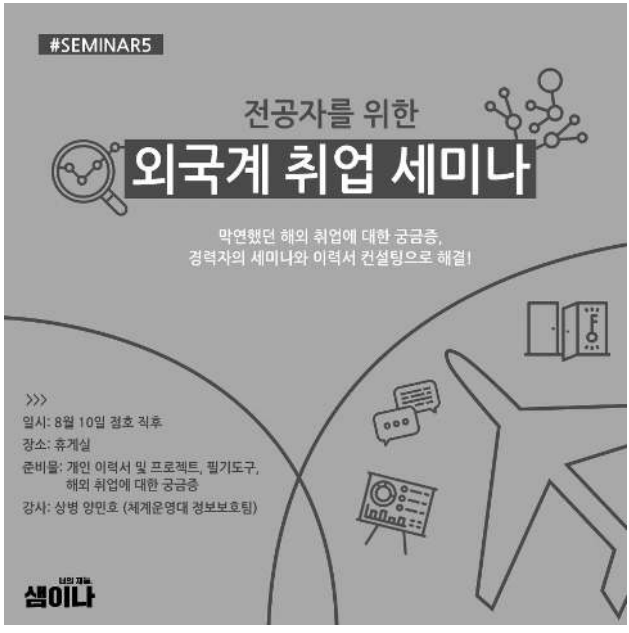
그는 GOOGLE TV로 외국계 기업에 처음으로 입사했다. 평소 관심 있었던 분야의 프로그램을 만들었던 것이 이 길의 시작이었다.
그들이 일하는 방식, 공유하는 문화는 수강자들의 마음을 잡기에 충분했다. 중요한 것은 해외 취업을 너무 남의 일로 치부하지 않아도 된다는 것. 이제부터 그의 이야기를 들어보자.

MinHo Ryang as a...

Googler

Non-Googler

SHARE.
SPECIAL.



Poster for a Foreign Company Job Seminar

4

Google + 4

왜 좋은가?

$$$$$
Flexible Time
Your Goal(OKR)
Bleeding Edge Technology
Full Support for *

Your 20% Project
Google University
All product, source code, dash-board

Multiple Region
Multiple Product / Track / Level

DogFooding
Google Slumni

안녕하세요, 외국계 취업세미나를 맡게 된 양민호입니다. 제 경험을 통해서 외국계의 취업이 어떤 방식으로 진행되는지, 어떻게 공략하면 좋을지 한 번 생각해보시면 좋을 것 같습니다.

이 슬라이드는 우선 청자의 관심을 끌기 위해 작성되었습니다. Google이 써져있는 곳에는 자신이 가고 싶어하는 다른 회사 이름을 넣어 생각하셔도 좋습니다.

굳이 한마디로 정의하자면 Flexible로 말할 수 있습니다. 내가 말하는 시간과 시간대부터, 일할 목표, 사용할 기술, 필요한 지원, 그리고 나만의 개인 프로젝트까지 이 모든걸 자신의 의지로 관철할 수 있습니다.

공부가 필요하다면 회사 내 다양한 교육 자원이나 TechTalk을 참고할 수 있고, 기존에 존재하는 프로덕트들의 자료나 코드, 심지어는 대시보드까지 볼 수 있어, 다양한 부분에서 성장을 하실 수 있습니다.

개밥먹기로 알려진 Dogfooding으로 새로운 기술/기능/프로젝트/제품을 맘껏 써볼 수도 있고, 매우 훌륭한 동료들을 얻을 수 있습니다. 이 장점들을 조합하여 얻을 수 있는 시너지로 저와 다른 Googler들은 리프레시와 에너지를 얻습니다.

Flexible Time과 Multiple Region을 조합하면 근무를 하면서도 여행을 다닐 수 있고, 20% Project와 Multiple Track/Level을 조합하면 내 아이디어를 실현하는 Project Owner가 될 수 있습니다.

나도 갈 수 있을까?

이력서 제출 → 면접 → 합격
인턴도 있다!
떨어져도 1년 후에 다시 도전할 수 있다.
상시채용

지원하는 과정에서 이질감을 느끼실 필요가 없습니다. 다른점은 공채가 없고 상시채용이라는 것입니다. (한국도 요즘 비슷하게 흘러가고 있죠) 일정이 따로 정해진게 아니라, 내가 지원하면 일정이 시작된다고 보시면 됩니다. 회사마다 지원을 위한 페이지가 있어, 거기에 모집 공고가 올라오며, 필요한 사람을 구하면 공고가 내려가는 구조입니다. 모집공고 / 이력서 제출 / 면접 / 합격 순서로 진행되는 거죠.

어떻게 하면 합격하나요?

Excellence

+Di

면접에서는 무슨 질문을 받나?
실제 면접관이 물어보는 시작 질문 리스트

아래는 제가 면접관일 때 물어보는 시작 질문들 리스트입니다.
이 질문들로부터 시작해, 실제 문제해결까지 쭉 면접을 진행합니다.

Easy

코딩에서 어떤 부분이 당신을 가장 설레게 하는가?
새 언어를 공부할 때, 가장 먼저 해보는 일이나 프로젝트가 있나?
IT가 아닌 다른 업무를 맡은 동료와 효과적인 커뮤니케이션을 할 수 있나?

코딩에서 중요하게 생각하는 철학이나 원칙이 있나?
새 프로젝트를 시작할 때 하는 행동들이 있나?
즐겨 쓰는 라이브러리가 있나? 최근 가장 인상깊게 본 남의 코드는?

프로젝트 일정은 어떤 방식으로 세우나?
협업하면서 문제를 겪은 적이 있나? 어떻게 해결했나?
최근 개발하다가 잘못된 전제로 인해 함정에 빠진 적이 있나?

Hard

Easy

문제 상황을 자료구조로 표현해낼 수 있나?
문제 상황을 알고리즘으로 풀어낼 수 있나?
시간/공간 복잡도로 표현할 수 있나?

문제 상황을 작게 조각 낼 수 있나?
해당 기술이 전제하는 것은 무엇이 있나?
해당 코드는 어떻게 실행되는가? 어떤 예외를 예상하고 있나?

Tech Stack을 왜 그렇게 선택하였는가?
지속가능한 개발을 해보았는가?
기술적으로 올바른 선택인가?

Hard

versity

+Googliness

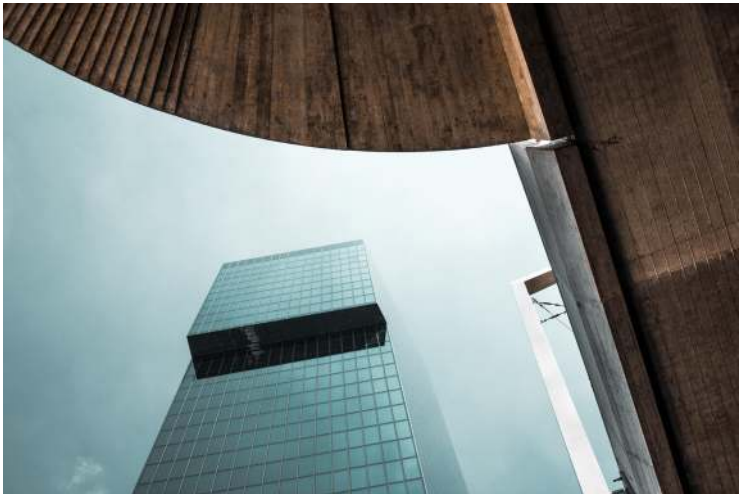
Job Title　　Job Description

Team
Position
Position Level
Location

Role
Responsibilities
Qualifications
　　Minimum
　　Preferred

모집 공고에서 보셔야 할 건, 왼쪽과 같습니다. 가장 중요한 부분은 Responsibilities와 Minimum Qualifications이 자신과 얼마나 맞는지 입니다.
예시로 Google Software Engineer (CA, Mountain View)의 JD(Job Description)를 들고 왔습니다.

Google Software Engineer (CA, Mountain View)



Responsibilities

Design, develop, test, deploy, maintain and improve software.
Manage individual project priorities, deadlines and deliverables.

Qualifications

Minimum qualifications:

BS degree in Computer Science, similar technical field of study or equivalent practical experience.
Software development experience in one or more general purpose programming languages.
Experience working with two or more from the following: web application development, Unix/Linux environments, mobile application development, distributed and parallel systems, machine learning, information retrieval, natural language processing, networking, developing large software systems, and/or security software development.
Working proficiency and communication skills in verbal and written English.

Preferred qualifications:

Master's, PhD degree, further education or experience in engineering, computer science or other technical related field.
Experience with one or more general purpose programming languages including but not limited to: Java, C/C++, C#, Objective C, Python, JavaScript, or Go.
Experience developing accessible technologies.
Interest and ability to learn other coding languages as needed.

자신에게 맞는 직업군, 포지션을 아는 것이 가장 우선입니다.
보다 자세한 모집 공고는 Google Career에서 확인이 가능합니다.
또한 관심 있는 Position과 Location을 설정하여
새로 올라오는 모집 공고를 메일로 받아보실 수도 있습니다.
아래는 세미나에서 보았던 직업군 예시입니다.

Engineering & Technology

Develop the products and tools of the future for billions of users.

Learn more See jobs

Sales, Service & Support

Equip businesses with the right tools to help them grow.

Learn more See jobs

Marketing & Communications

Connect users to the product magic

Learn more See jobs

Design

Create beautiful and intuitive experiences.

Learn more See jobs

Business Strategy

Deliver the analysis and insights we use to innovate

Learn more See jobs

Finance

Design and manage solutions to support our global business

Learn more See jobs

Legal

Shape the future of internet policy and law.

Learn more See jobs

People

Find, grow, and support the people who make Google great

Learn more See jobs

Facilities

Reimagine what it means to be at work.

Learn more See jobs

Software Engineer

Your work is at the core of everything we build. Develop massive, complex software systems that scale globally.

Product Manager

Architect the future of our products by bridging engineering and business as you manage a product's full lifecycle, from strategic planning to development and launch.

Sales Engineer

Be a trusted technical advisor to customers as you guide them through their journey to the cloud. Shape the future of our products by advocating for your customers' requirements.

Technical Program Manager

Rely on your strong technical experience to oversee all the essential activities of a particular program, including planning, communications, and execution.

Technical Solutions

Manage vital business relationships and troubleshoot complex engineering problems in this hybrid Tech/Business role.

Electrical Engineer

Design, develop, and maintain some of the world's biggest and most complex electrical control systems and equipment.

Test Engineer

Develop environments and infrastructure for testing the complex software and/or hardware systems deployed across Google.

Network Engineer

Design and implement enterprise and carrier network systems and architecture vital to Google's operations.

Sourcing/Supply Chain

Own relationships with Google's strategic internal and external partners in order to manage Google's inventory and procurement needs.

Systems Integrator

Integrate systems :)

Operations Engineer

Install, test, and maintain hardware and systems software for Google's data centers.

Corporate Operations/Audio Video Engineer

Bolster technical, front line user support for all of our internal tools and communication technologies.

Developer Relations

Use your engineering background to bridge the technical community and Google, creating programs and advocating for developers who use Google's platforms.

Mechanical Engineer

Design and analyze products and prototypes in a manufacturing and product-oriented development environment.

Security/Privacy Engineer

Hack Google... if you can. Work on finding security flaws, building secure infrastructure, or ensuring data privacy as part of a diverse engineering team.

Technical Writer

Document complexity with elegance and simplicity.

이력서는 이렇게 쓰세요

지원부터 안되면 어떻게 하나? 라고 걱정하시는 분들이 있습니다만, 절대 떨어지지 않는 이력서를 작성하시면 걱정 없습니다. 세미나 속 세미나, <Resume 작성법>입니다.

우선 자신이 지원하는 JD에 맞게 Resume를 작성해주셔야 합니다. 이는 JD에 있는 표현을 재사용하셔도 좋고, 자신이 이 자리에 맞는 사람이라는걸 표현하기만 하셔도 됩니다. JD에 관련없는 내용은 빼시는게 좋습니다.

그리고 작성된 내용이 외부에서 검증 가능해야 합니다. Linkedin에 올리는게 첫번째 방법이고, link를 포함하거나, 같이 일한 친구의 이름을 적는것도 좋습니다. 제출하기 전, 해당 link나 peer가 온라인 상태인지 꼭 확인하세요.

Work Experience나 Project를 작성할 때에는 OKR 법칙에 맞춰서 기록해주세요. 자신이 어떠한 일을 했다는 단편적인 사실보다는, 자신이 어떤 일을 하고 싶었고, 이를 어떻게 진행했으며, 얼마나 달성하였는지를 수량화하여 기록해 주셔야 합니다.

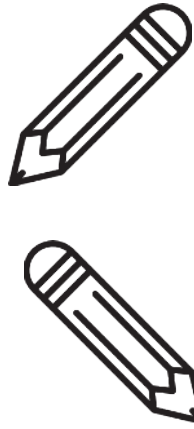
예시) A/B/C 환경에서 D 웹서비스를 만들어, N 사용자와 M 동접을 확보했으며, 어떤 기능을 제공하여 P만큼의 사용자 만족을 끌어냈습니다.

이런 내용이 담긴 Resume를 통해, 면접관이 나를 궁금해하게 만들고 면접에서 나오는 질문을 내 편으로 만들 수 있습니다.

Resume

Education
Work Experiences
Extra Activities
- **Project**
- **Awards**

Skills & Others



+Resume

Related with JD
Publicly Accessible
- **Peer Review**

OKR
- **Objective**
- **Key Results**

Ready to interview

Example) OKR=Measurability

정보보호팀에서 관제업무를 맡았다. → **정보보호팀에서 INFOCON을 위한 관제를 주로하고,**

10 + Accidents/Month → 1 + Accidents/Month
5 + Scopes: Firewall, IPS, UTM, NAC, Agencies
3 + Networks: Intranet, Internet, Mobile

침해사고가 관제될 시 대응을 하며,

Average Response Time: 1 week → 0 day

피해보고서를 작성하였다.

Average Report Accuracy: 80% → 95%

이력서는 면접의 중요한 자료가 된다. 이날 양민호 강사는 “이력서에서 소개된 Work Experience나 Project의 Background, Problem Set, Technical Choice, Technical Challenging, Result 등을 정리하는 것이 면접질문 대비에 도움이 될 것이다”라고 말했다.