정보시각화

What is information visualization?

데이터그램

2015. 11. 10.

@nacyo_t

낚옷

가	각	간	갓
나	낙	난	낫
다	닥	단	닷
ㄱ	ㄴ	ㅅ	

(ㄱ,ㄴ,ㅅ)×(가,나,다)

SMARTSTUDY

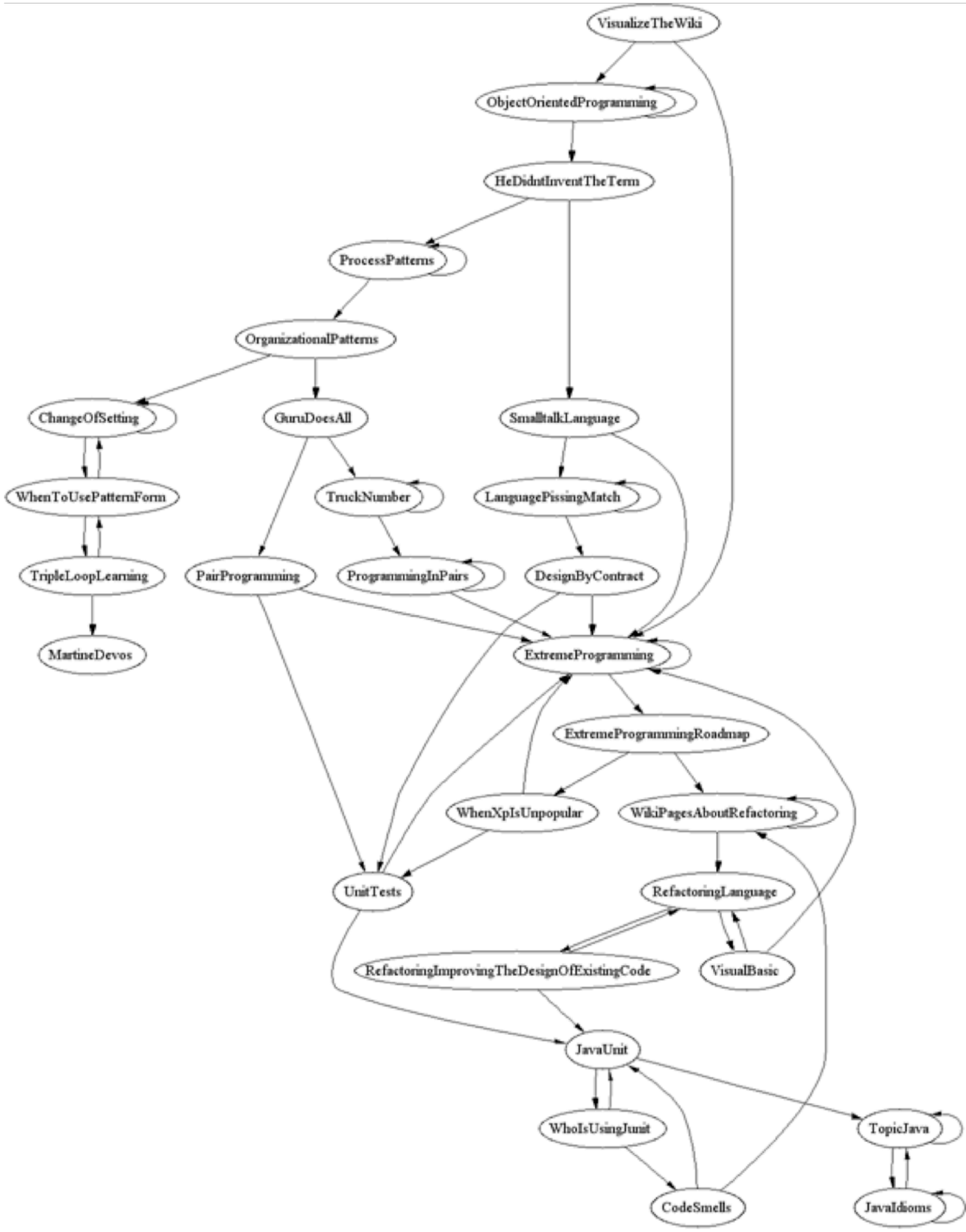
Programmer

blog.nacyot.com

I ♥ Infoviz

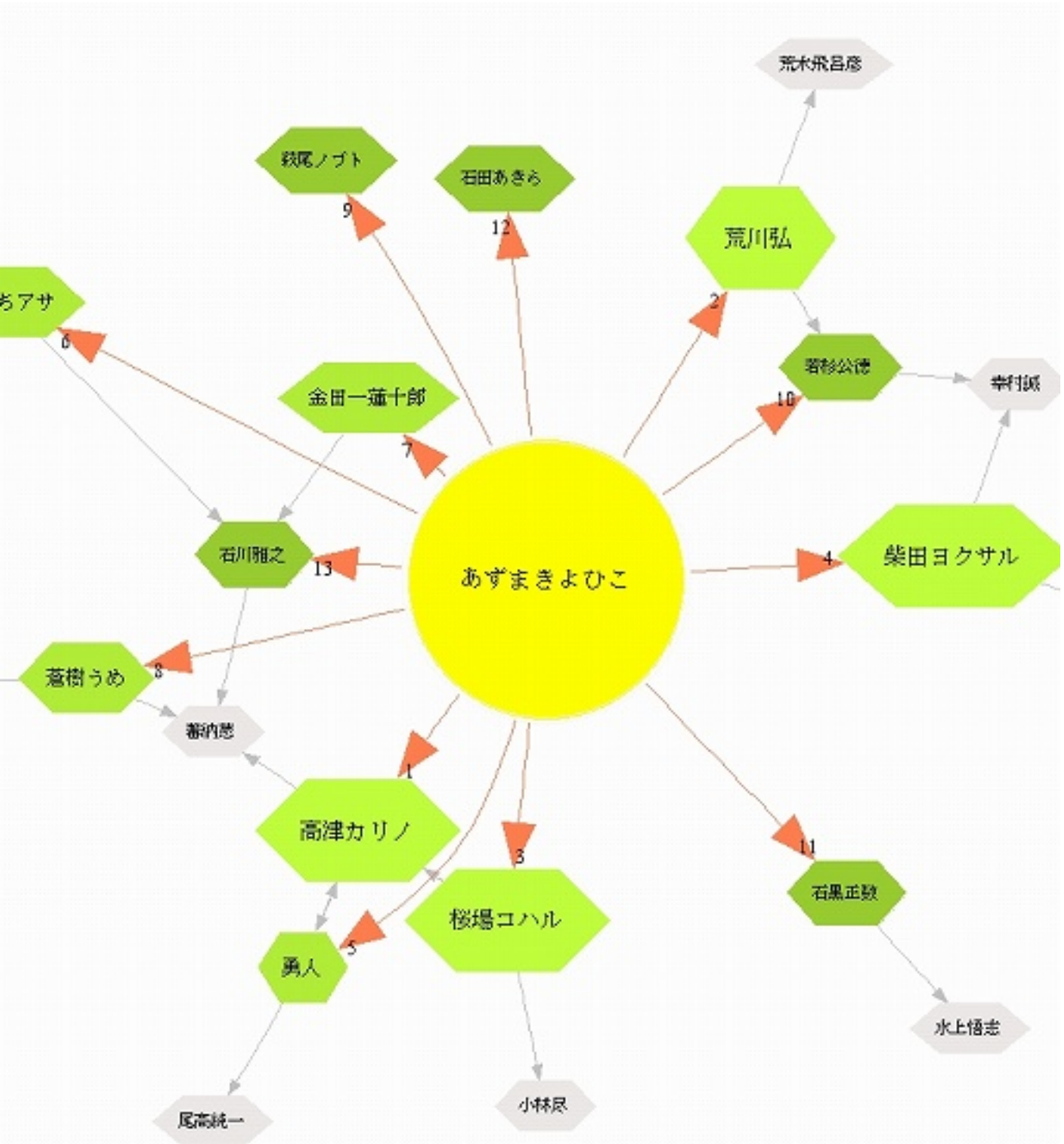
First Impact

위키 항목들 간의 연결 관계



WikiWiki

- c2.com - Graph Viz
- moinmoin
- moniwiki



만화가 지도(2007)
설문조사 기반
같이 언급된 만화가
Graphviz
HTML - 하이퍼링크

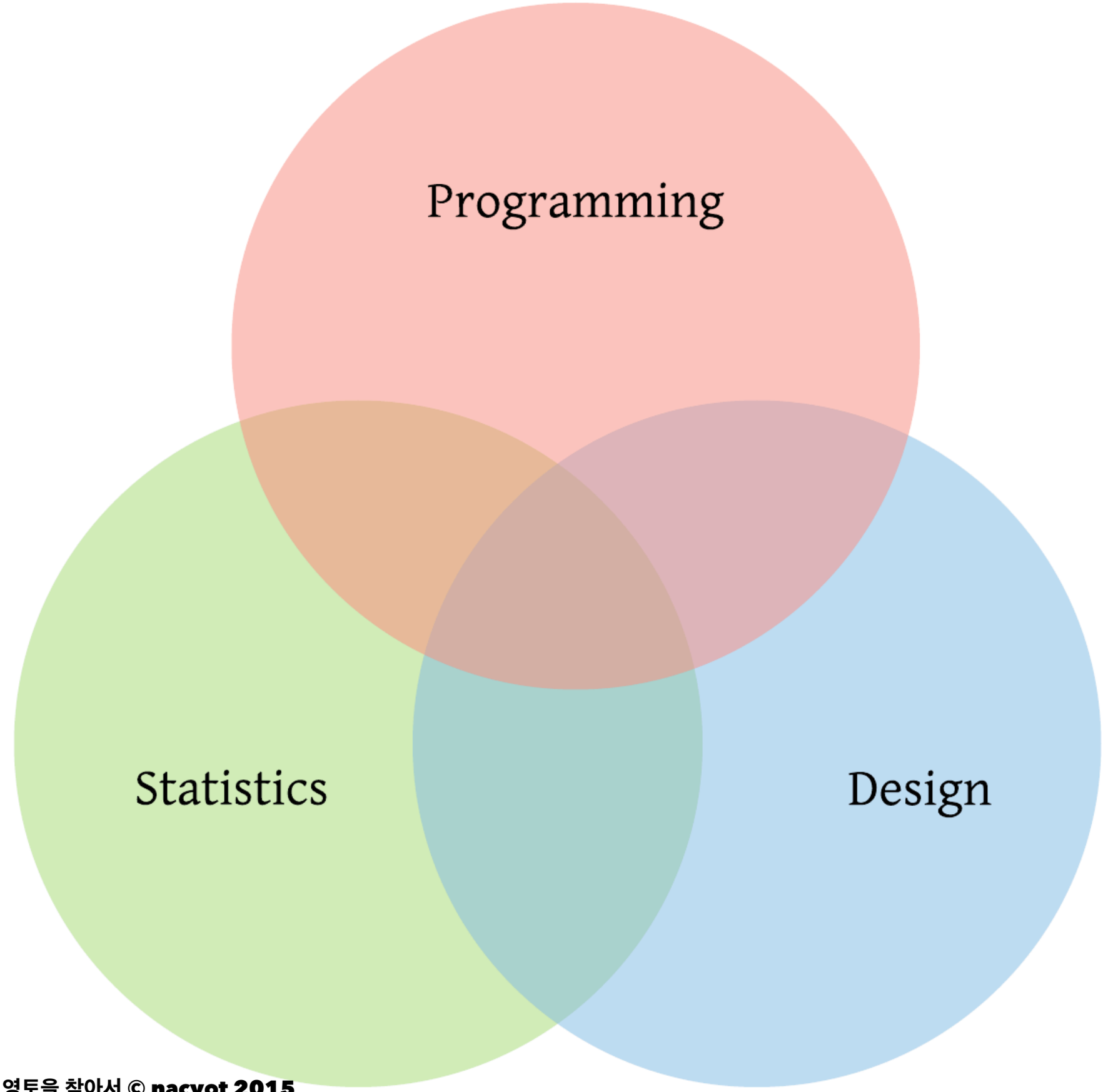
what is

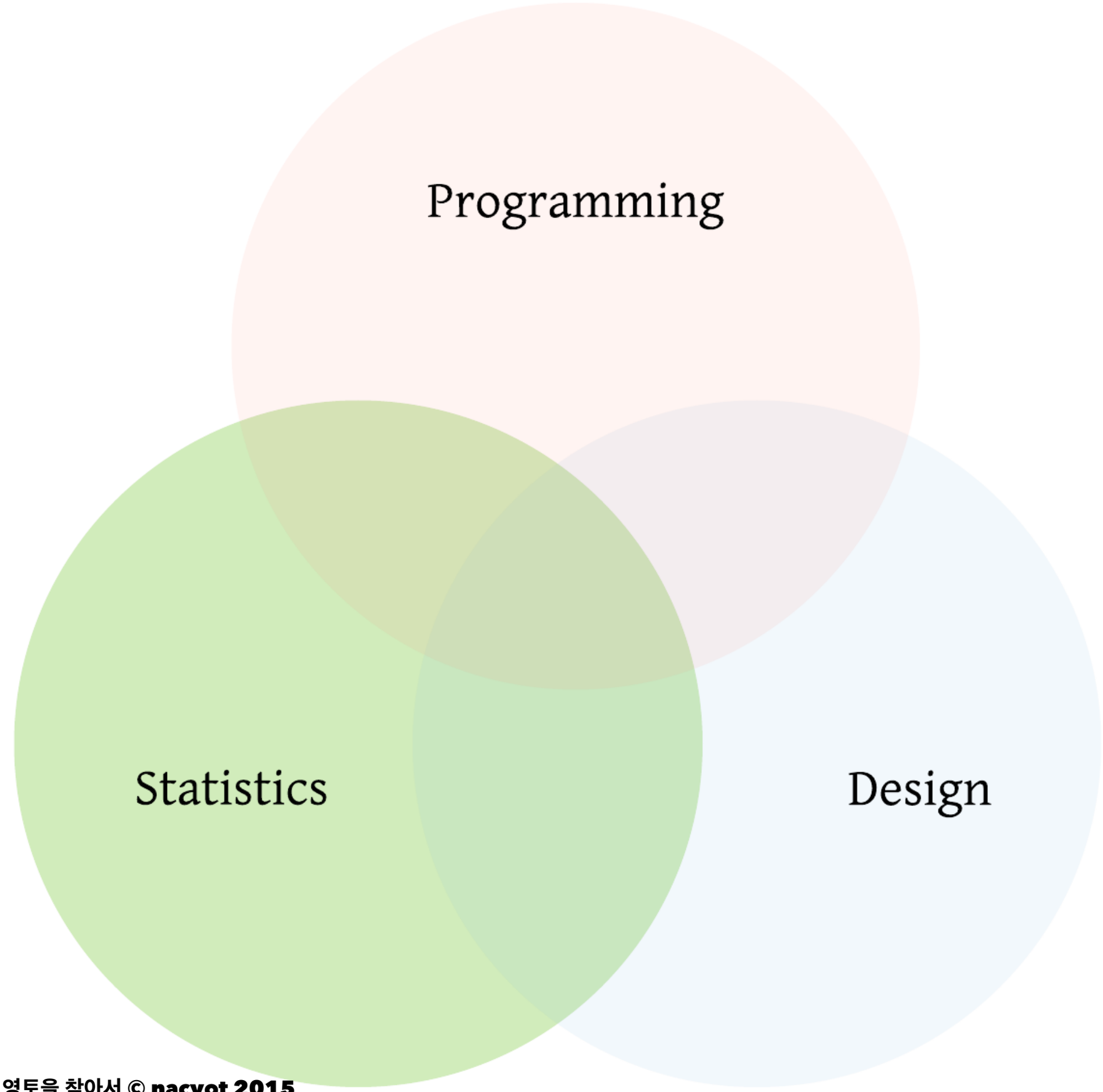
Information Visualization

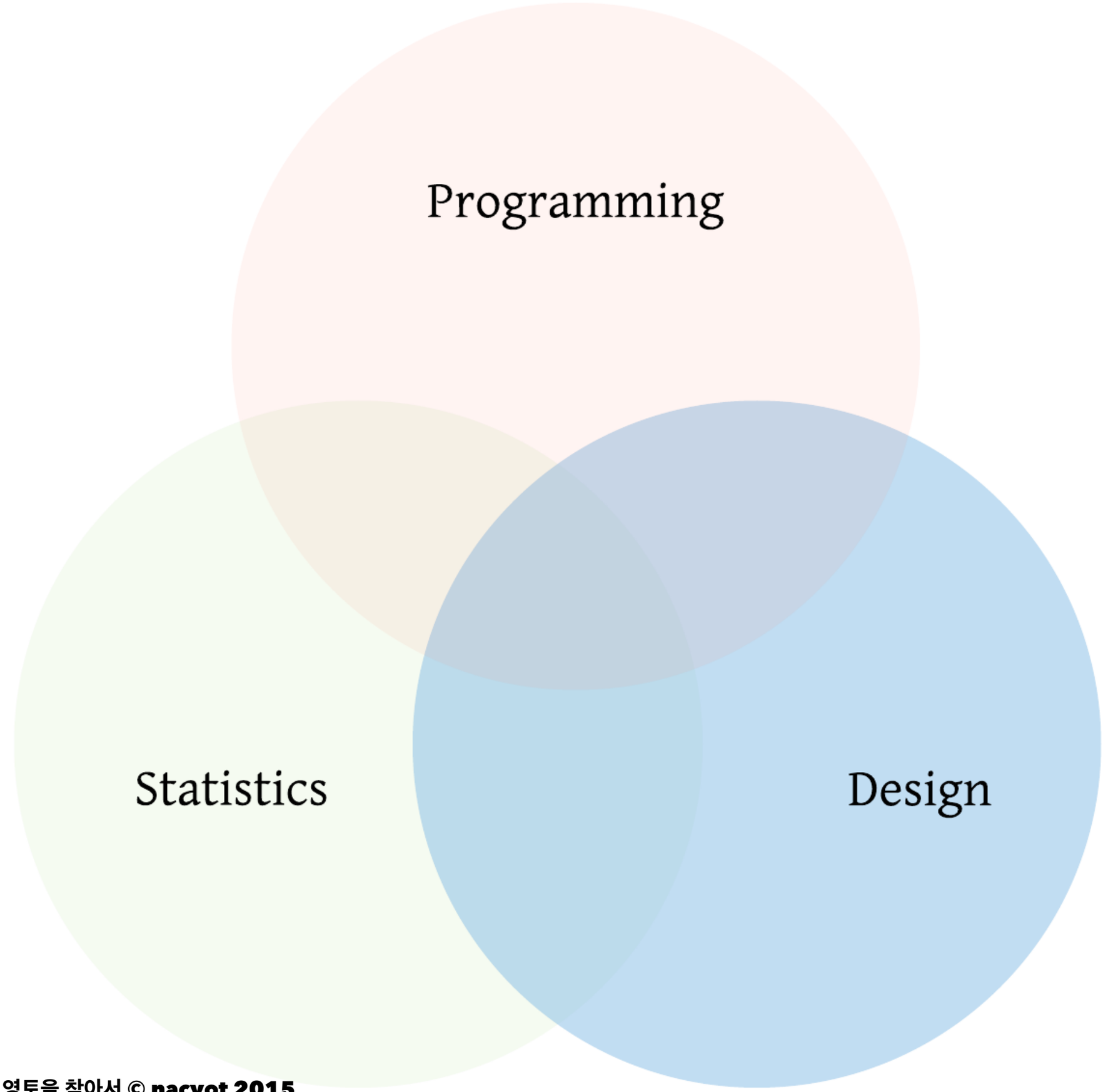
?

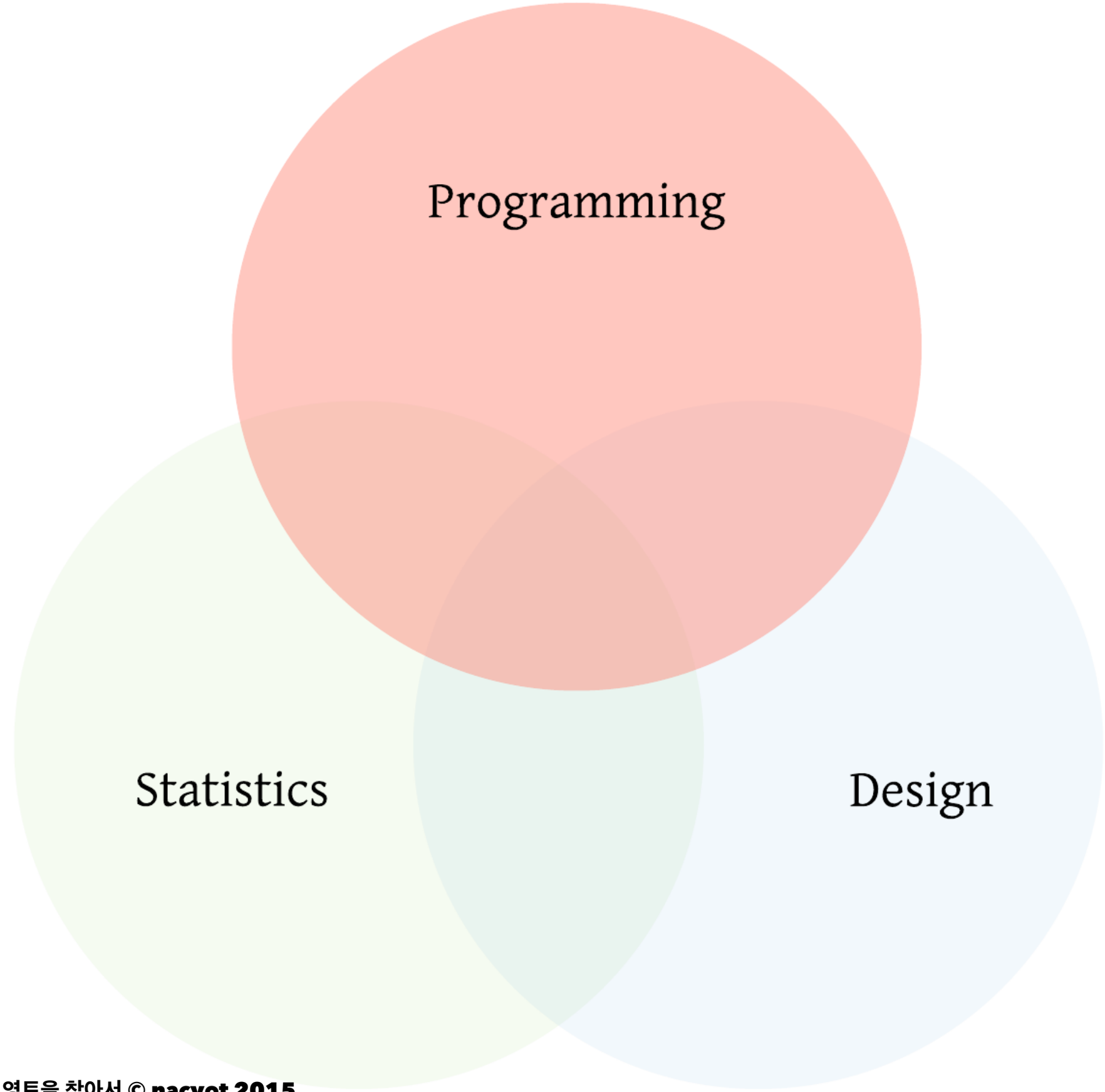
정보시각화 고유의 영토을 찾아서

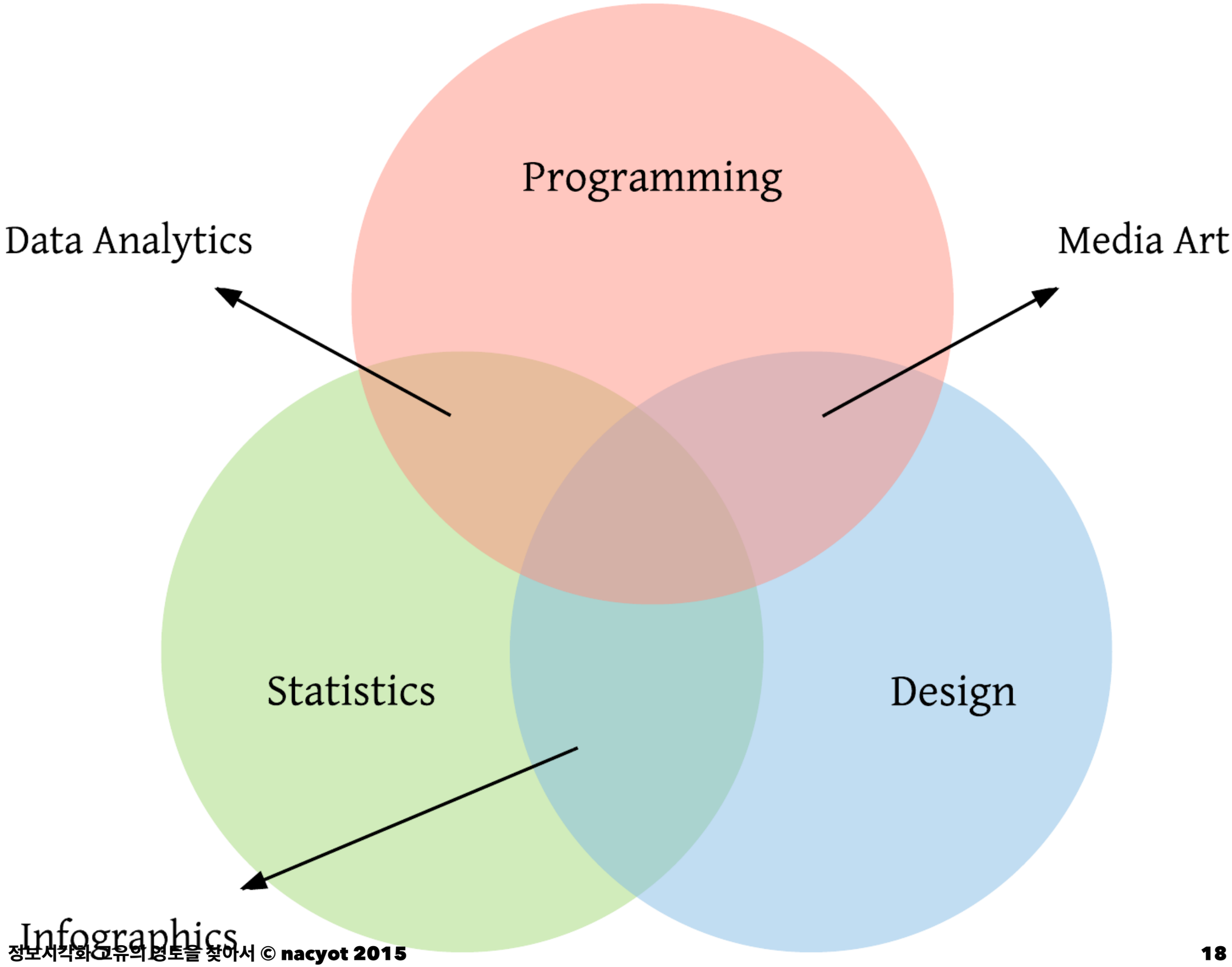
Positioning

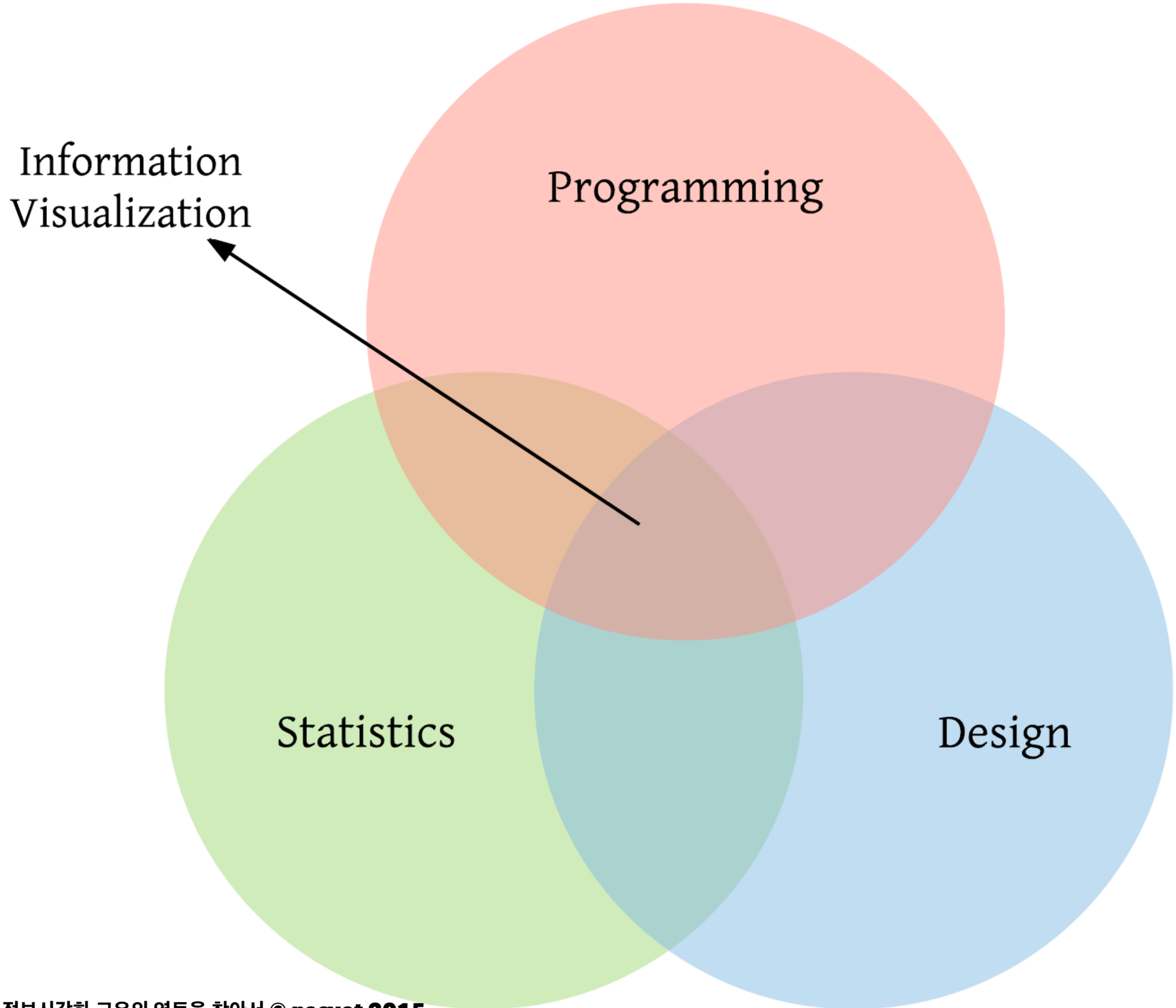

Programming
Statistics
Design

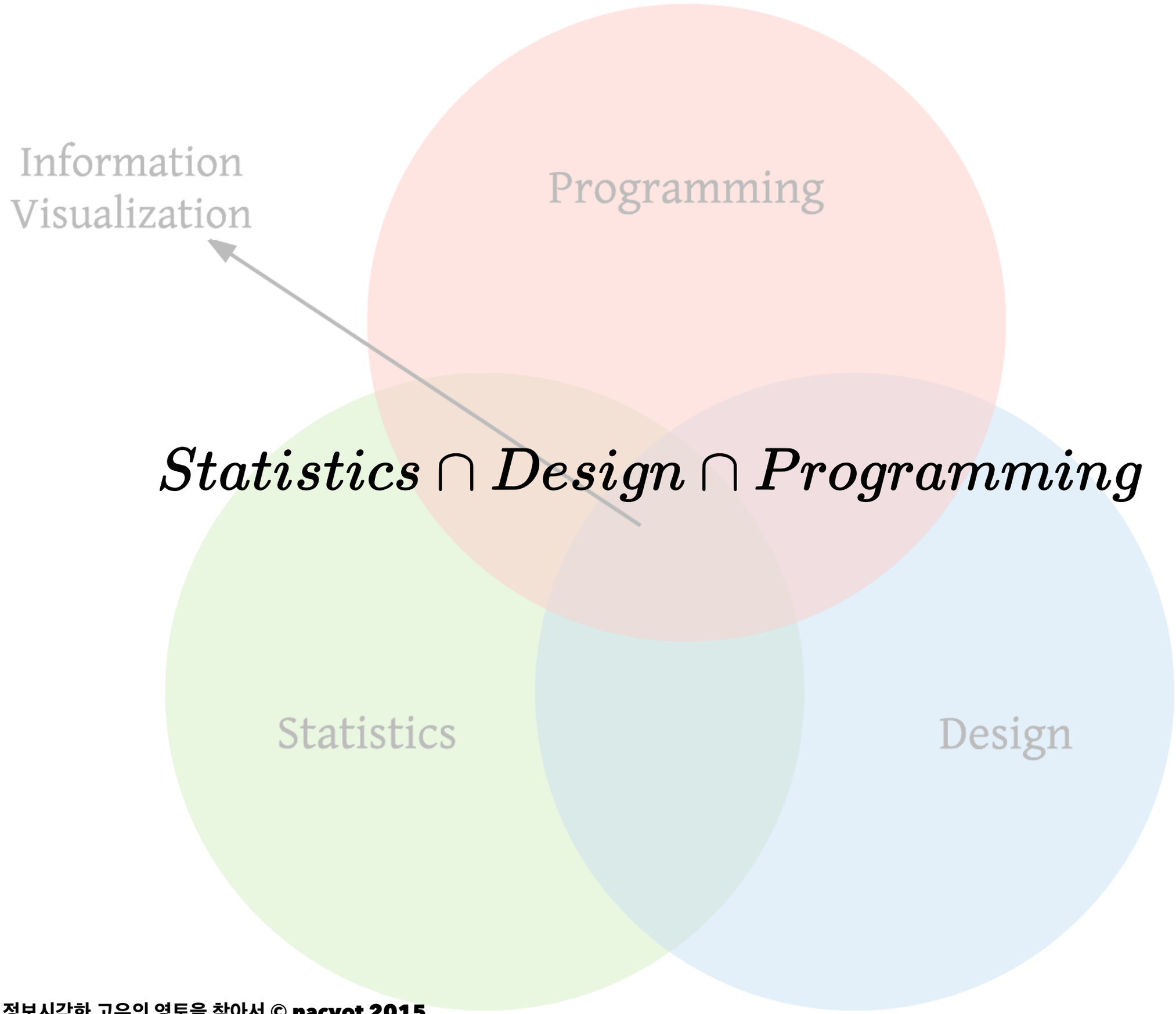

Programming
Statistics
Design


Programming
Statistics
Design

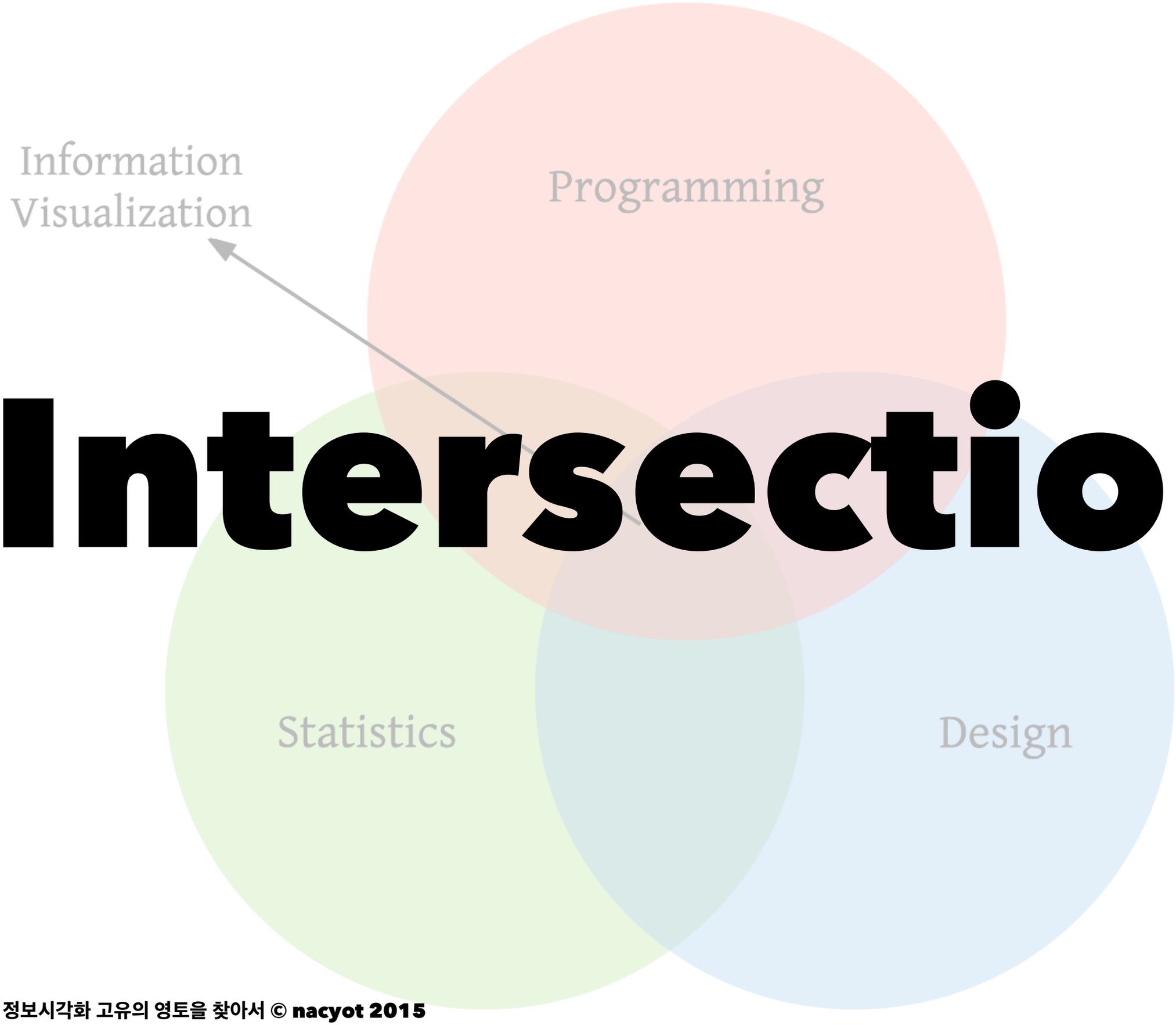

Programming
Statistics
Design


Programming
Data Analytics
Media Art
Statistics
Design
Infographics

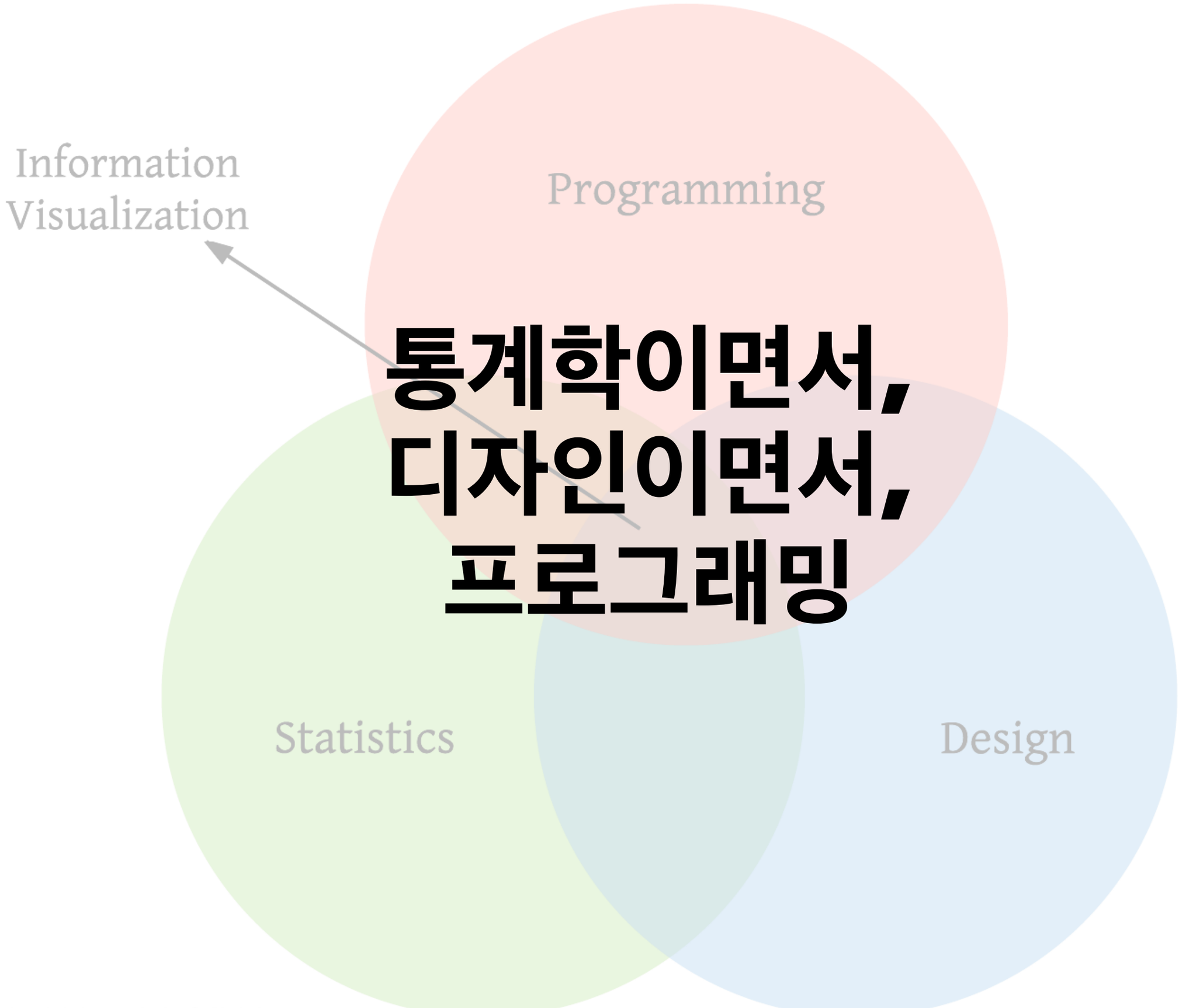

Information
Visualization
Programming
Statistics
Design



$$Statistics \cap Design \cap Programming$$



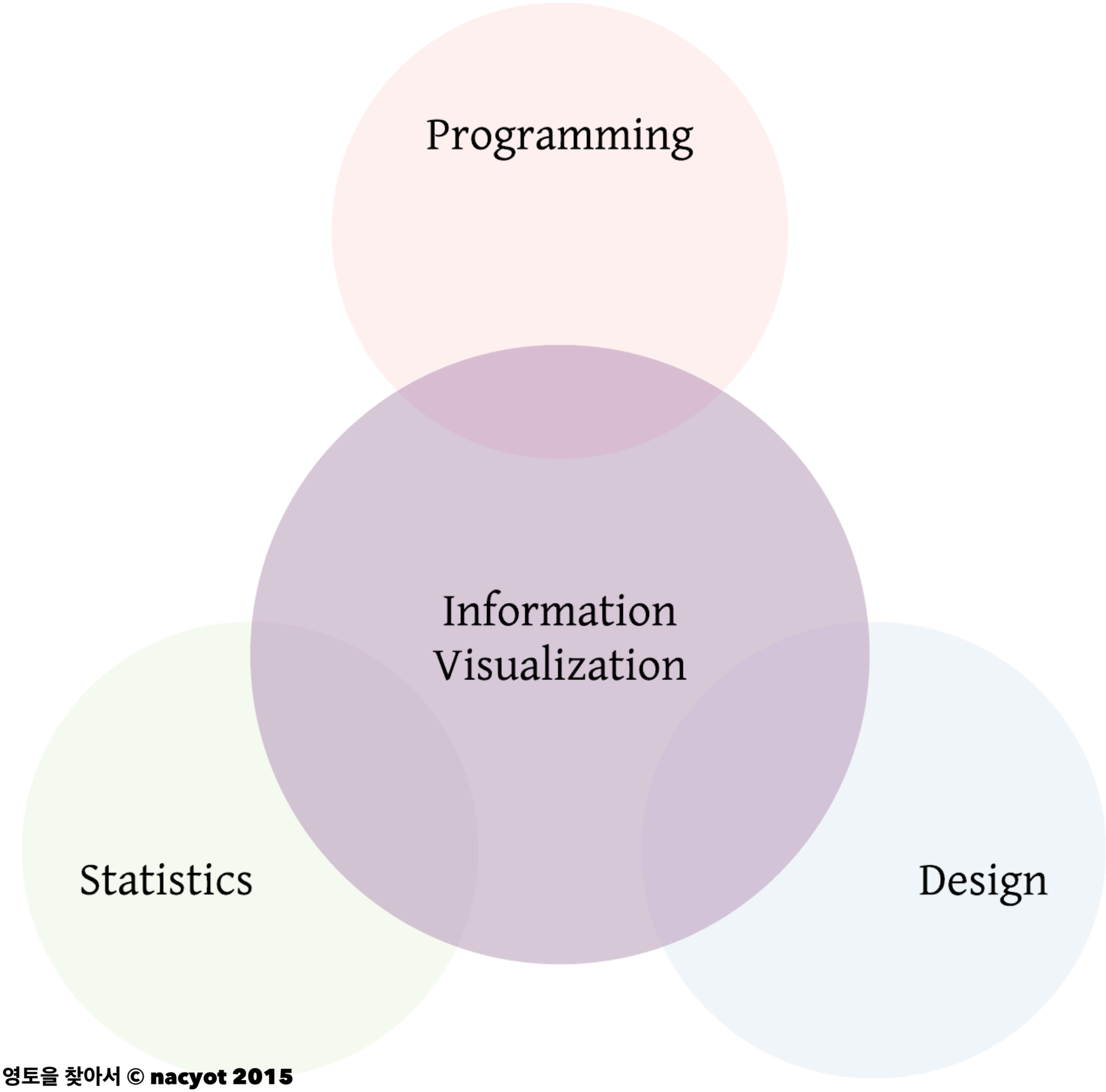
Intersection

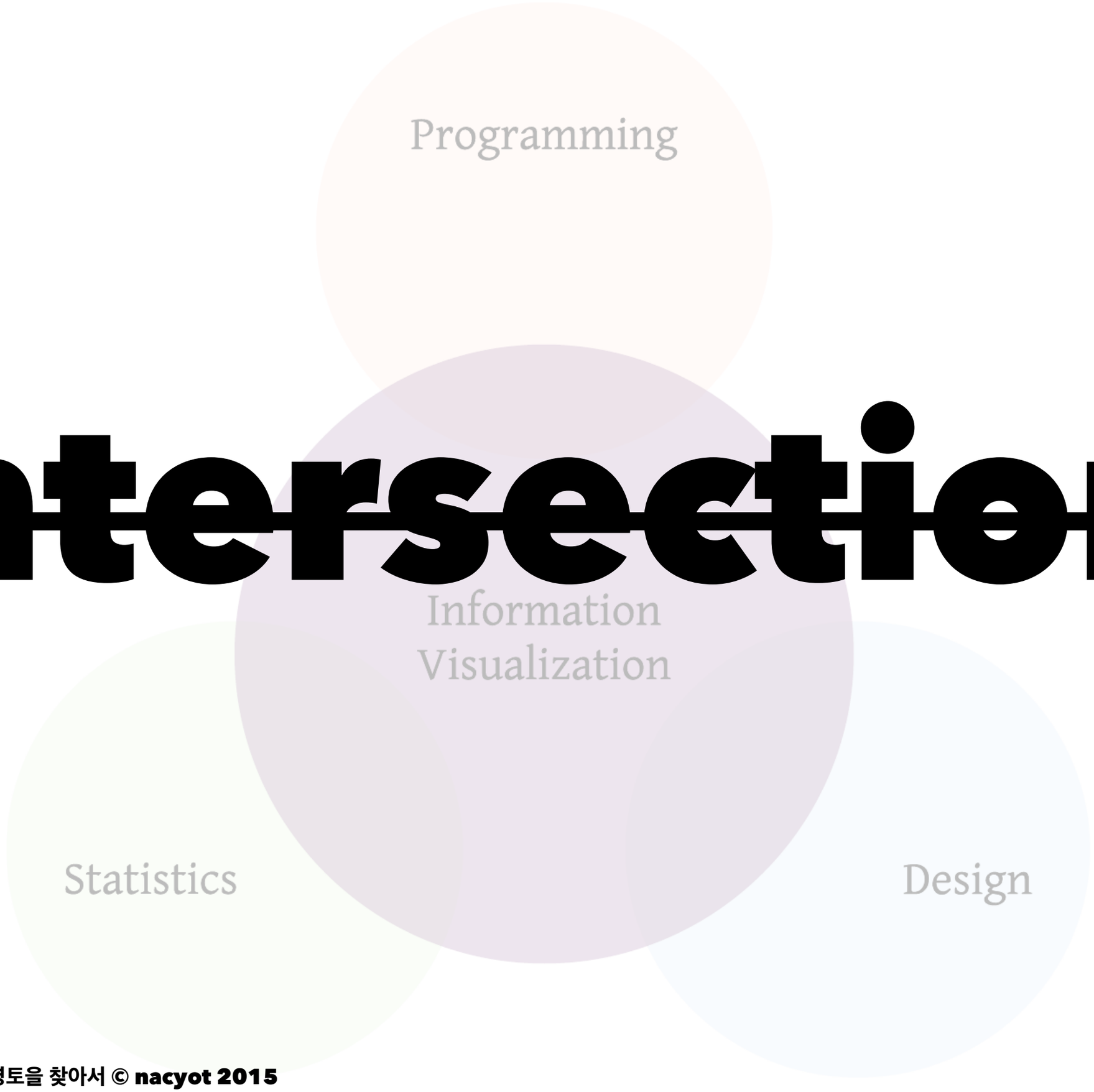


통계학이면서, 디자인이면서, 프로그래밍

고유의 영토은 어디에...

다르게 위치짓기


Programming
Information
Visualization
Statistics
Design



Intersection

Programming
통계학이면서,
디자인이면서,
프로그래밍
Information
Visualization
Statistics
Design

Programming
Visualization
itself
Information
Visualization
Statistics
Design

오늘의 이야기

Definition

Information Visualization

Information visualization or information visualisation is the study of (interactive) visual representations of abstract data to reinforce human cognition. The abstract data include both numerical and non-numerical data, such as text and geographic information

Wikipedia - Information Visualization

정보시각화

정보시각화는 사람의 인지를 확장할 수 있게 해주는 추상적 데이터의 (인터렉티브한) 시각적 표현에 대한 연구이다. 이 때 추상적 데이터는 수치적인 데이터와 텍스트나 지리정보와 같은 비수치적 데이터를 모두 포함한다.

정보시각화

정보시각화는 사람의 인지를 확장할 수 있게 해주는 **추상적 데이터**의 (인터렉티브한) **시각적 표현**에 대한 연구이다. 이 때 추상적 데이터는 수치적인 데이터와 텍스트나 지리정보와 같은 비수치적 데이터를 모두 포함한다.

정보시각화

시각적 표현
추상적 데이터

시각적 표현

회화

더 익숙하고
일반적인 것

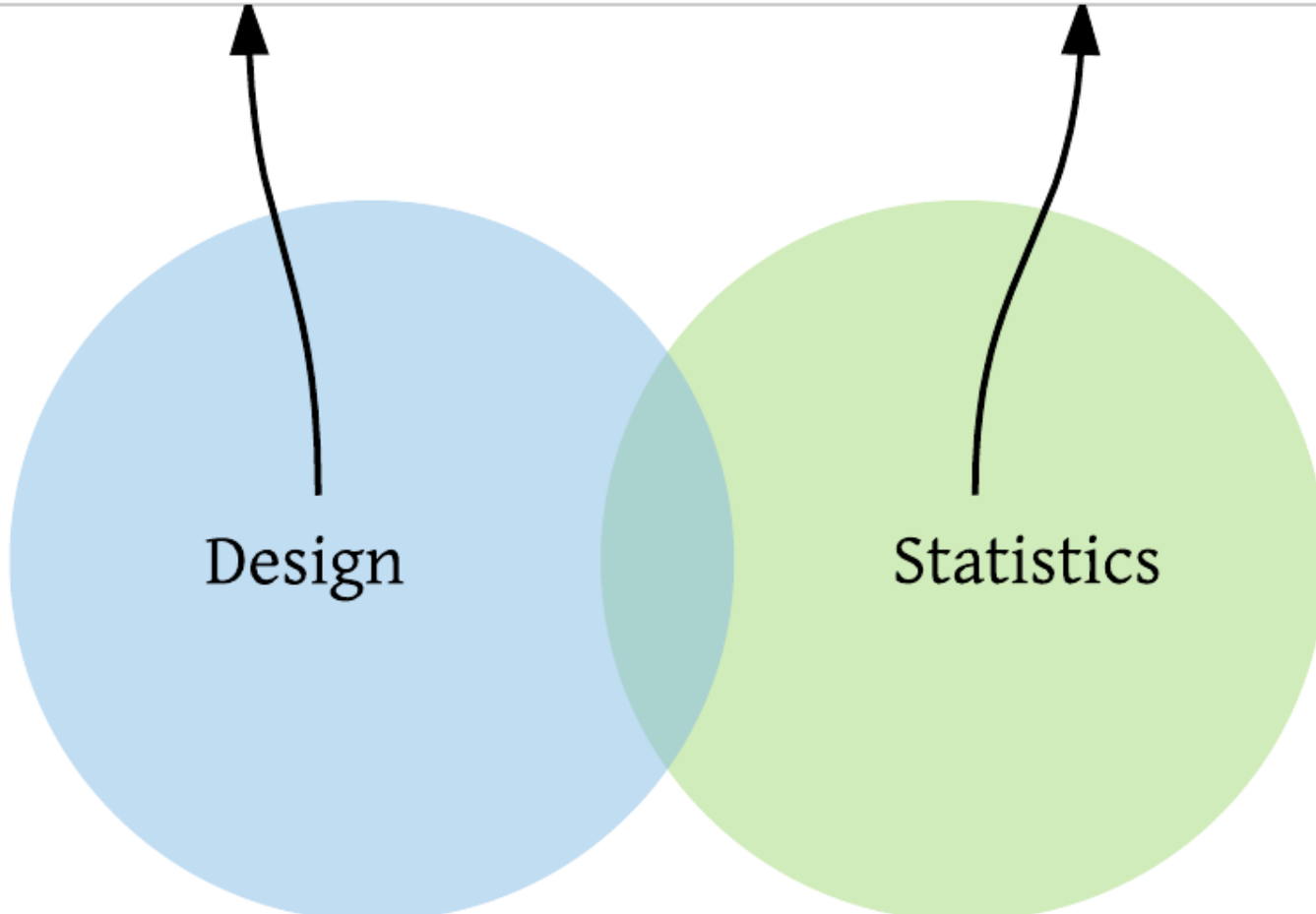
회화와 시각화의 차이

추상적 데이터

	공통점	차이점
	표현 방법	**대상**
회화	시각적 표현	정물, 풍경, 인물, 상상, ...
정보 시각화	시각적 표현	추상적 데이터

공통점 차이점

	표현 방법	대상
회화	시각적 표현	정물, 풍경, 인물, 상상, ...
정보 시각화	시각적 표현	추상적 데이터



$Statistics \cap Design$

여전히 교집합

정교하게 경계짓기

회화

그랑드 자트 섬의 일요일 오후

Un dimanche après-midi à l'Île de la Grande Jatte

207 * 308cm

1884-1886

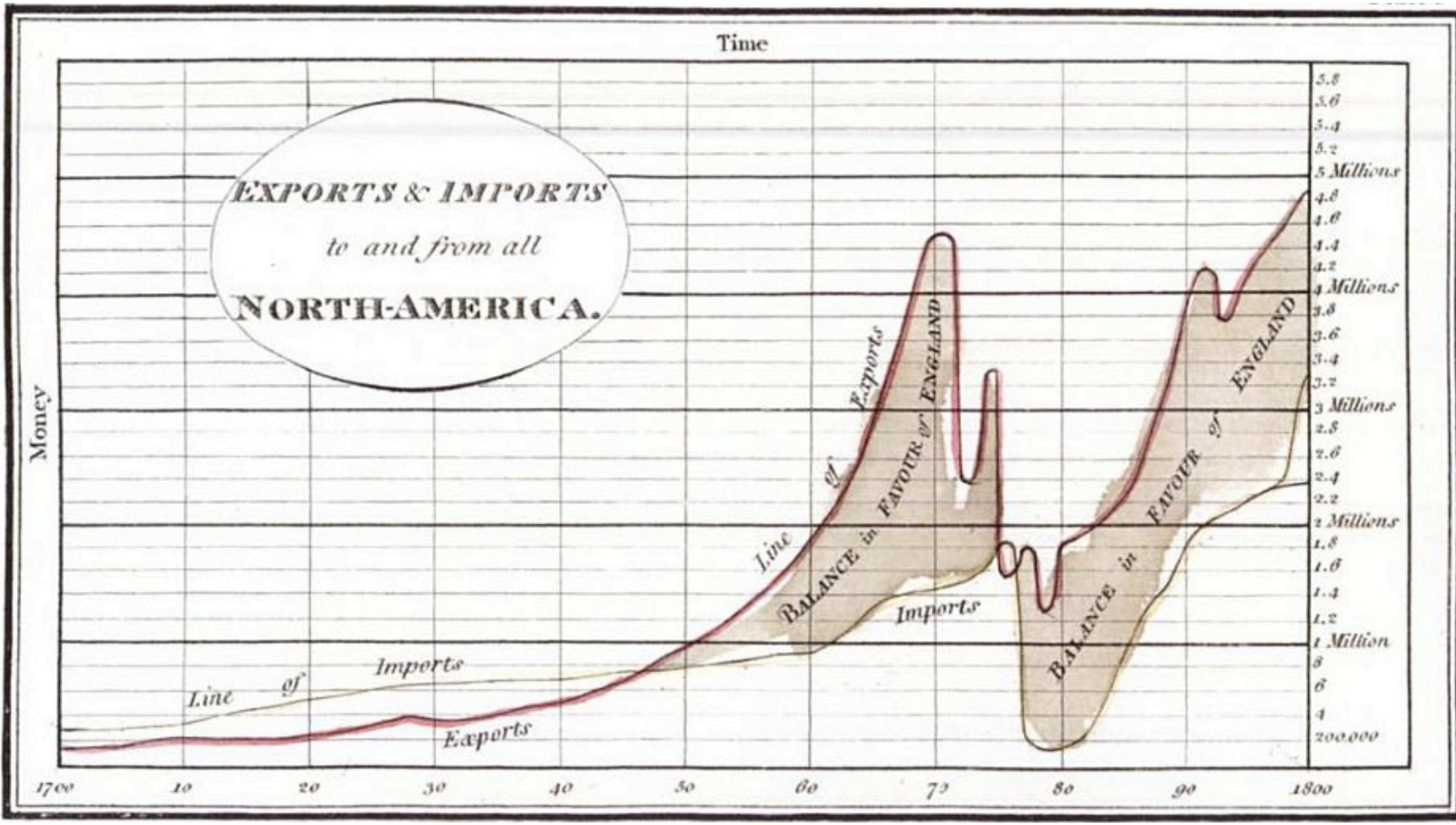
Pointilism

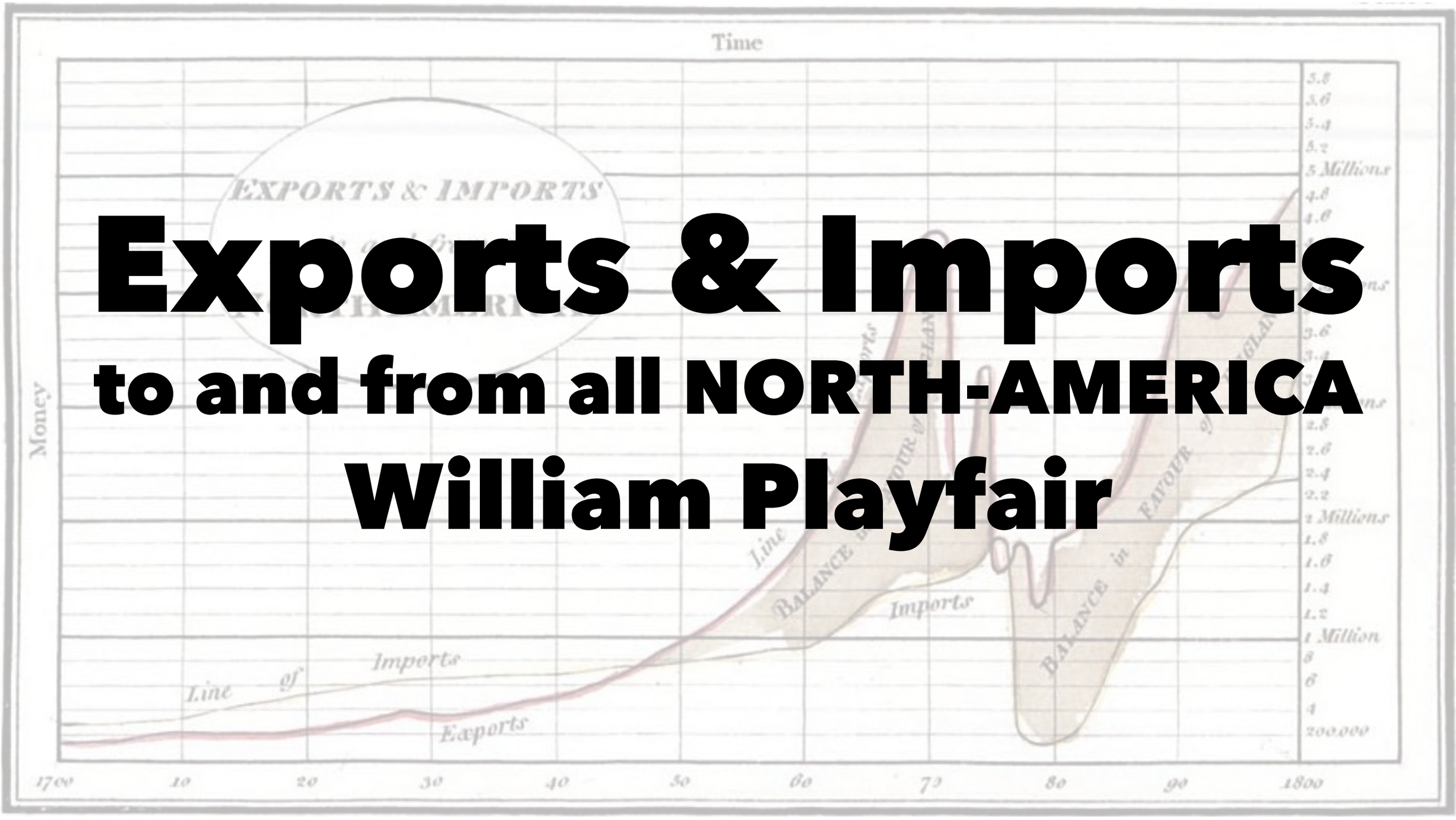
무수히 많은 점들

회화의 본질

verb. 그리다

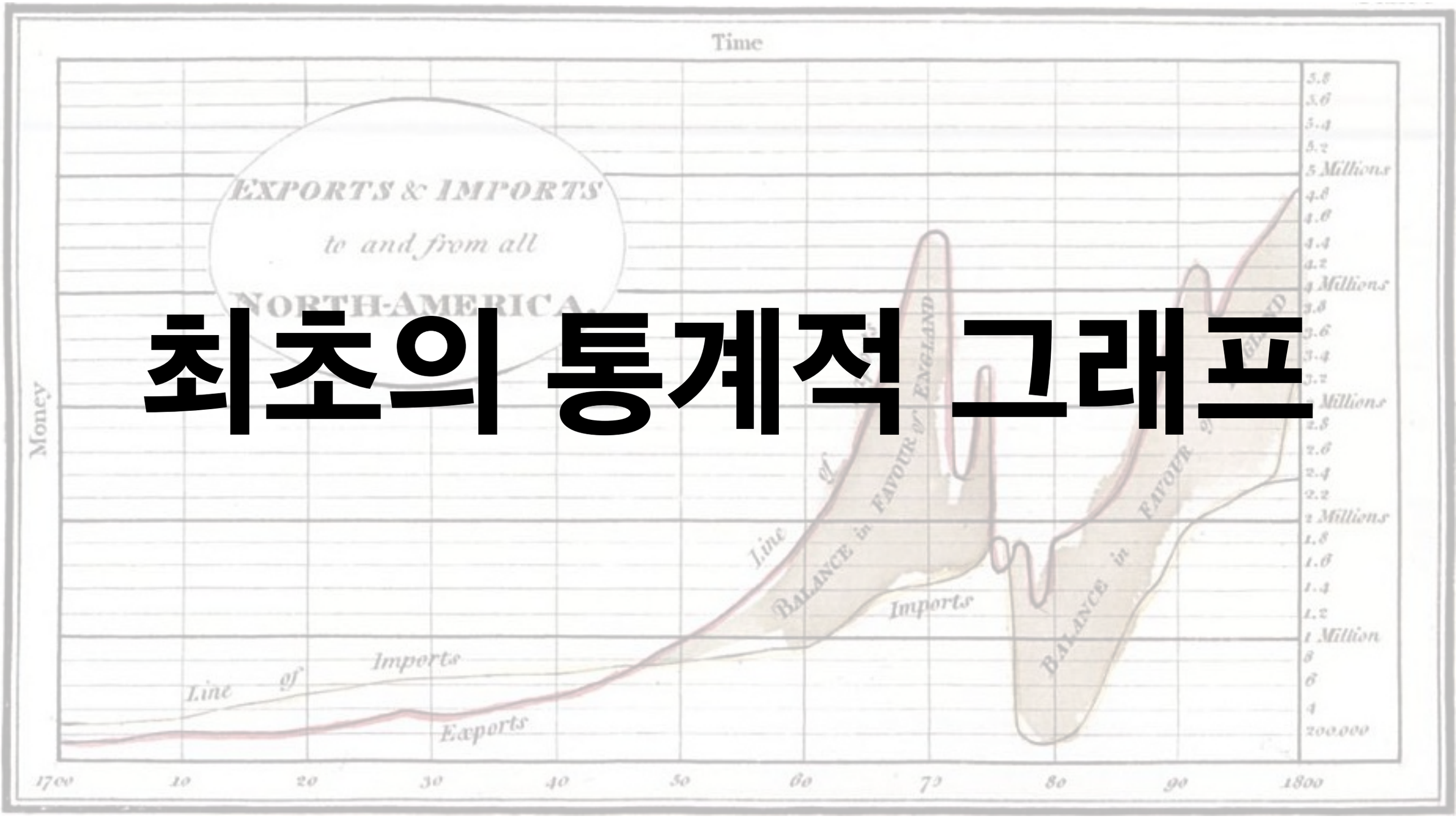
정보시각화


EXPORTS & IMPORTS
to and from all
NORTH-AMERICA.
Time
Money
Line of Imports
Line of Exports
BALANCE in FAVOUR of ENGLAND
Exports
Imports
BALANCE in FAVOUR of ENGLAND
5.8
5.6
5.4
5.2
5 Millions
4.8
4.6
4.4
4.2
4 Millions
3.8
3.6
3.4
3.2
3 Millions
2.8
2.6
2.4
2.2
2 Millions
1.8
1.6
1.4
1.2
1 Million
8
6
4
200.000
1700
10
20
30
40
50
60
70
80
90
1800

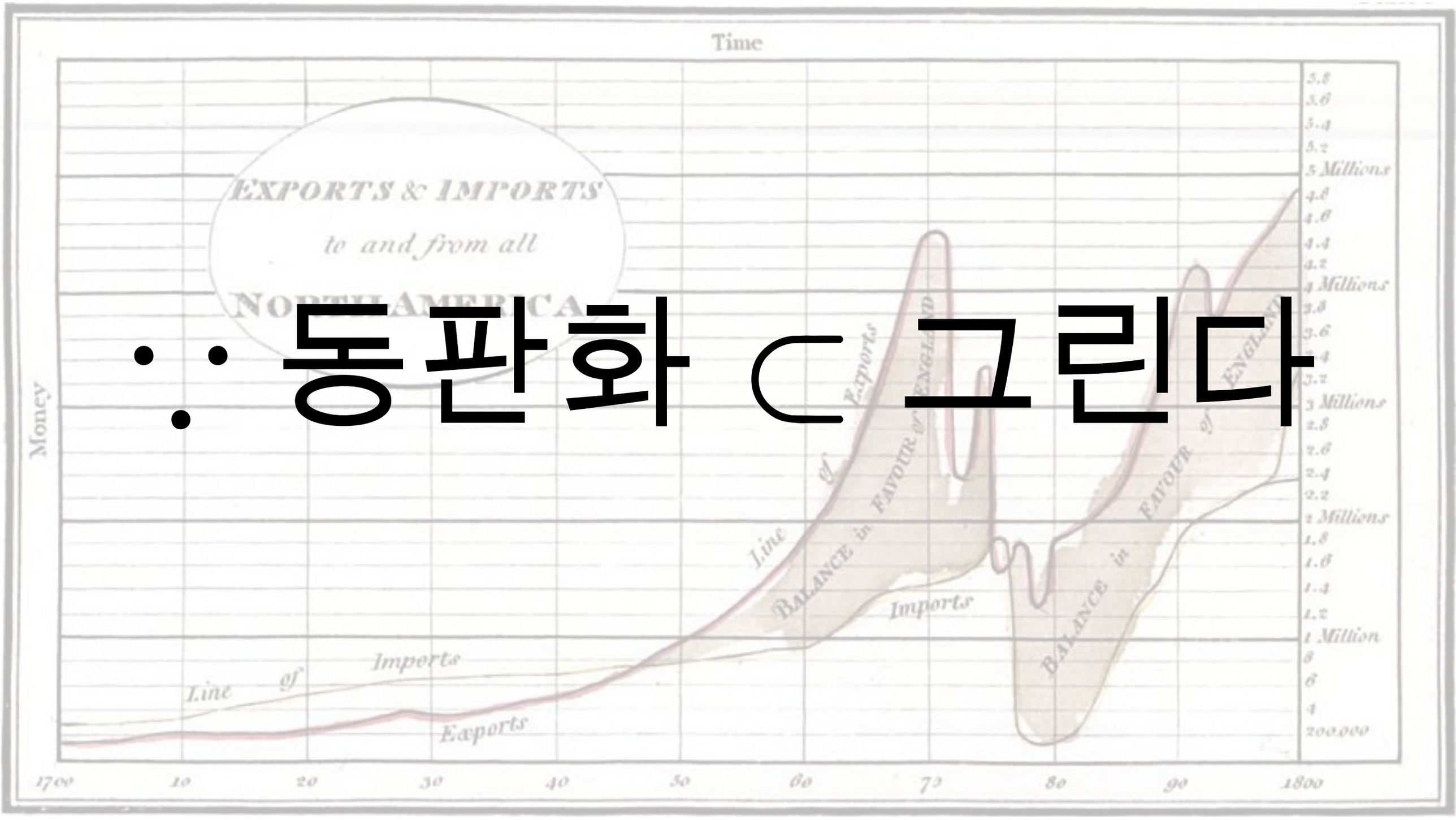


Exports & Imports
to and from all NORTH-AMERICA
William Playfair

최초의 통계적 그래프

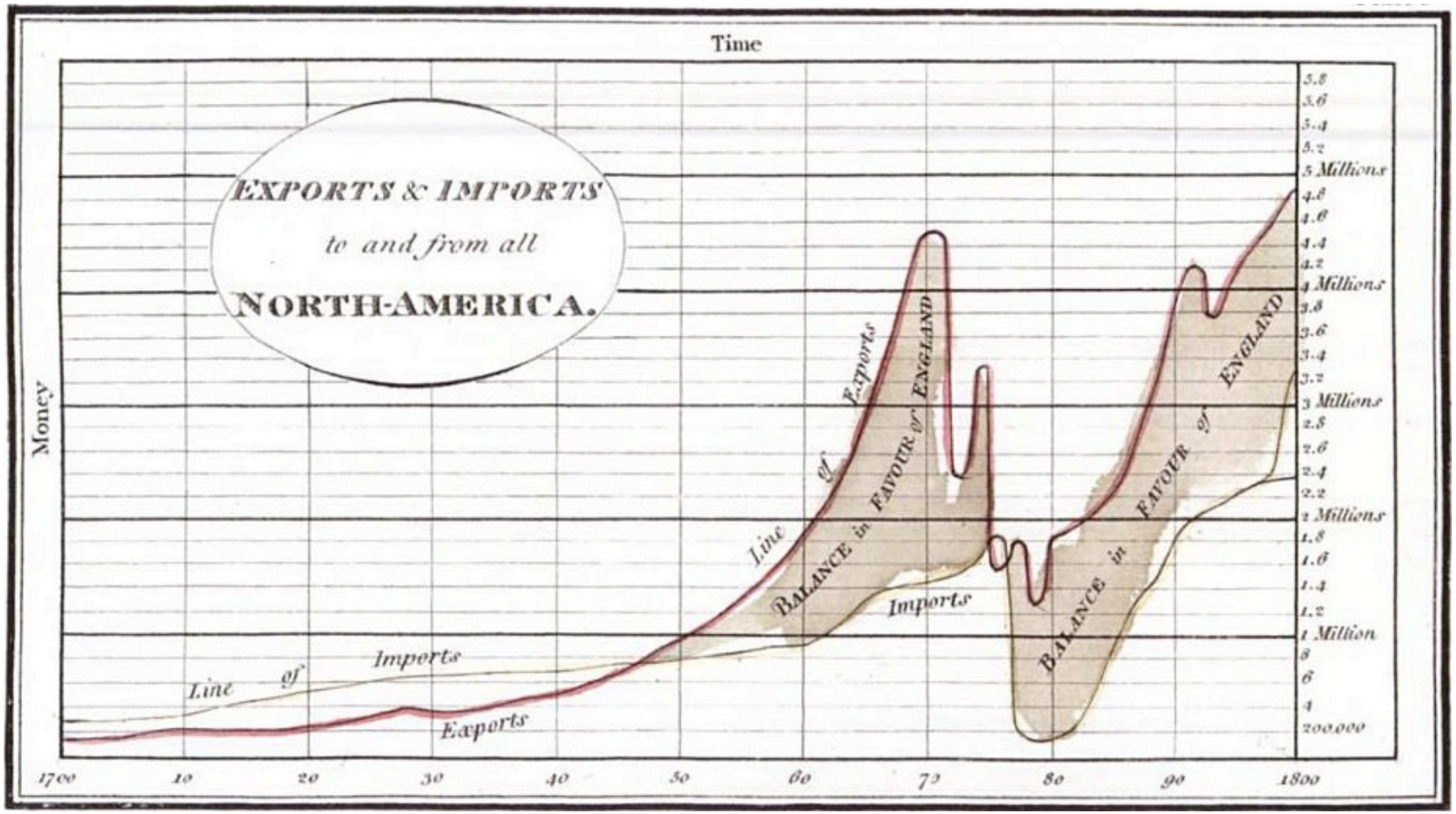

Time
EXPORTS & IMPORTS
to and from all
NORTH-AMERICA.
Money
Line of Imports
Line of Exports
BALANCE in FAVOUR of ENGLAND
Imports
BALANCE in FAVOUR
5.8
5.6
5.4
5.2
5 Millions
4.8
4.6
4.4
4.2
4 Millions
3.8
3.6
3.4
3.2
3 Millions
2.8
2.6
2.4
2.2
2 Millions
1.8
1.6
1.4
1.2
1 Million
8
6
4
200,000
1700
10
20
30
40
50
60
70
80
90
1800

마치 회화처럼 보임
시각적 표현이라는 특징을 공유하기 때문


Time
EXPORTS & IMPORTS
to and from all
NORTH AMERICA
Money
Line of Imports
Line of Exports
BALANCE in FAVOUR of ENGLAND
Imports
BALANCE in FAVOUR of ENGLAND
Exports
5.8
5.6
5.4
5.2
5 Millions
4.8
4.6
4.4
4.2
4 Millions
3.8
3.6
3.4
3.2
3 Millions
2.8
2.4
2.2
2 Millions
1.8
1.6
1.4
1.2
1 Million
8
6
4
200.000
1700
10
20
30
40
50
60
70
80
90
1800

∵ 동판화 ⊂ 그린다

Ceci n'est pas une pipe.
Magritte



Ceci n'est pas une peinture

$$Visualization \neq Painting$$

$$\because Abstract\ Data$$

회화

정물, 풍경, 인물, 상상, …

왜 추상적 데이터만 특별한가?

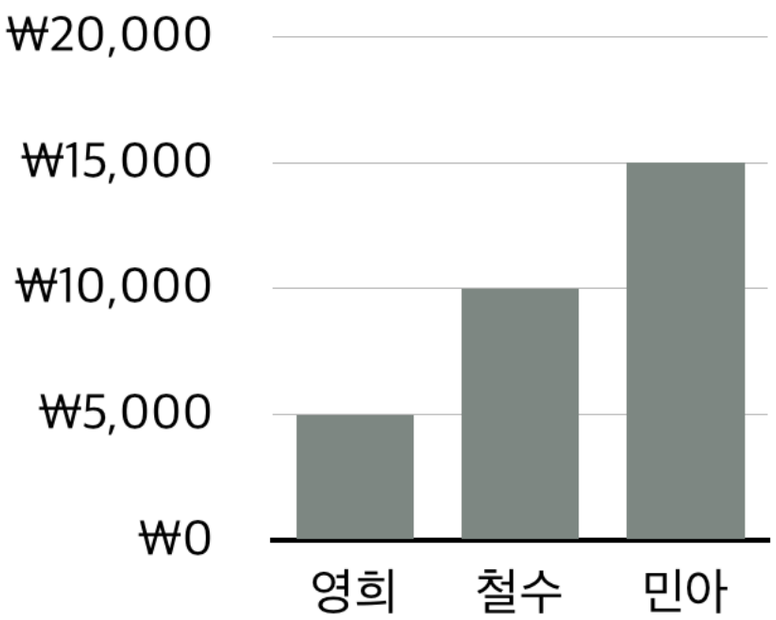
통계적 그래프의 자동성

통계적 그래프

반드시 데이터를 표현

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



통계적 그래프의 자동성

We cannot change the location of a point or the color of an object (assuming these are data-representing attributes) without lying about our data and violating the purpose of the statistical graphic to represent data accurately and appropriately. Consequently, the core of a graphics system must rest on specification.

– The Grammar of Graphics 6p

통계적 그래프의 자동성

우리는 기저의 데이터를 변경시키거나 데이터를 정확하고 적절하게 표현해야한다는 통계적 그래픽의 목적을 배반하지 않는 한, 점들의 위치나 객체의 색(이러한 요소들은 데이터를 표현하는 속성들이라는 가정하에)들을 바꿀 수 없다. 따라서 그래픽 시스템의 핵심은 스펙(*specification*)에 있다.

– The Grammar of Graphics 6p

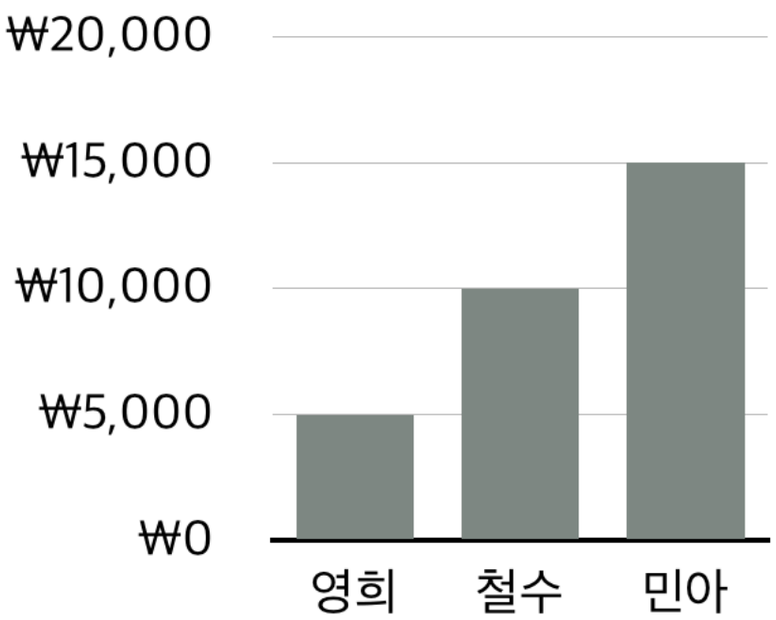
통계적 그래프를 변화시키는 방법

1. 기저의 데이터를 변경
2. 통계적 그래픽의 목적을 배반한다

기저의 데이터 변경

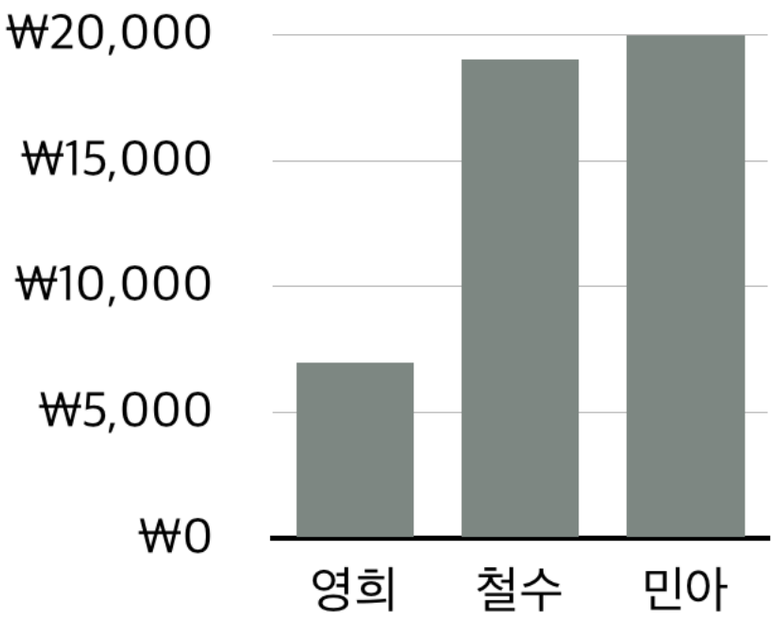
용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



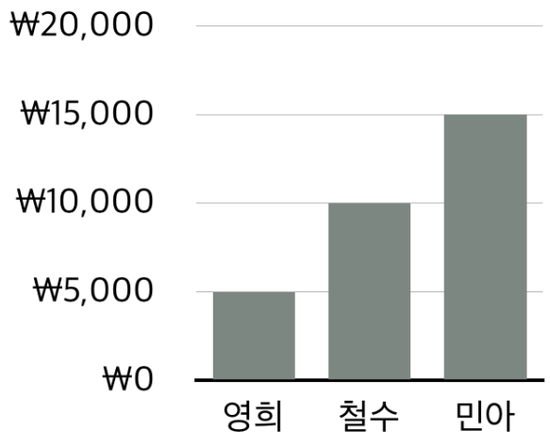
용돈

	용돈
영희	₩7,000
철수	₩19,000
민아	₩20,000



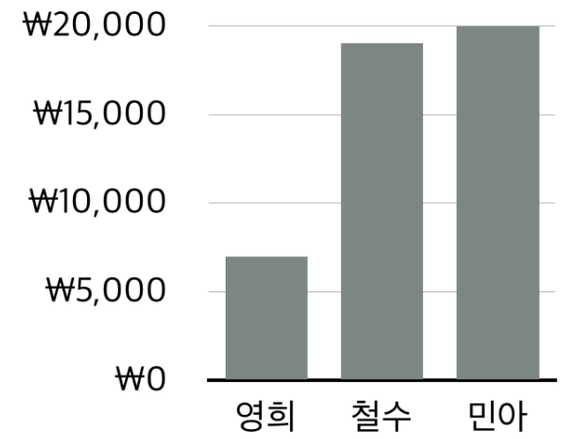
용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



용돈

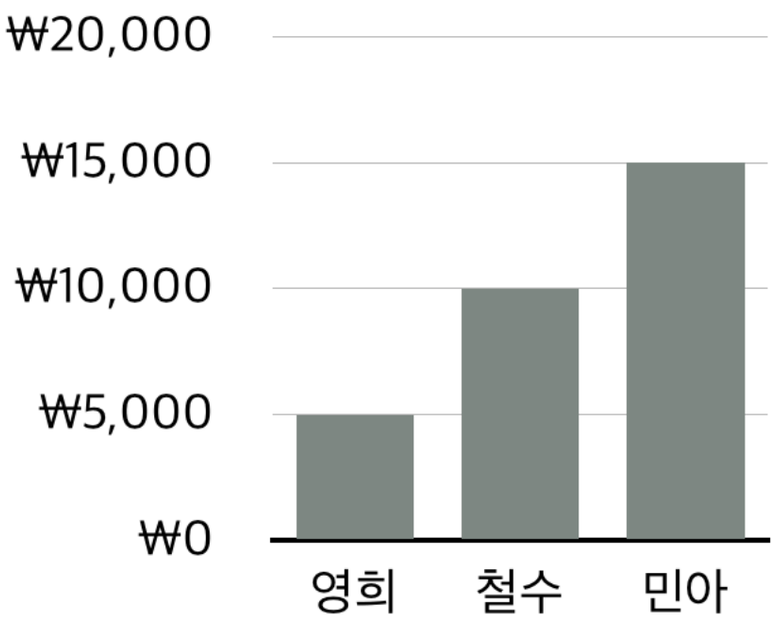
	용돈
영희	₩7,000
철수	₩19,000
민아	₩20,000



통계적 그래픽의 목적을 배반한다

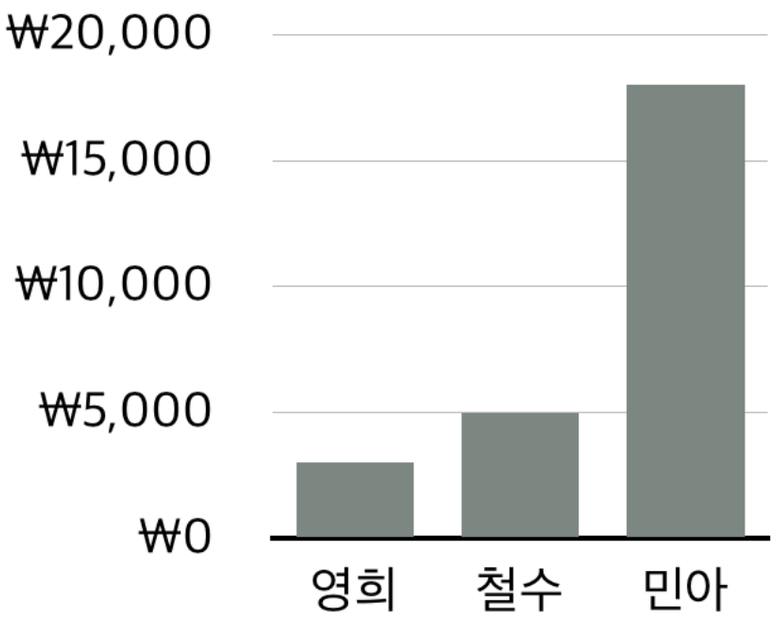
용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



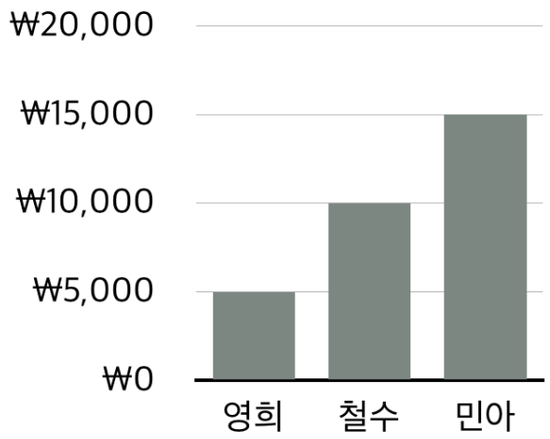
용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



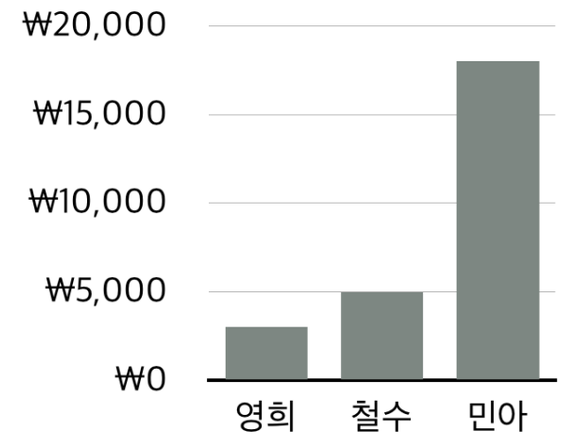
용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



그래프는 데이터에 대해서 자동적으로 그려진다

Junk Chart

그래프는 데이터에 대해서 자동적으로 그려진다

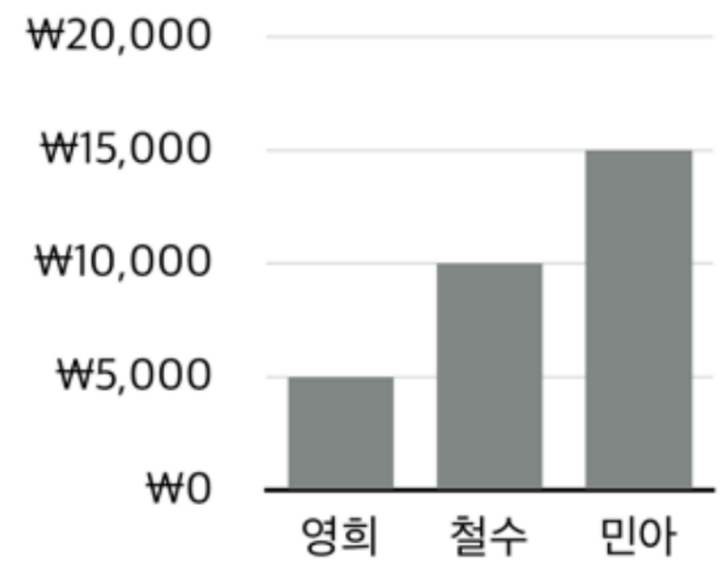
다르게 표현하면
무엇도 (직접) 그리지 않는다

Data

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000

Graph



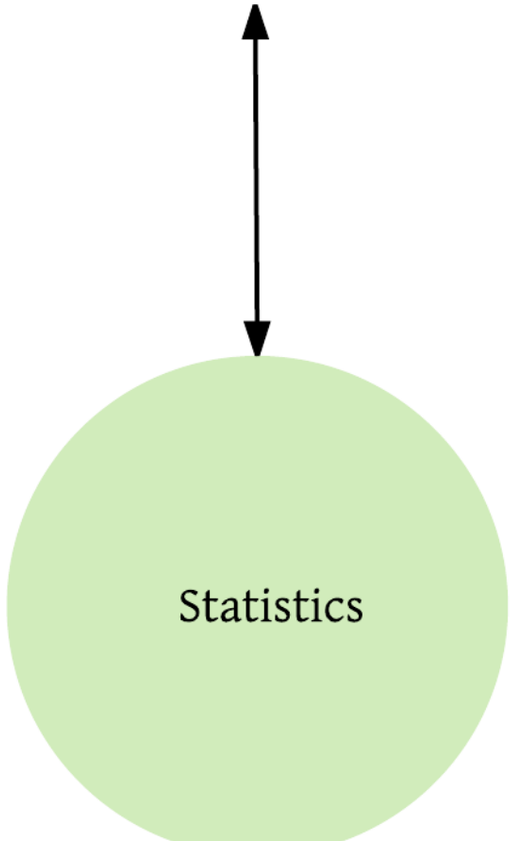
정보시각화의 특징

시각적 표현
추상적 데이터

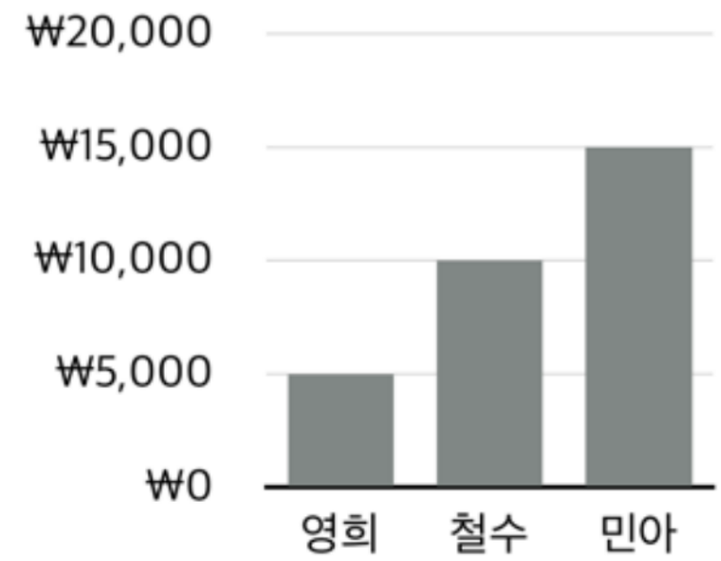
Data

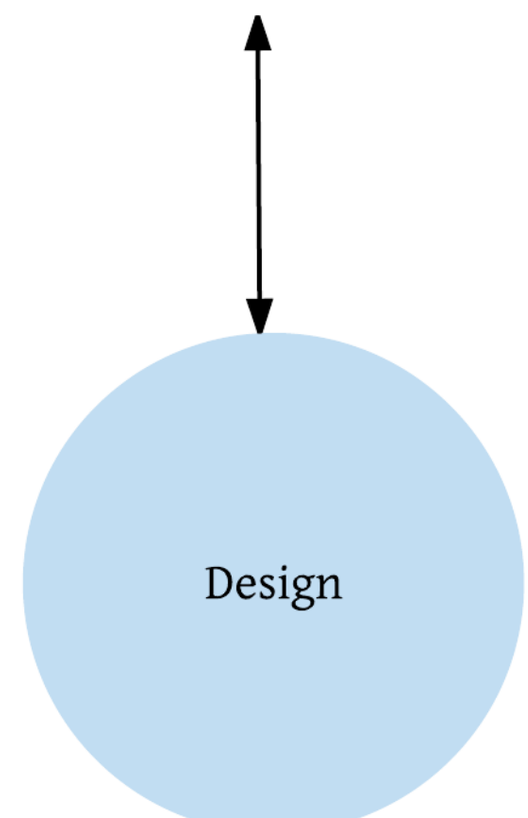
용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



Graph





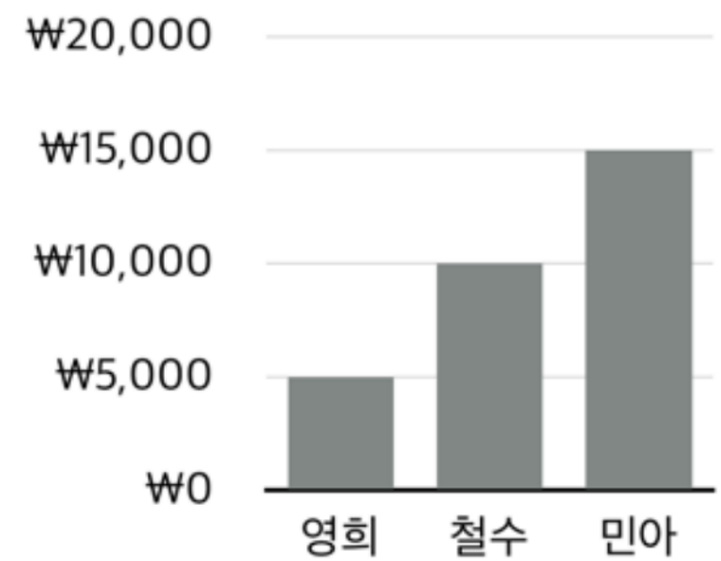
아직 이야기하지 않은 것

Data

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000

Graph



Data

용돈

	용돈
연희	₩5,000
	₩
	₩

Graph

HOW

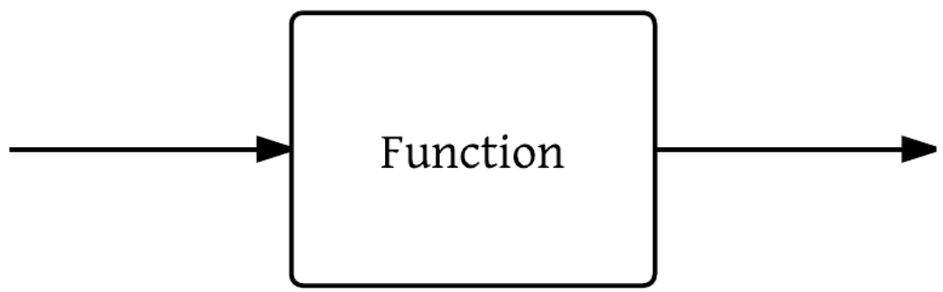
데이터는 어떻게 그래프가 되는가?

그래프는 자연발생적이지 않다
1800년대에 처음으로 발명

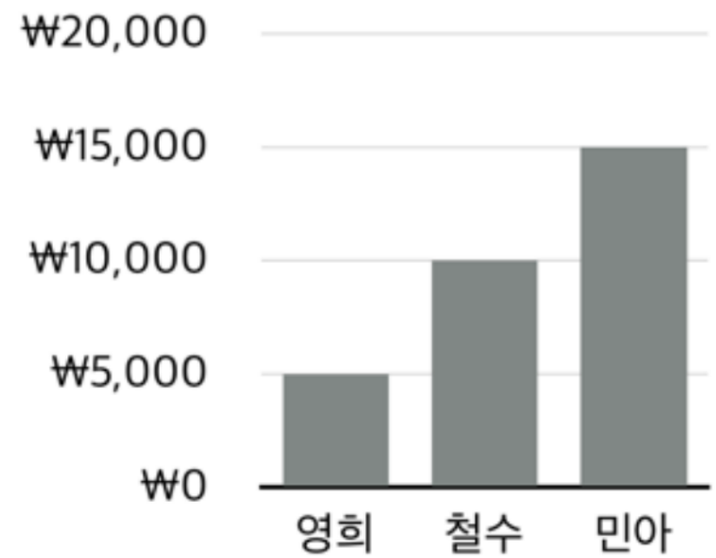
Data

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000



Graph



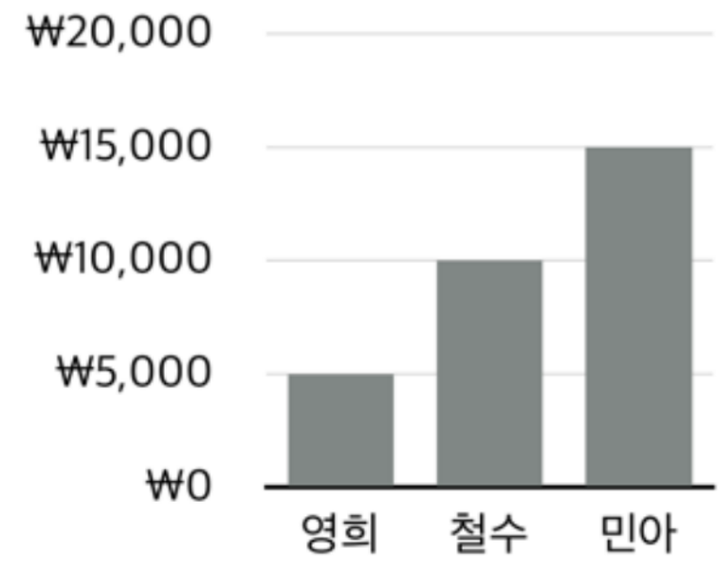
Data

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000

Bar Chart Function

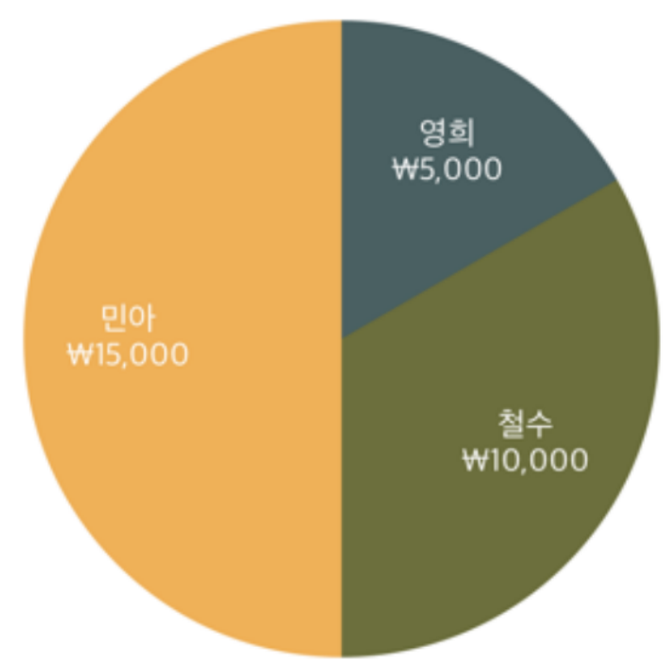
Graph



Input : Data

Pie Chart Function

Output : Graph



다시 The Grammar of Graphics

통계적 그래프의 자동성

우리는 기저의 데이터를 변경시키거나 데이터를 정확하고 적절하게 표현해야한다는 통계적 그래픽의 목적을 배반하지 않는 한, 점들의 위치나 객체의 색(이러한 요소들은 데이터를 표현하는 속성들이라는 가정하에)들을 바꿀 수 없다. 따라서 그래픽 시스템의 핵심은 스펙(*specification*)에 있다.

– The Grammar of Graphics 6p

통계적 그래프의 자동성

우리는 기저의 데이터를 변경시키거나 데이터를 정확하고 적절하게 표현해야한다는 통계적 그래픽의 목적을 배반하지 않는 한, 점들의 위치나 객체의 색(이러한 요소들은 데이터를 표현하는 속성들이라는 가정하에)들을 바꿀 수 없다. **따라서 그래픽 시스템의 핵심은 스펙(*specification*)에 있다.**

– The Grammar of Graphics 6p

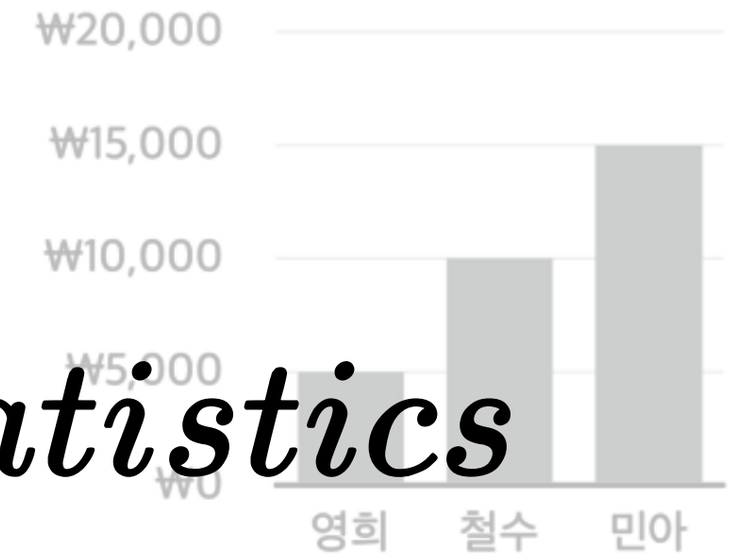
Data

Graph

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000

Bar Chart Function



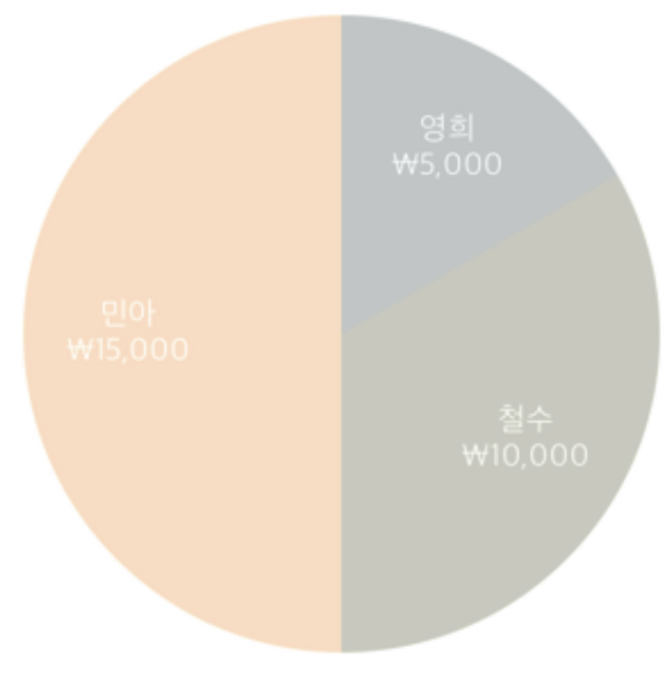
$$Specification \neq Statistics$$

$$Specification \not\subset Statistics$$

Input : Data

Pie Chart Function

Output : Graph



Data

Graph

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000

Bar Chart Function

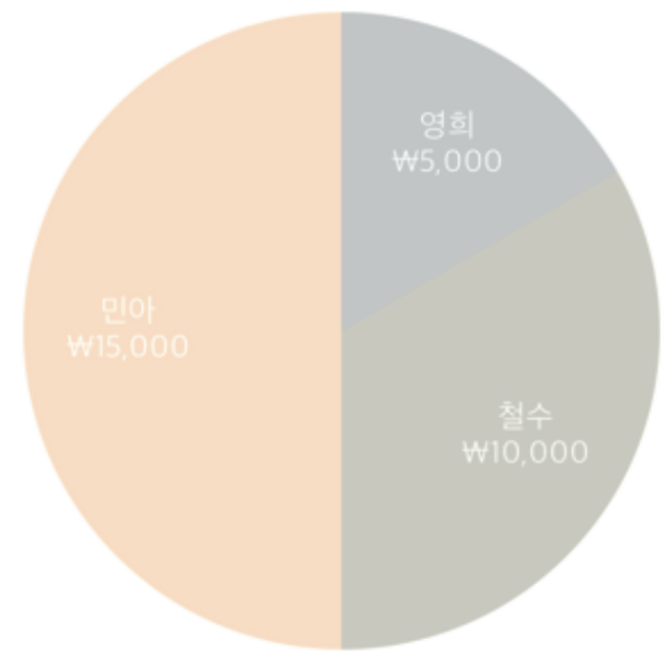
$$Specification \neq Design$$

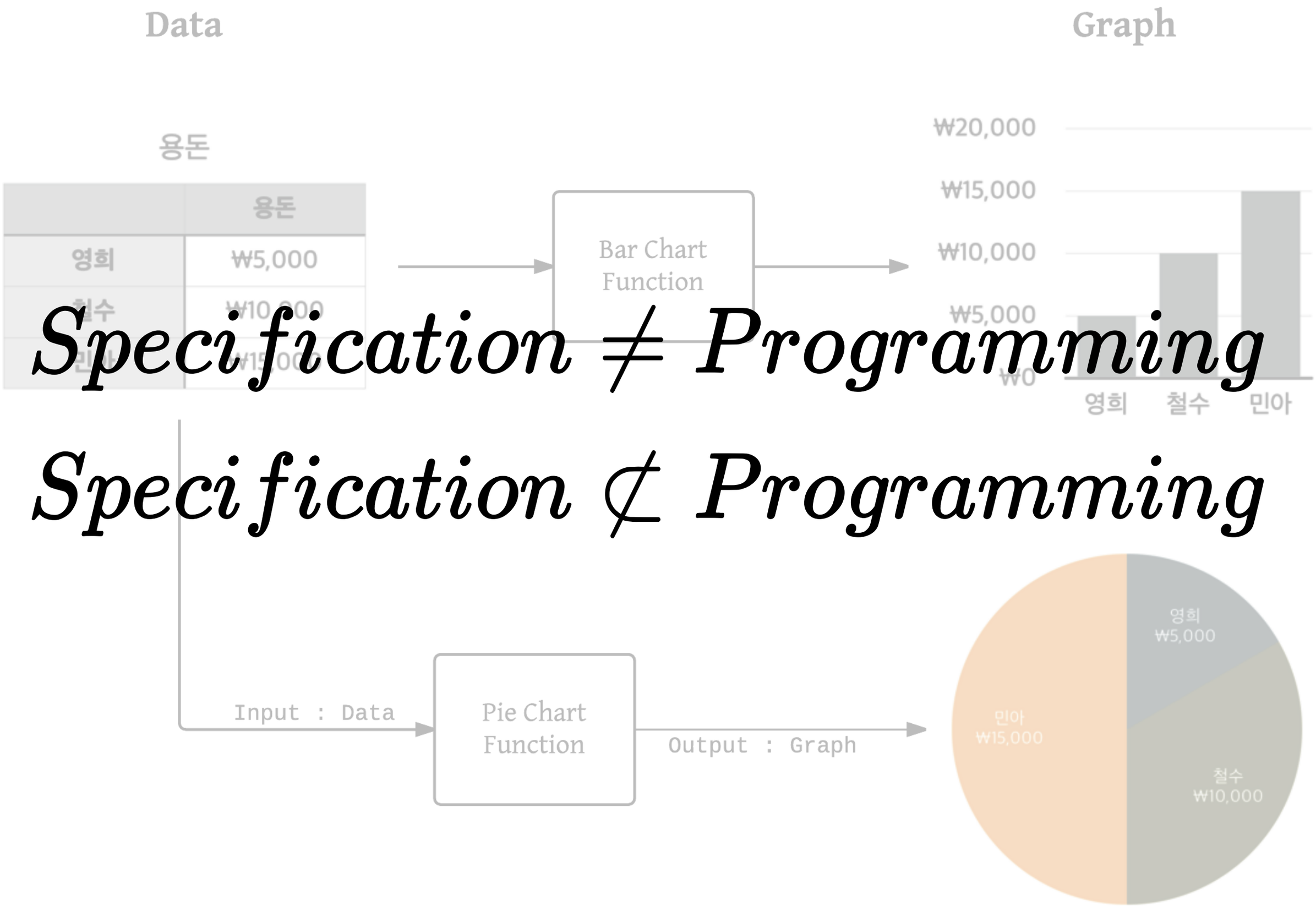
$$Specification \not\subset Design$$

Input : Data

Pie Chart Function

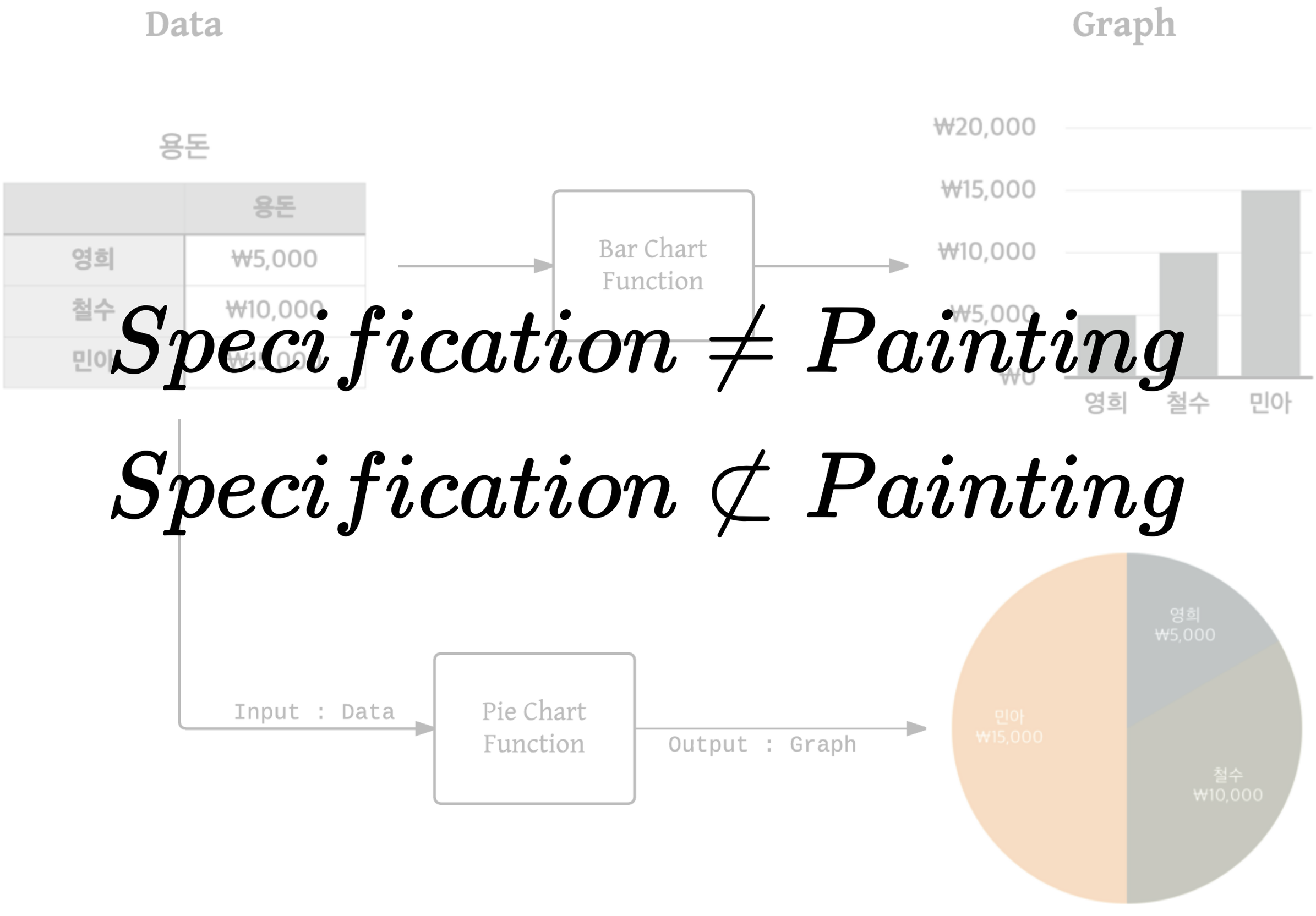
Output : Graph



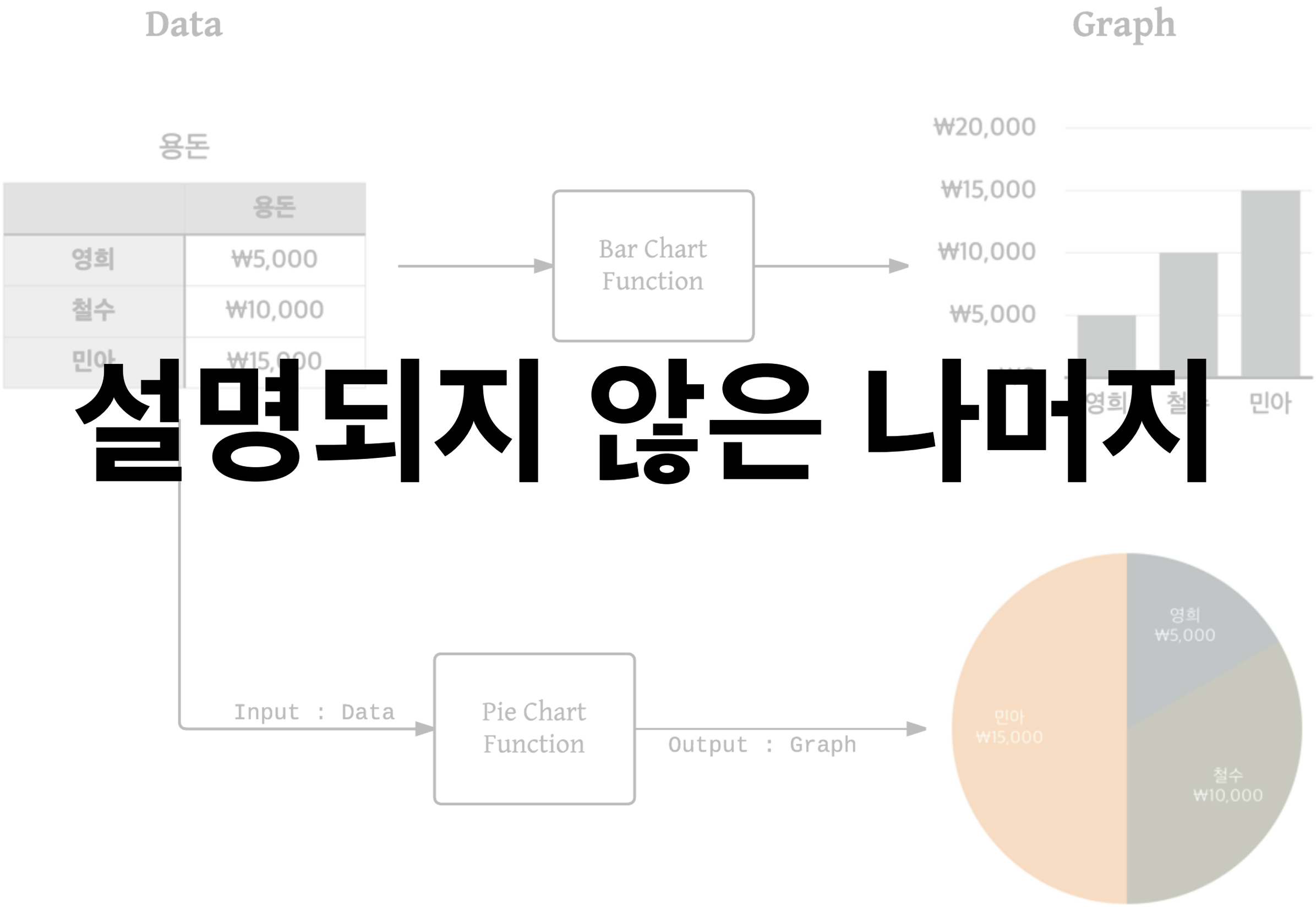


$$Specification \neq Programming$$

$$Specification \not\subset Programming$$


Data
Graph
용돈
용돈
영희
₩5,000
철수
₩10,000
민아
₩15,000
Bar Chart Function
₩20,000
₩15,000
₩10,000
₩5,000
₩0
영희
철수
민아
Specification ≠ Painting
Specification ⊄ Painting
Input : Data
Pie Chart Function
Output : Graph
영희 ₩5,000
민아 ₩15,000
철수 ₩10,000

설명되지 않은 나머지


Data
Graph
용돈
용돈
영희 ₩5,000
철수 ₩10,000
민아 ₩15,000
Bar Chart Function
₩20,000
₩15,000
₩10,000
₩5,000
영희
철수
민아
Input : Data
Pie Chart Function
Output : Graph
영희 ₩5,000
민아 ₩15,000
철수 ₩10,000

Data

Graph

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000

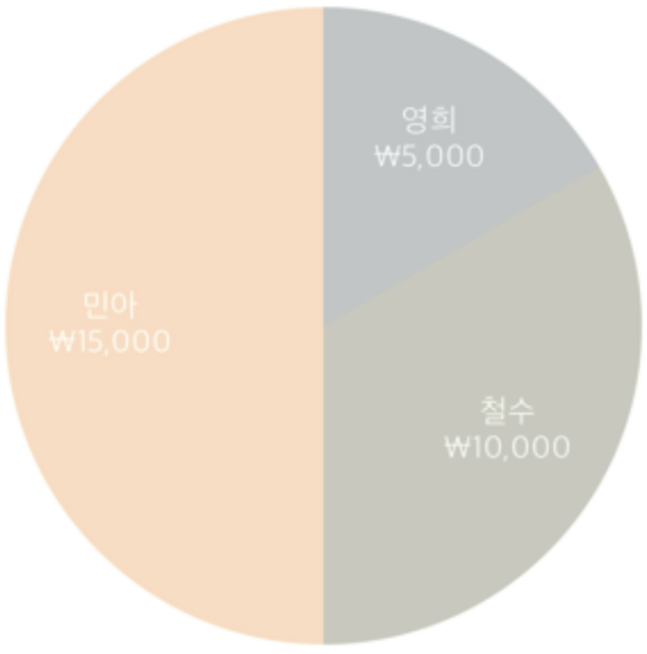
Bar Chart Function

바로 여기가 정보시각화 고유의 영토

Input : Data

Pie Chart Function

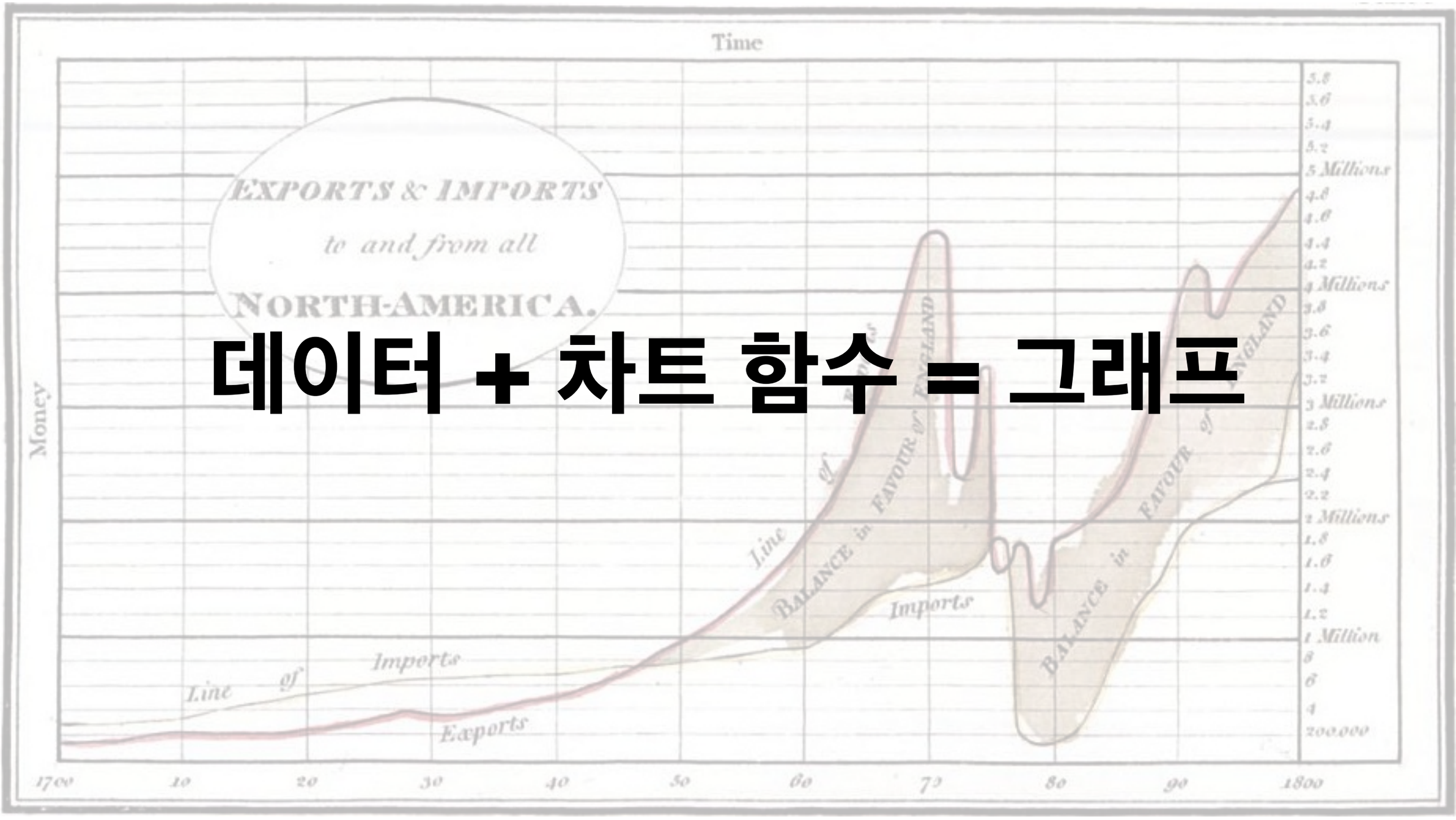
Output : Graph



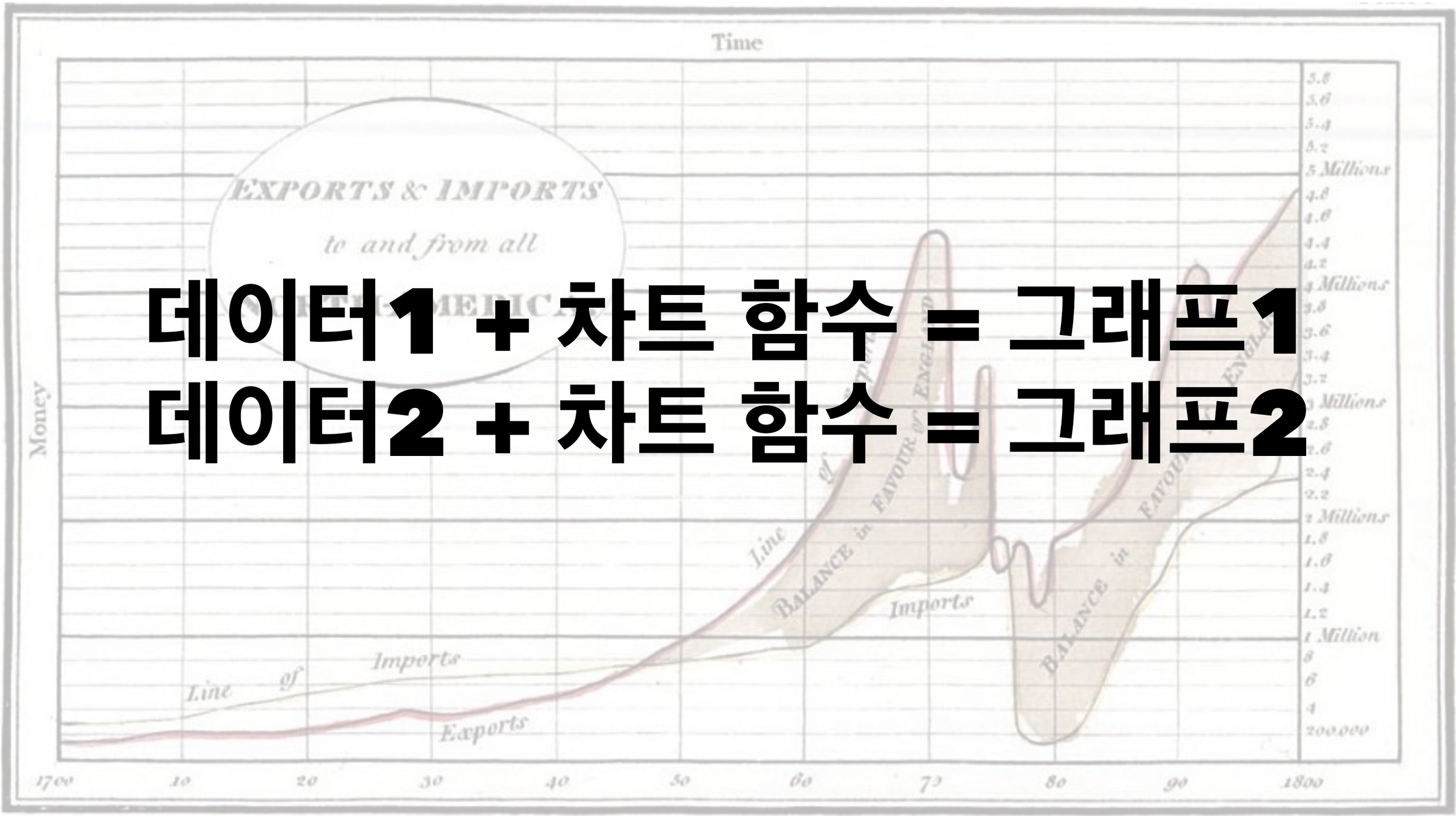
왜 시각화는 회화가 아닌가

1. 표현하는 대상이 다르다
2. 데이터에 의해 자동적으로 그려진다

그리고 한 가지 더

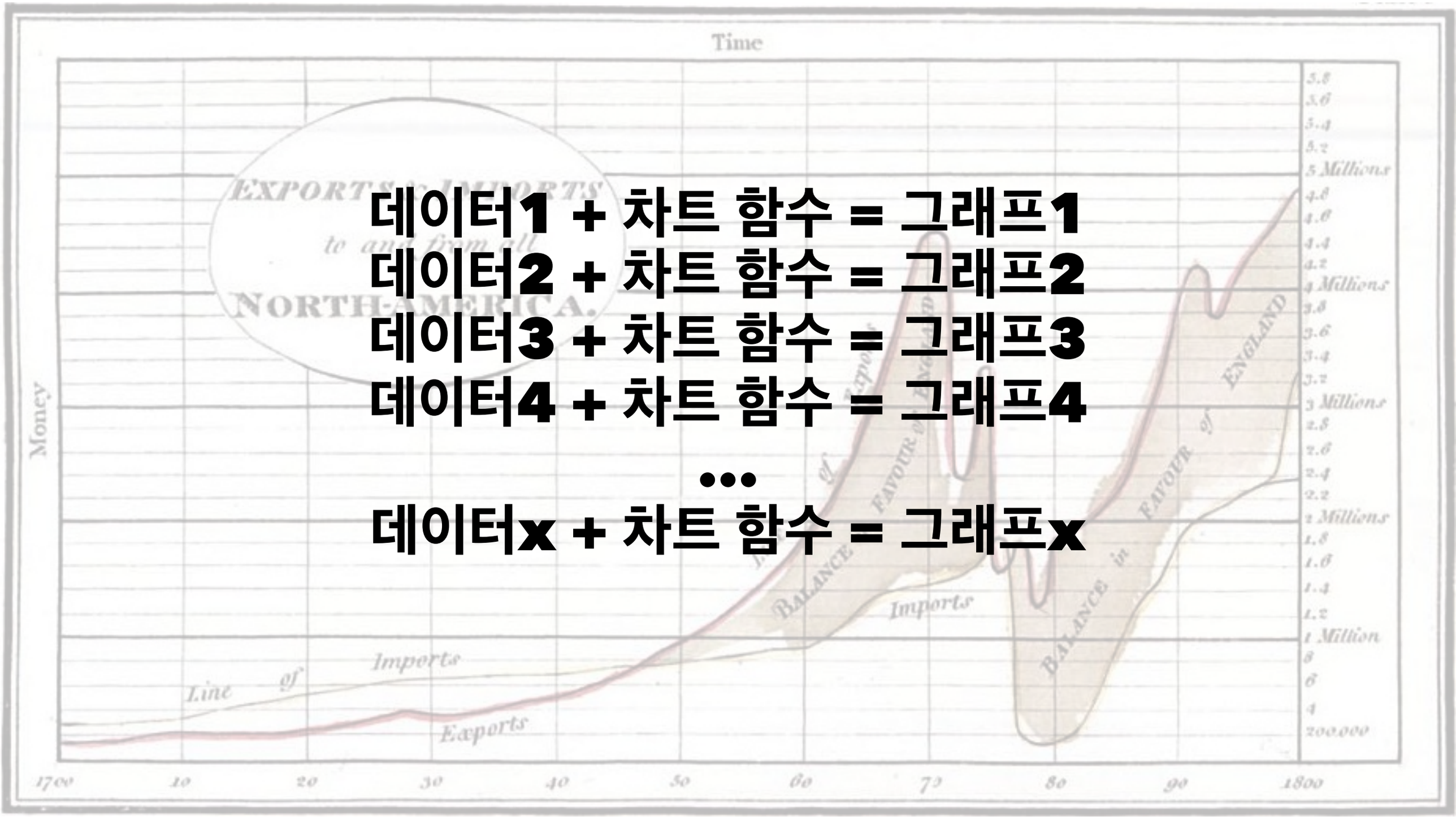

Time
EXPORTS & IMPORTS
to and from all
NORTH-AMERICA.
Money
Line of Imports
Line of Exports
Imports
BALANCE in FAVOUR of ENGLAND
BALANCE in FAVOUR of ENGLAND
5.8
5.6
5.4
5.2
5 Millions
4.8
4.6
4.4
4.2
4 Millions
3.8
3.6
3.4
3.2
3 Millions
2.8
2.6
2.4
2.2
2 Millions
1.8
1.6
1.4
1.2
1 Million
8
6
4
200,000
1700
10
20
30
40
50
60
70
80
90
1800

데이터 + 차트 함수 = 그래프



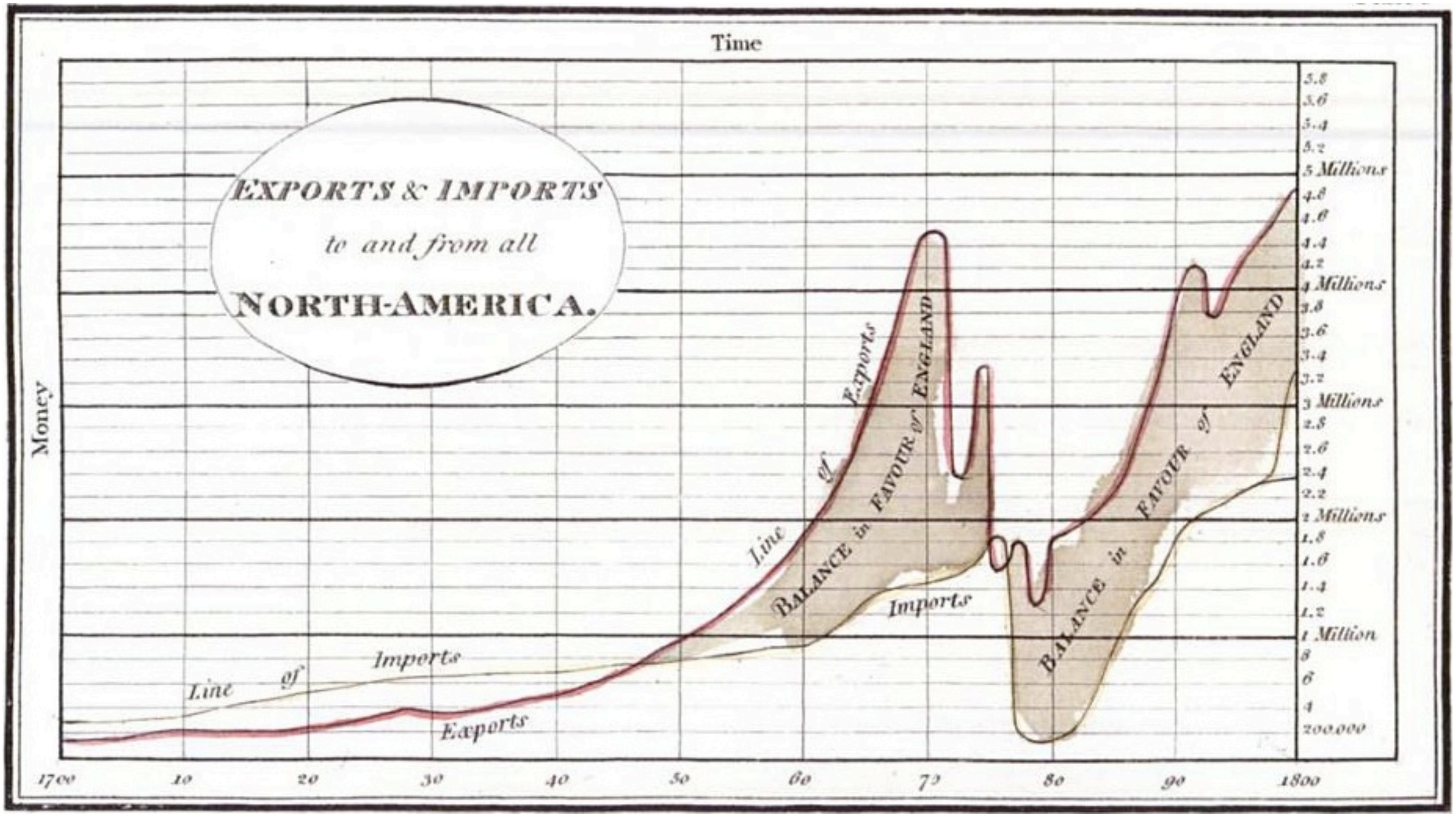
데이터1 + 차트 함수 = 그래프1

데이터2 + 차트 함수 = 그래프2

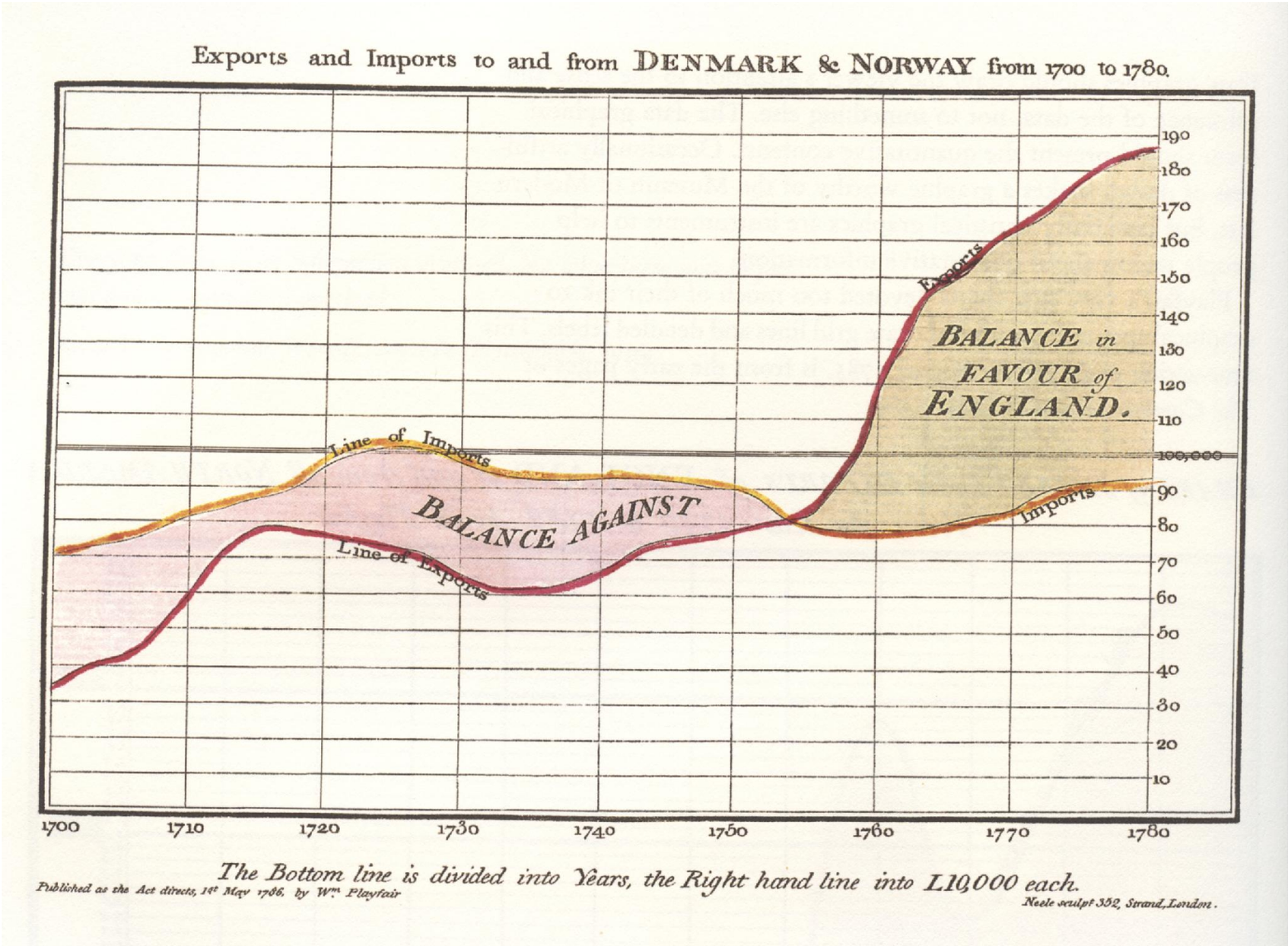


데이터1 + 차트 함수 = 그래프1
데이터2 + 차트 함수 = 그래프2
데이터3 + 차트 함수 = 그래프3
데이터4 + 차트 함수 = 그래프4
•••
데이터x + 차트 함수 = 그래프x

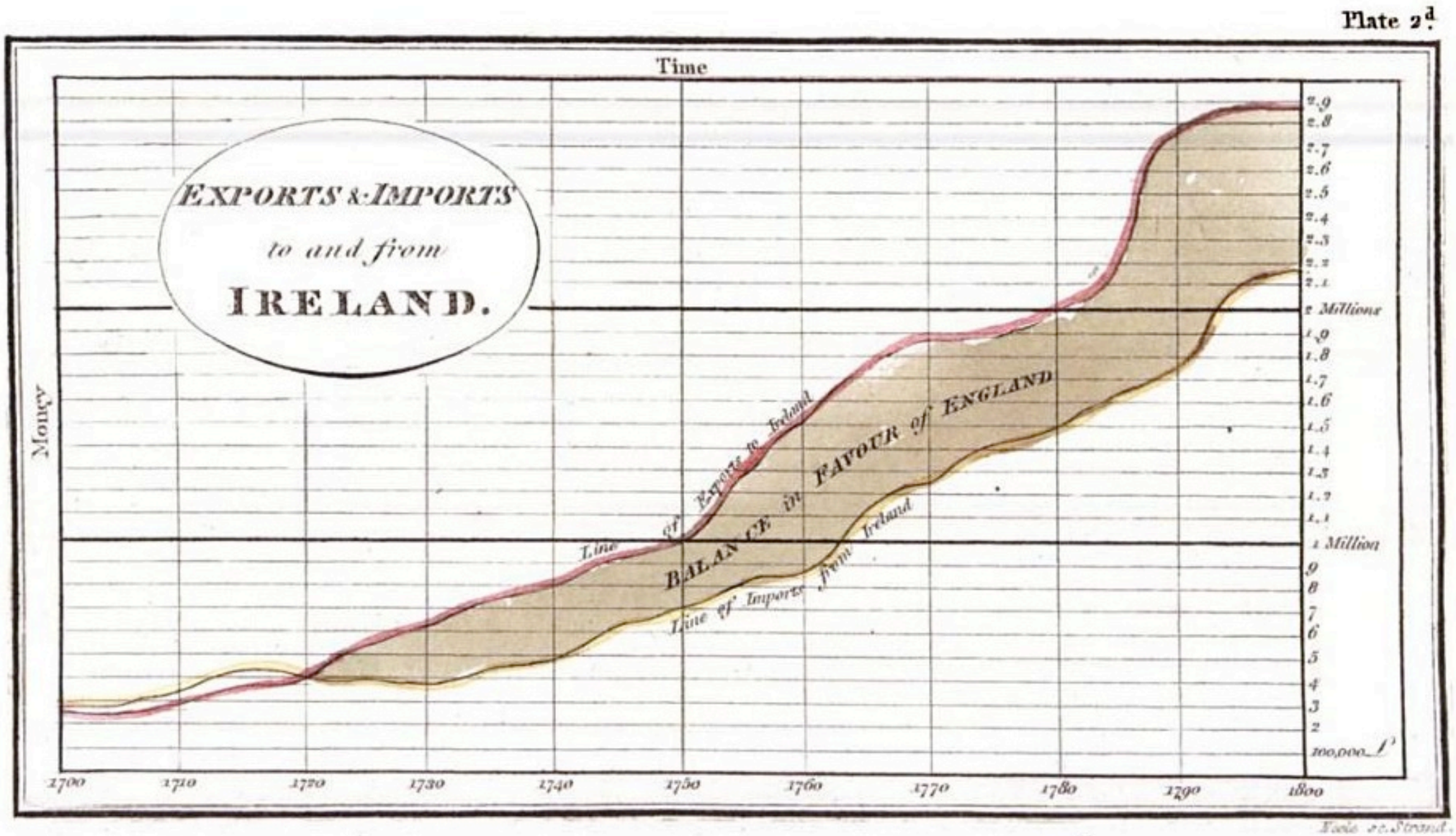
영국 대 북미 무역 수지


Time
EXPORTS & IMPORTS
to and from all
NORTH-AMERICA.
Money
Line of Exports
BALANCE in FAVOUR of ENGLAND
Imports
Line of Imports
Exports
BALANCE in FAVOUR of ENGLAND
5.8
5.6
5.4
5.2
5 Millions
4.8
4.6
4.4
4.2
4 Millions
3.8
3.6
3.4
3.2
3 Millions
2.8
2.6
2.4
2.2
2 Millions
1.8
1.6
1.4
1.2
1 Million
8
6
4
200.000
1700
10
20
30
40
50
60
70
80
90
1800

영국 대 노르웨이 & 덴마크 수지



영국 대 아일랜드 무역 수지



회화는 그 자체로 구상적으로 완결
시각화는 추상적이므로 비완결적

윌리엄 플레이페어는 잊혀지고 차트 형식만 살아남은 이유

추상적인 무언가

$$Chart \approx Function \approx Specification$$

Specification의 정체

데이터를 시각적 표현으로 변환하는 규칙들의 집합

아까 차트

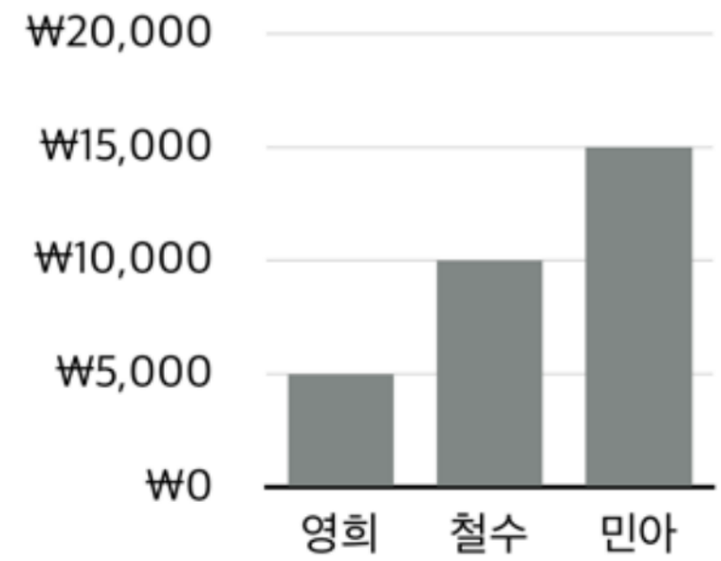
Data

용돈

	용돈
영희	₩5,000
철수	₩10,000
민아	₩15,000

Bar Chart Function

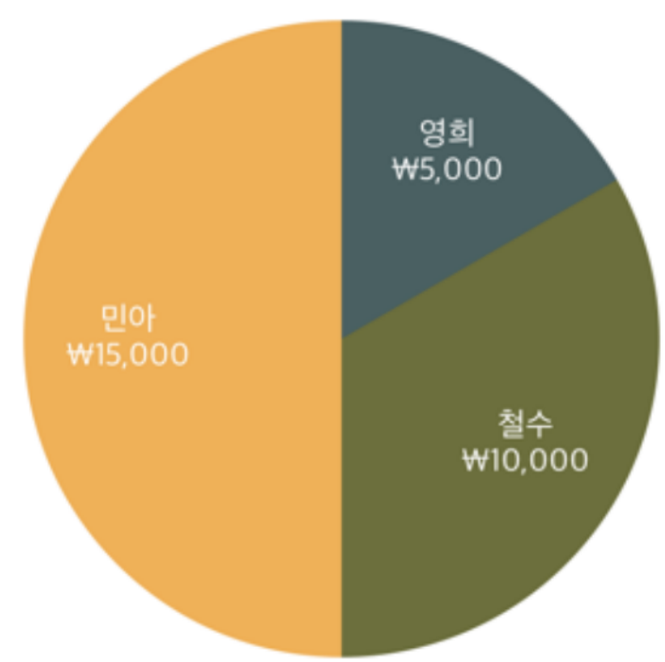
Graph



Input : Data

Pie Chart Function

Output : Graph



그런데, 시각화

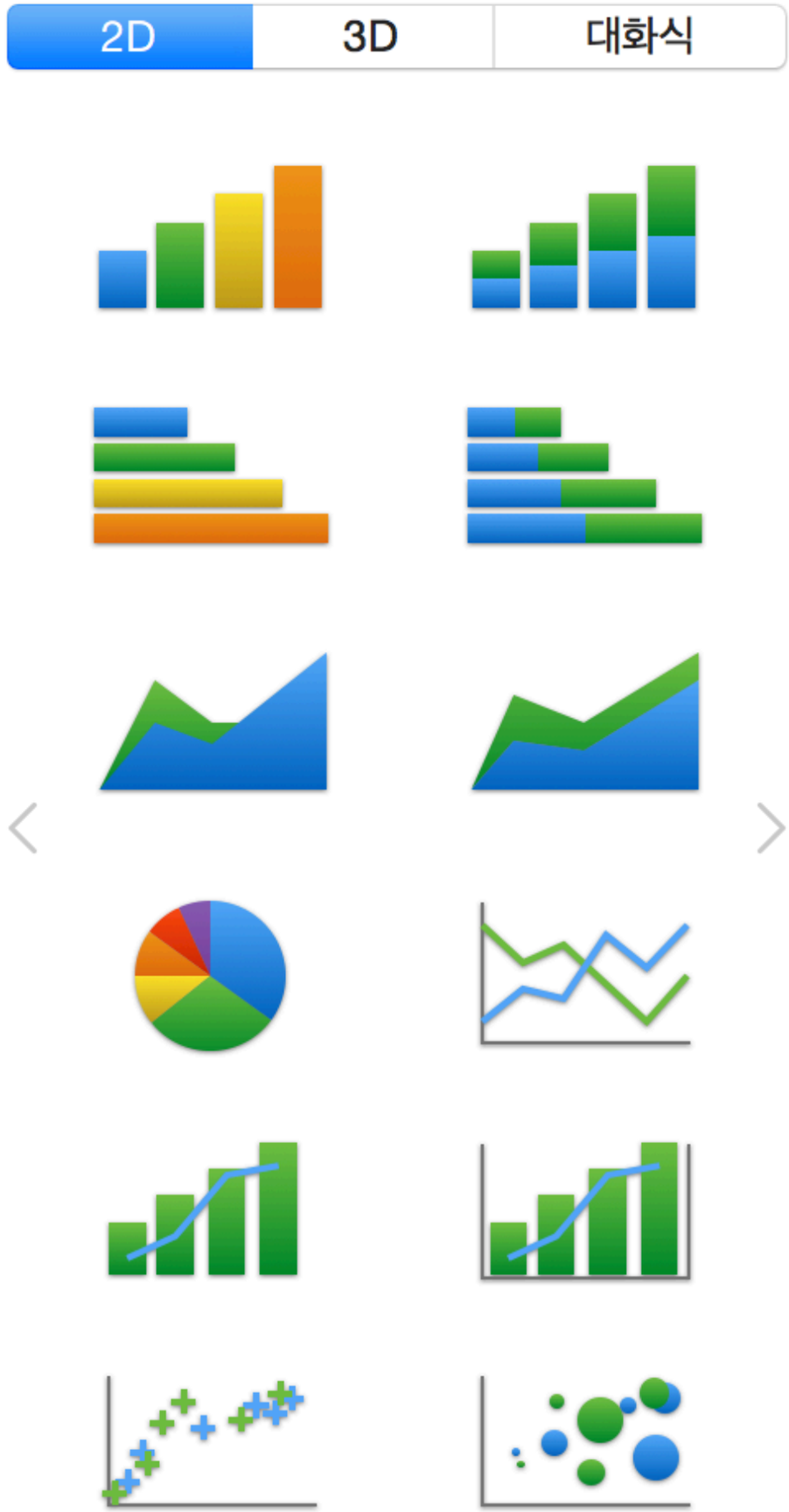
WES ANDERSON
Obama
McCain
Kerry
Bush
Gore
30 sec

SNOWMEN

뭔가 속은 기분

차트

일반적이고 정형화된 규칙 집합
미리 디자인된 규칙집합
프로그램에서 주어지는 규칙집합


2D
3D
대화식

2D 3D 대화식

Spec, Spec
Spec, Spec
Spec, Spec
Spec, Spec
Spec, Spec
Spec, Spec

Rules, Rules
Rules, Rules
Rules, Rules
Rules, Rules
Rules, Rules
Rules, Rules

차트 라이브러리
차트 지원 프로그램

각 차트 타입은 그래프를 생성하는 하나의 규칙집합

그래픽 문법의 세계로

그래픽의 문법(1)

Grammar makes language expressive. A language consisting of words and no grammar (statement = word) expresses only as many ideas as there are words. By specifying how words are combined in statements, a grammar expands a language's scope.

– Leland Wilkinson

The Grammar of Graphics 1p

그래픽의 문법(1)

문법은 언어의 표현을 풍부하게 만든다. 단어만 있고 문법이 없는 언어가 있다면(즉, 단어가 곧 문장), 오직 단어가 존재하는 만큼의 아이디어만을 표현할 수 있을 것이다. 문장 안에서 단어가 어떻게 구성되는 지를 규정함으로서, 문법은 언어의 범위를 확장한다.

– Leland Wilkinson

The Grammar of Graphics 1p

그래픽의 문법(2)

This book is about grammatical rules for creating perceivable graphs, or what we call graphics. The grammar of graphics takes us beyond a limited set of charts (words) to an almost unlimited world of graphical forms (statements).

– Leland Wilkinson

The Grammar of Graphics 1p

그래픽의 문법(2)

이 책은 인지할 수 있는 그래프나 우리가 그래픽이라고 부르는 것을 만들기 위한 문법적 규칙에 관한 책이다. 그래픽의 문법은 우리를 제한된 종류의 차트들(단어)로부터 거의 무한한 그래픽 형태(문장)의 세계로 데려갈 것이다.

– Leland Wilkinson

The Grammar of Graphics 1p

$$Word \approx Chart$$

차트의 한계

단어적 세계

막대 그래프, 원 그래프, 꺾은선 그래프, 누적 막대그래프, 레이더 차트, 산포도, 상자 수염 그램, 평행 그래프, 히스터그램, 상포토 행렬, 3차원 막대그래프, 가로 막대 그래프, 영역 그래프, ...

문법적으로 분석하기

GPL

Graphic Production Language

GPL

- The Grammar of Graphics의 이론에서 사용
- SPSS에 실제 작동하는 구현이 있음
- 그래픽의 문법을 기술하기 위한 고유의 문법
- 고수준 선언적 언어(요소를 직접 그리지 않음)

How to make a pie

Figure 2.1 *How to make a pie*

The Grammar of Graphics 23p

From data to graphic

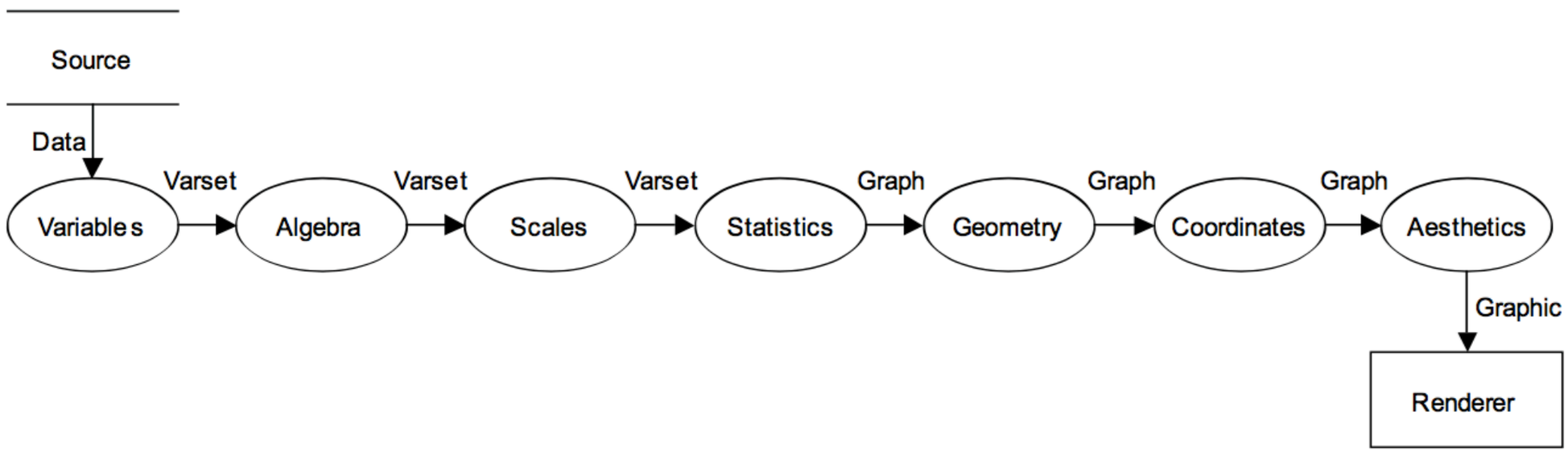


Figure 2.2 *From data to graphic*

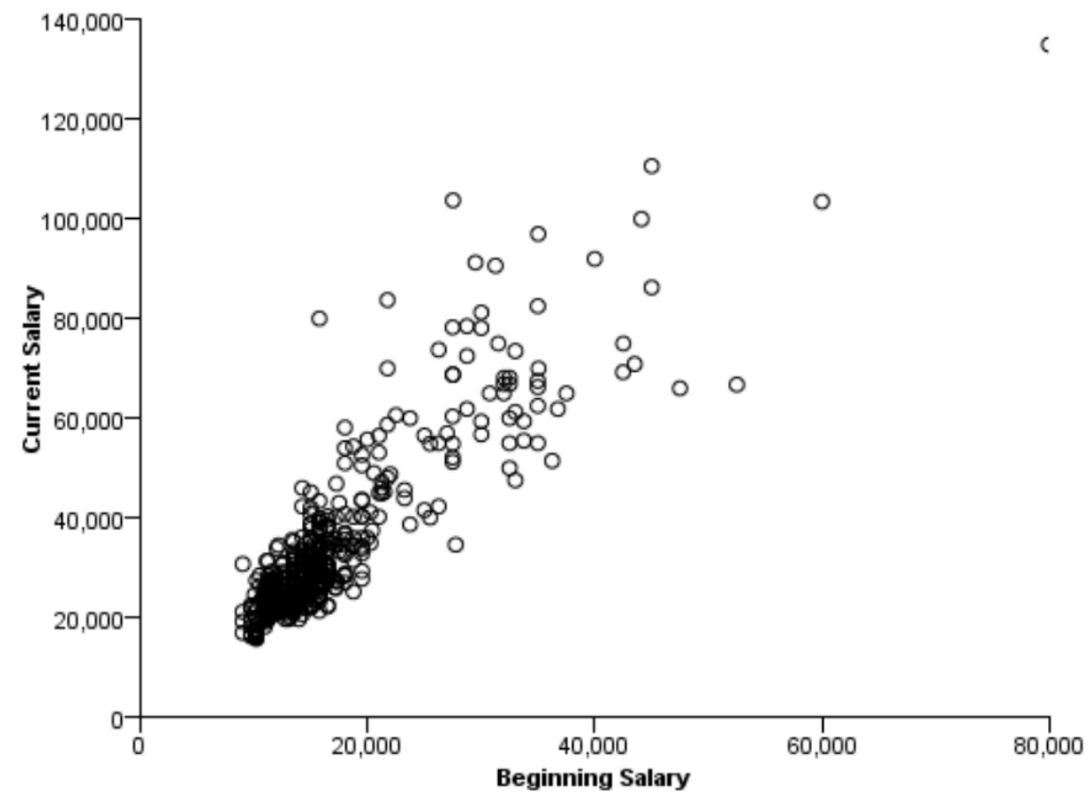
The Grammar of Graphics 24p

GPL - scatter plot

```
SOURCE: s = userSource(id("Employeedata"))
DATA: salbegin=col(source(s), name("salbegin"))
DATA: salary=col(source(s), name("salary"))
GUIDE: axis(dim(2), label("Current Salary"))
GUIDE: axis(dim(1), label("Beginning Salary"))
ELEMENT: point(position(salbegin*salary))
```

GPL Reference Guide for IBM SPSS Statistics

Figure 1-4
Simple 2-D scatterplot



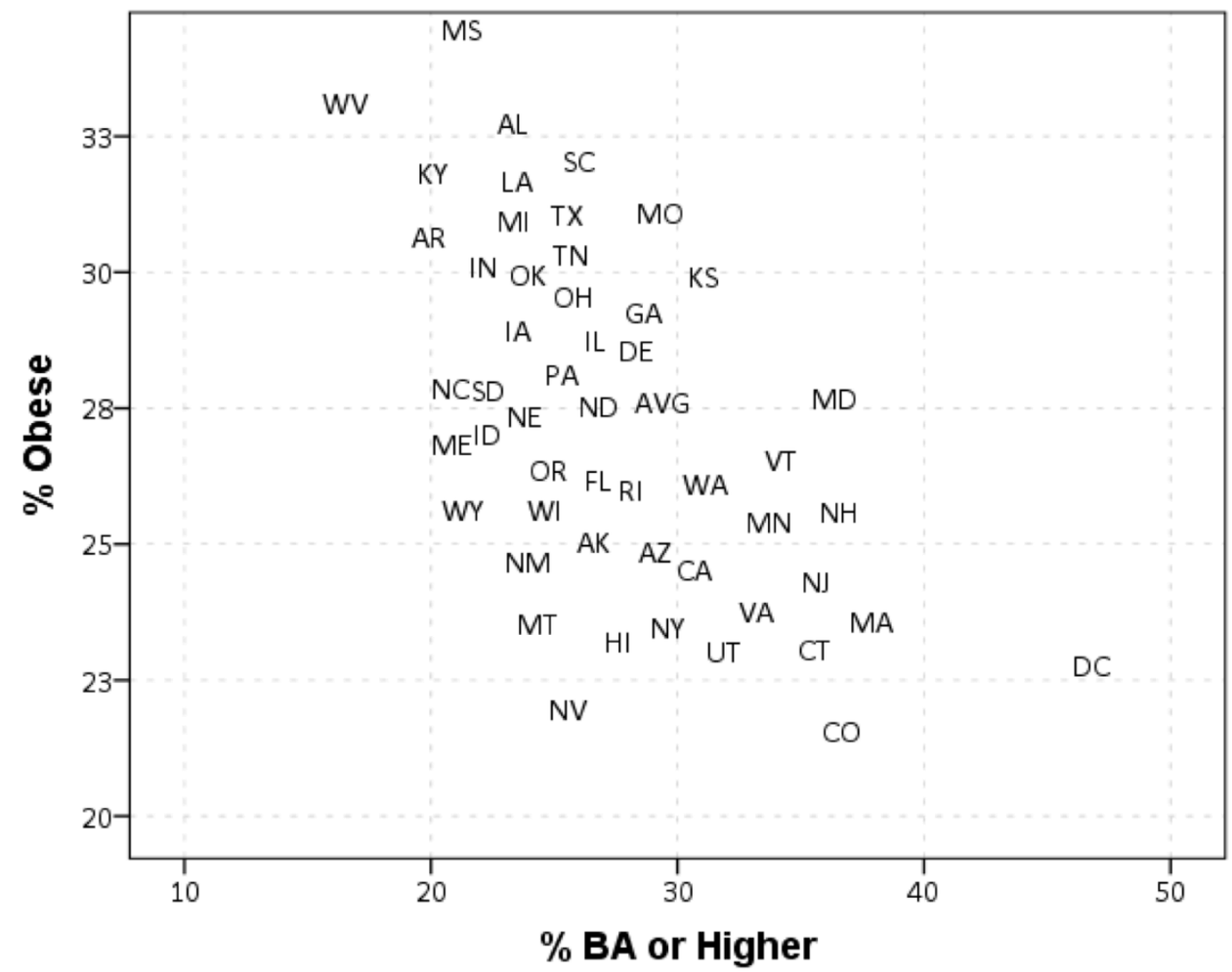
장점이 뭘까요?

점 대신 텍스트를 사용하고 싶다면?

GPL - scatter plot?

```
SOURCE: s=userSource(id("graphdataset"))
DATA: BAORHIGHER=col(source(s), name("BAORHIGHER"))
DATA: OBESITY=col(source(s), name("OBESITY"))
DATA: StateAbbr=col(source(s), name("StateAbbr"),
                    unit.category())
GUIDE: axis(dim(1), label("% BA or Higher"))
GUIDE: axis(dim(2), label("% Obese"))
ELEMENT: point(position(BAORHIGHER*OBESITY),
               label(StateAbbr),
               size(0))
```

Andrew Wheeler - Labeling tricks in SPSS plots



문법의 차이

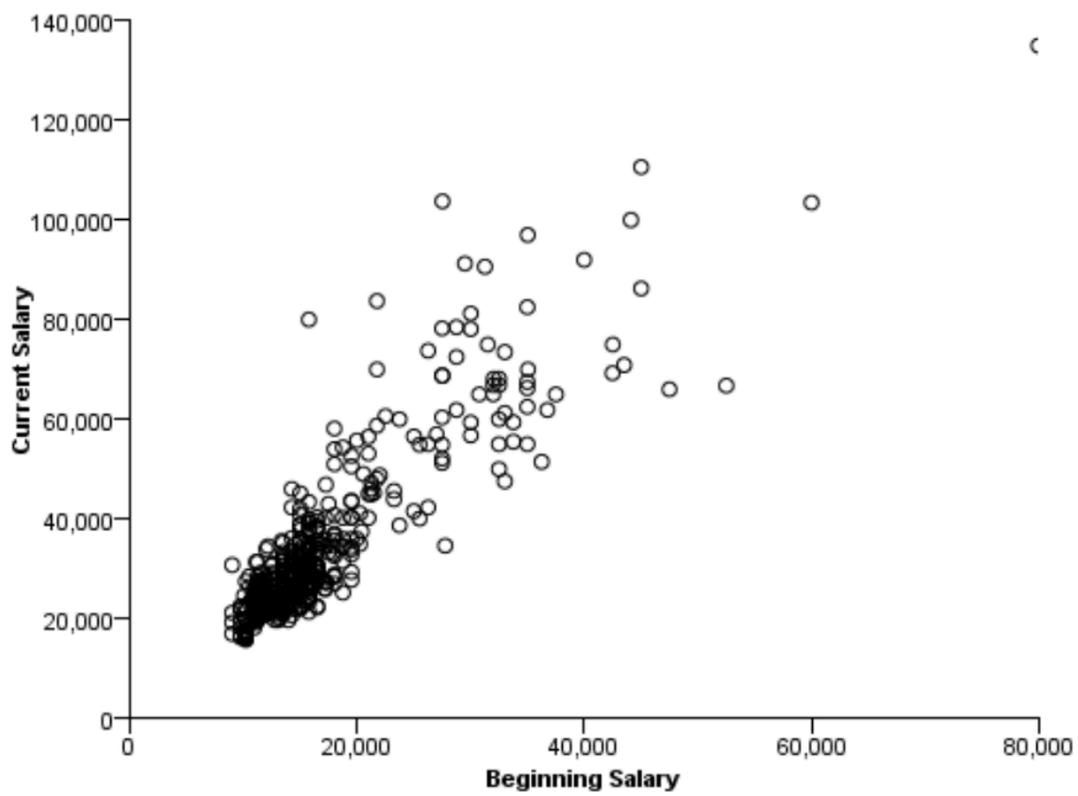
점으로 표현하기

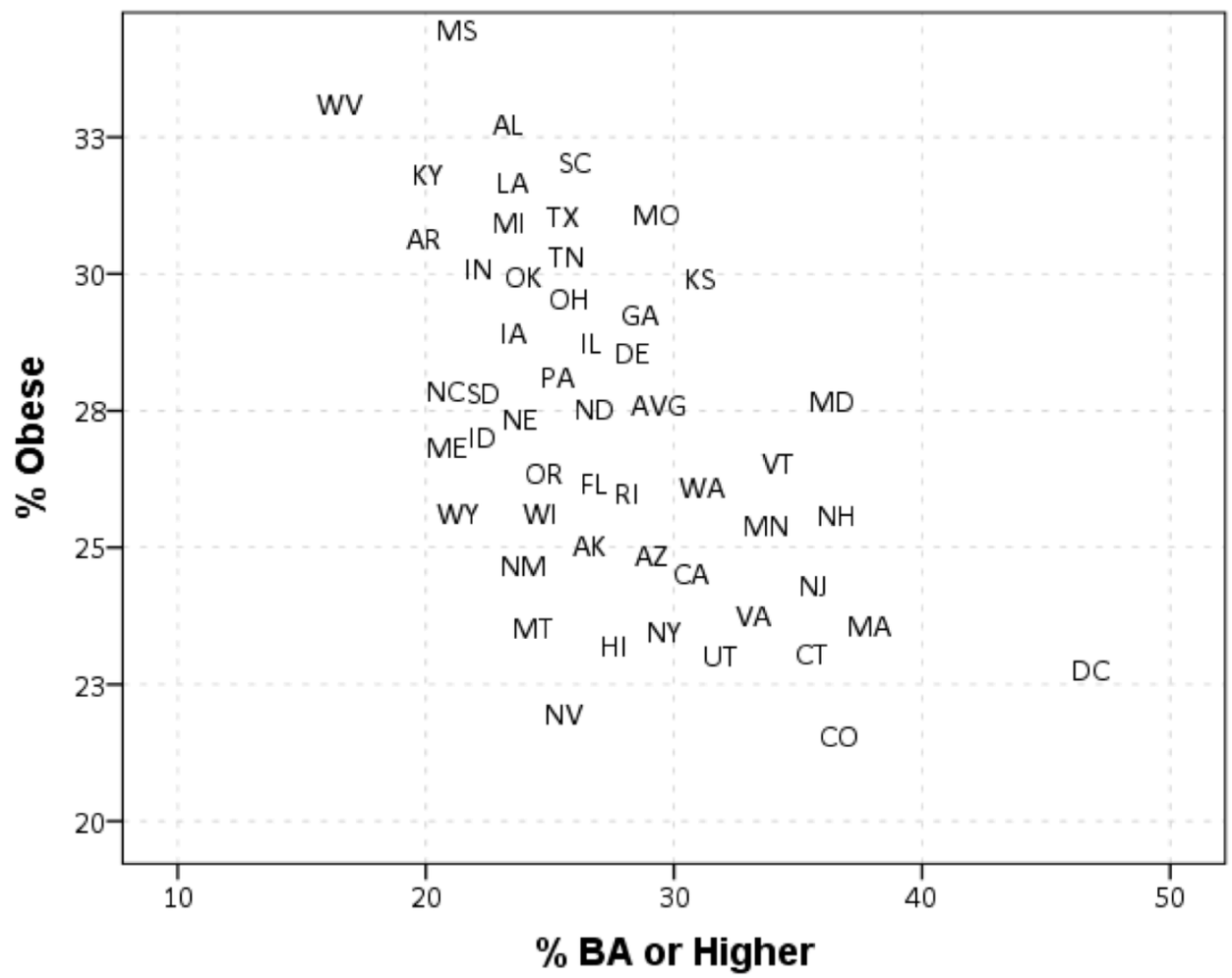
```
ELEMENT: point(position(salbegin*salary))
```

텍스트로 표현하기

```
ELEMENT: point(position(BAORHIGHER*OBESITY),
               label(StateAbbr),
               size(0))
```

Figure 1-4
Simple 2-D scatterplot





거의 모든 차트 프로그램에서 불가능한 일

definition of scatter diagram

a two-dimensional graph in rectangular coordinates consisting of points whose coordinates represent values of two variables under study

Merriam-Webster scatter diagram

definition of scatter diagram

a two-dimensional graph in rectangular coordinates consisting of **points** whose coordinates represent values of two variables under study

Merriam-Webster scatter diagram

scatter plot

정의에 점으로 표현한다는 의미가 포함
대부분의 차트는 이 정의에 충실

단어의 세계
문법의 세계

단, GPL은 거의 사용되지 않음

실구현이 있다고는 하나, 아무도 모름
현재 실제로 사용가능한지도 불투명

ggplot2

GoG의 유산

ggplot2의 gg

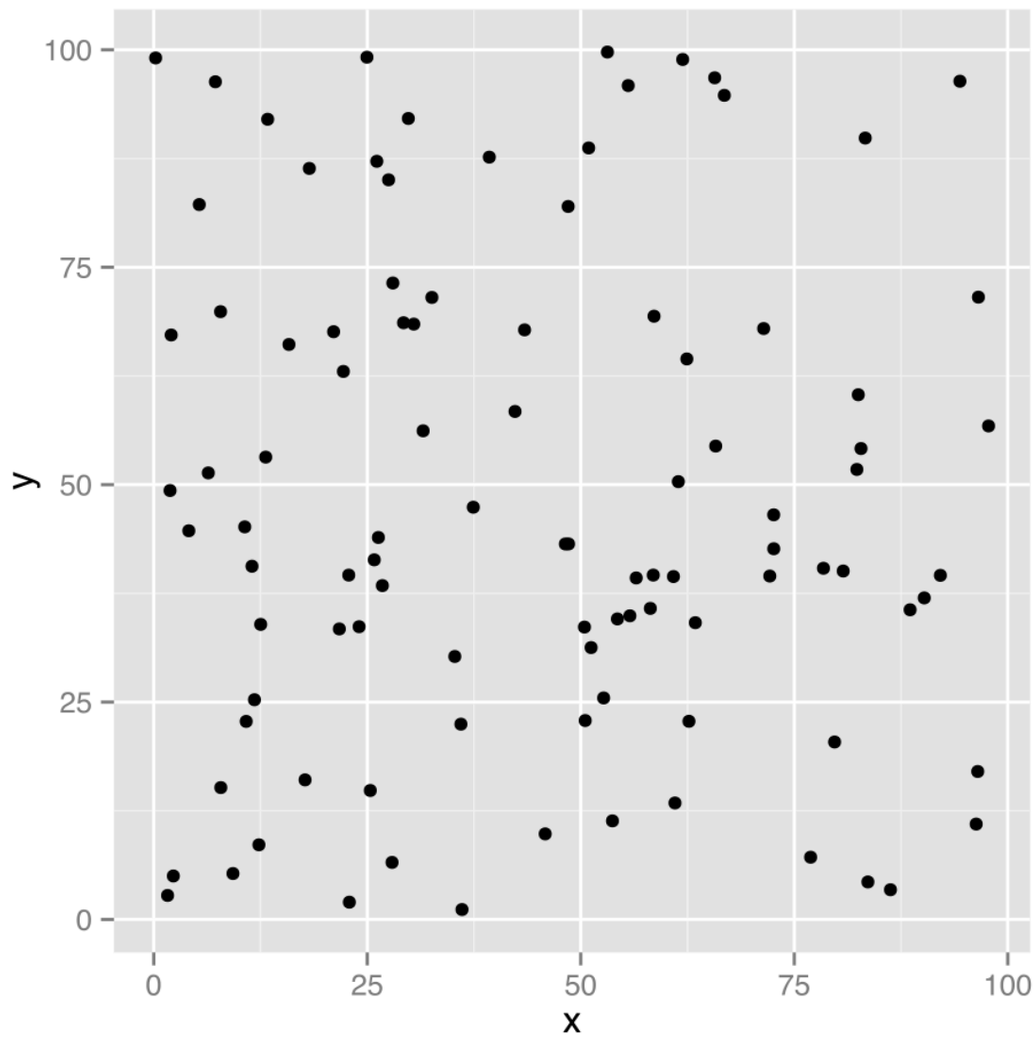
The Grammar of Graphics

ggplot2에서 산포도 그리기 (1)

```
> dataset = data.frame(x=runif(100)*99, y=runif(100)*99)
> head(dataset)
#   x   y
# 1 0.6458447   67.28732
# 2 13.53177    76.61337
# 3 74.25634    50.06963
# 4 0.01905423  80.6326
# 5 31.17823    86.01166
# 6 75.63284    34.53802
```

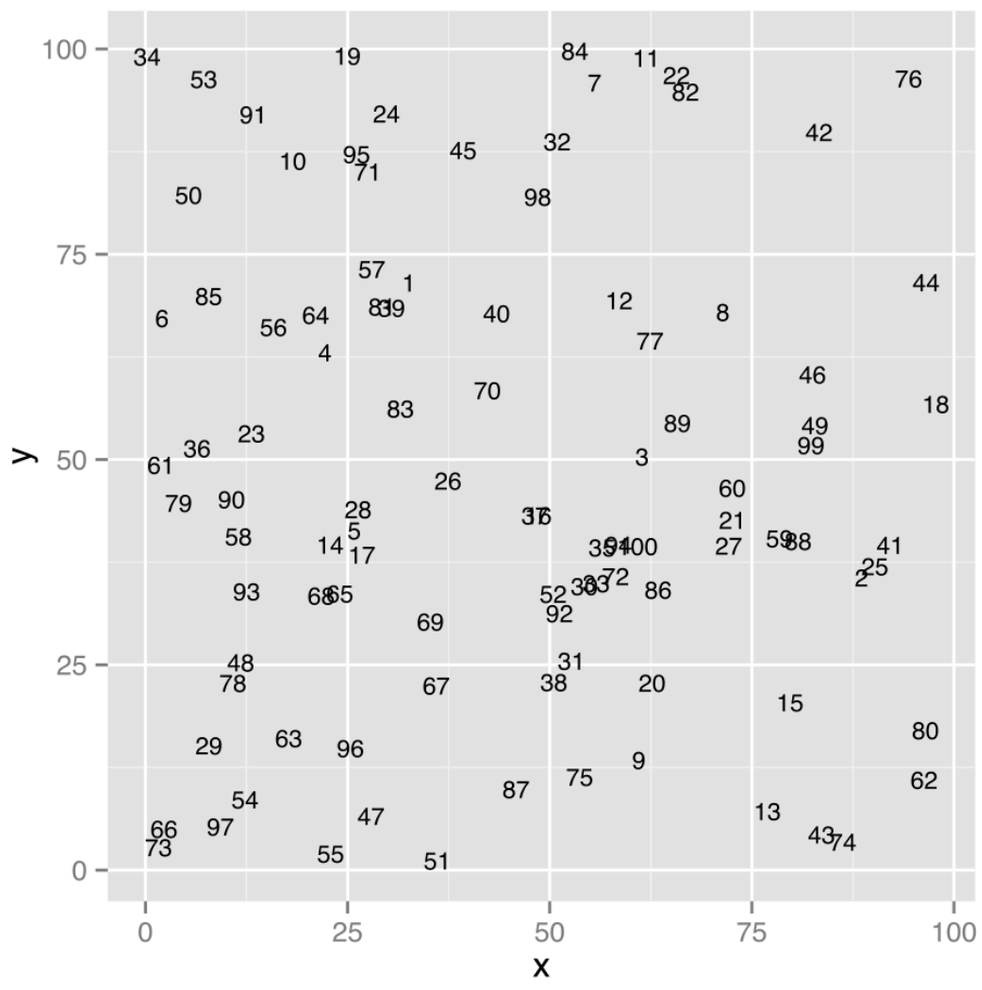
ggplot2에서 산포도 그리기 (2)

```
ggplot(dataset, aes(x = x, y = y)) + geom_point()
```



점 대신 텍스트 사용하기

```
id = seq(1, 100)
ggplot(dataset, aes(x = x, y = y)) + geom_text(label=id, size=3)
```



ggplot2의 특징

- 차트 도구들에 비해 압도적인 표현력
  - 조합을 통해 매우 다양한 표현 가능
  - 현재 R에서 가장 많이 사용되는 패키지
- GoG의 가장 실용적인 구현이라고 할 수 있음
  - GoG의 완전한 구현은 아님
  - 제약도 많고, 독자적인 특성들도 있음
- 잘 선별된 기본 값들

GPL과 ggplot2 비교

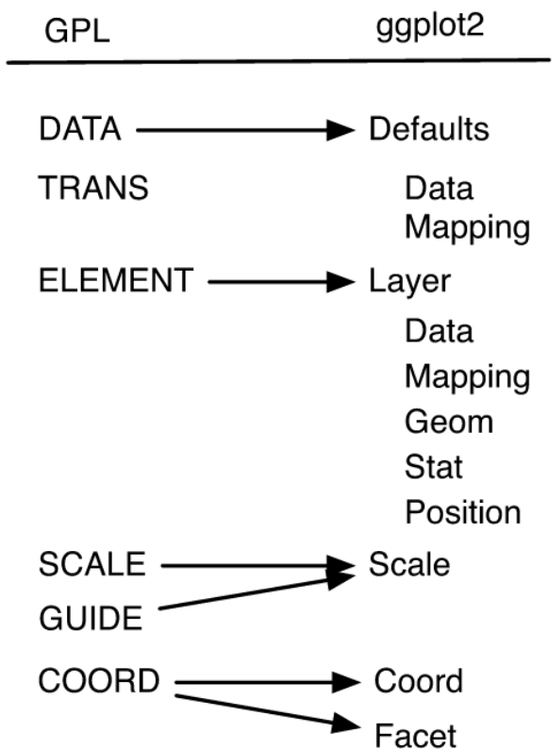


Figure 4. Mapping between components of Wilkinson's grammar (left) and the layered grammar (right). TRANS has no correspondence in `ggplot2`: its role is played by built-in R features.

[A Layered Grammar of Graphics - Hadley Wickham]

ggplot2의 한계

- 조합을 통한 통계적 그래프의 변주
  - 사용가능한 시각적 요소의 제한
  - 표현 자체를 발명하기는 어려움
- 매우 고수준 언어임
  - 작동방식에 개입하기가 쉽지 않음

더 저수준으로

그리고 Protovis

단어(차트)를 최대한 쪼개보자

막대그래프

단어

막대/그래프

의미

막/대/그/래/프

글자

차트(단어)를 뜯어보기

차트를 쪼개면 나오는 것

Rule (1)

- GPL / ggplot2
  - Data -> 블랙박스 -> 시각적 요소
  - 고유의 문법으로 기술(간접적, 블랙박스)
- Protovis / D3.js
  - Data -> 시각적 요소
  - 특히 D3.js에서는 직접 SVG 요소들을 다룸
  - 데이터가 어떻게 맵핑되는 지 직접 정의(직접적)

Rule (2)

- GPL / ggplot2
  - 이 데이터를 x좌표에 대응시켜줘
  - 내부적으로 적절히 처리됨
- Protovis / D3.js
  - 이 데이터로부터 이렇게 계산한 결과가 x 좌표야
  - 프로그래머가 직접 데이터와 시각적 요소를 제어

ex) 막대 그래프

막대 그래프를 구성하는 규칙들

막대그래프

- 막대는 0을 기준선으로 시작한다.
- 데이터의 크기는 막대의 높이를 결정한다
- 막대는 적절한 너비를 가진다
- 모든 막대의 너비는 같다
- 데이터의 순서는 막대의 위치를 결정한다
- Y축은 가장 0에서 가장 큰 값 + @로 한다
- …

이 규칙들을 직접 프로그램으로 작성

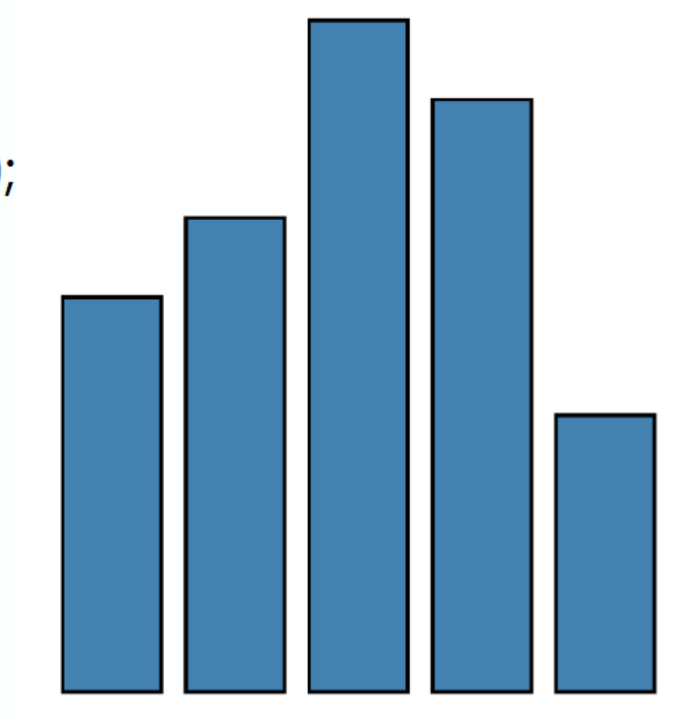
의외로 어려움

누구나 막대 그래프를 이해하지만 규칙을 알고 있는 경우는 드물다

아무도 안 가르쳐 줘요

막대그래프
Protovis Specification

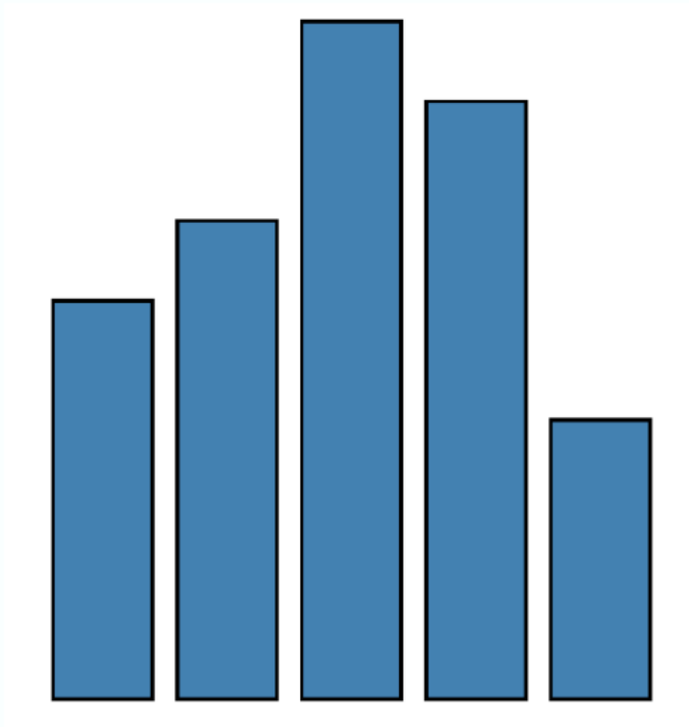
```
var vis = new pv.Panel();
vis.add(pv.Bar)
  .data([1, 1.2, 1.7, 1.5, 0.7])
  .visible(true)
  .left(function(d) this.index * 25);
  .bottom(0)
  .width(20)
  .height(function(d) d * 80)
  .fillStyle("blue")
  .strokeStyle("black")
  .lineWidth(1.5);
vis.render();
```



from CSE512

RECT $\lambda : D \rightarrow R$

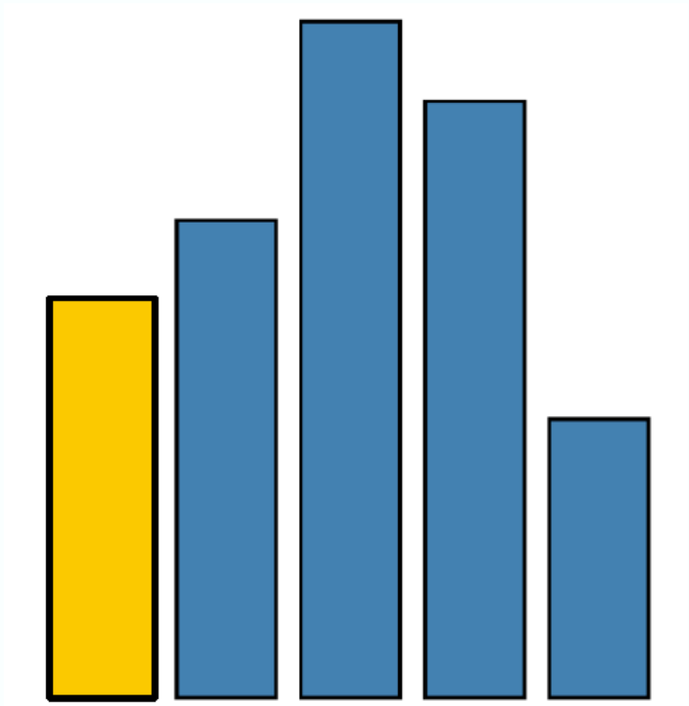
data	1	1.2	1.7	1.5	0.7
visible	true				
left	λ: index * 25				
bottom	0				
width	20				
height	λ: datum * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

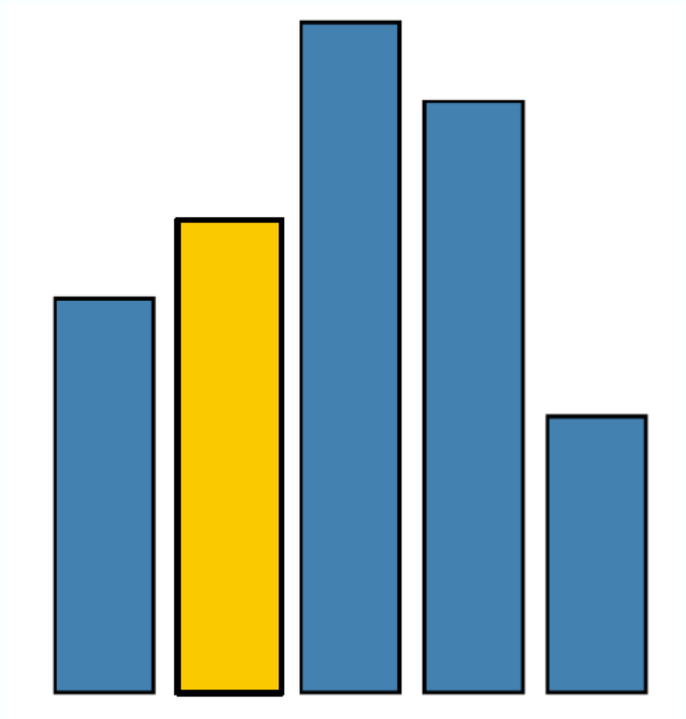
data	1	1.2	1.7	1.5	0.7
visible	true				
left	0 * 25				
bottom	0				
width	20				
height	1 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

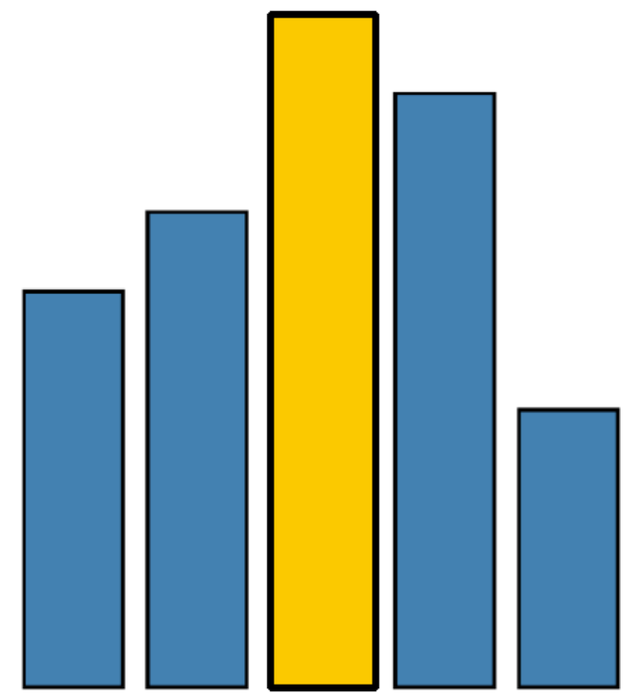
data	1	1.2	1.7	1.5	0.7
visible	true				
left	1 * 25				
bottom	0				
width	20				
height	1.2 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

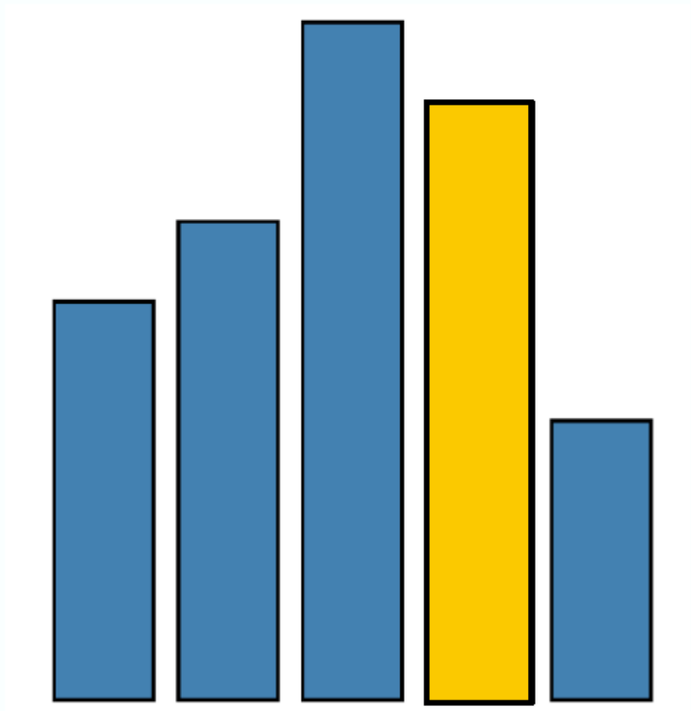
data	1	1.2	1.7	1.5	0.7
visible	true				
left	2 * 25				
bottom	0				
width	20				
height	1.7 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

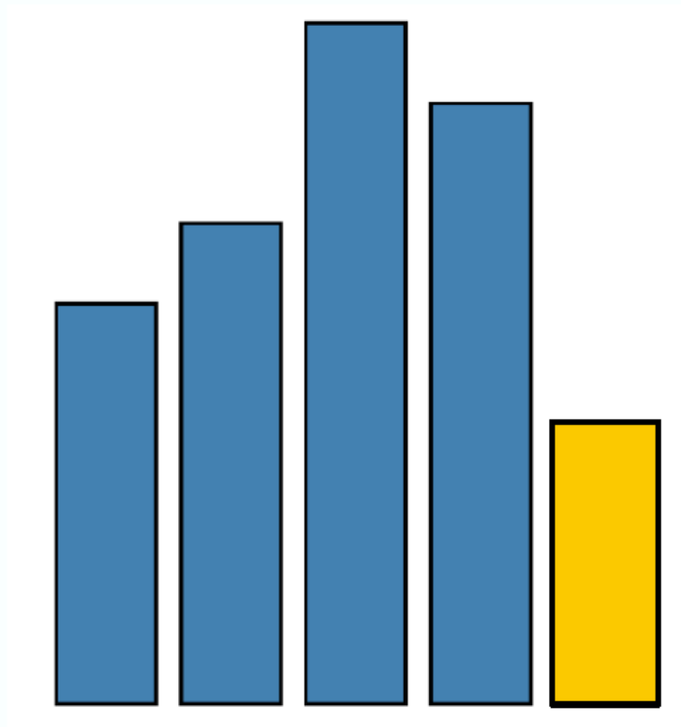
data	1	1.2	1.7	1.5	0.7
visible	true				
left	3 * 25				
bottom	0				
width	20				
height	1.5 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

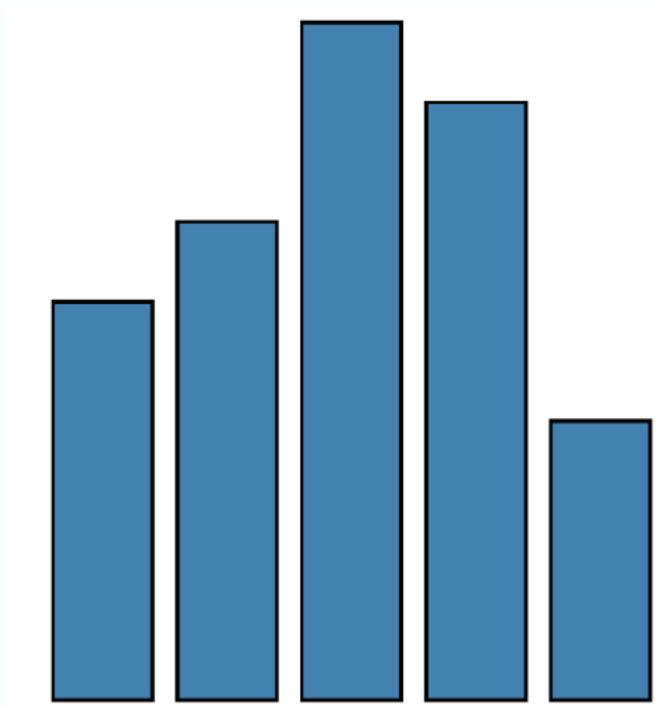
data	1	1.2	1.7	1.5	0.7
visible	true				
left	4 * 25				
bottom	0				
width	20				
height	0.7 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	λ: index * 25				
bottom	0				
width	20				
height	λ: datum * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

λ

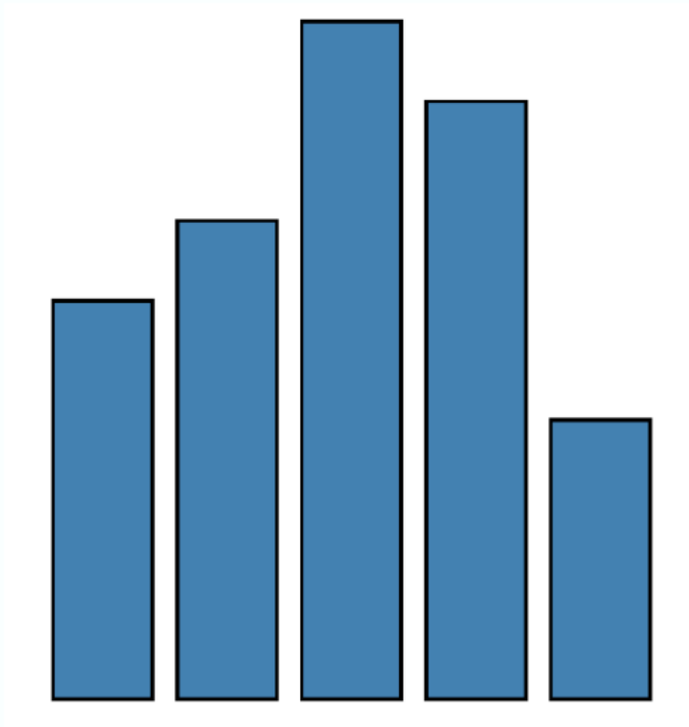
돈오의 순간

MARK	$\lambda : D \rightarrow R$
data	λ
visible	λ
left	λ
bottom	λ
width	λ
height	λ
fillStyle	λ
strokeStyle	λ
lineWidth	λ
...	λ

from CSE512

RECT $\lambda : D \rightarrow R$

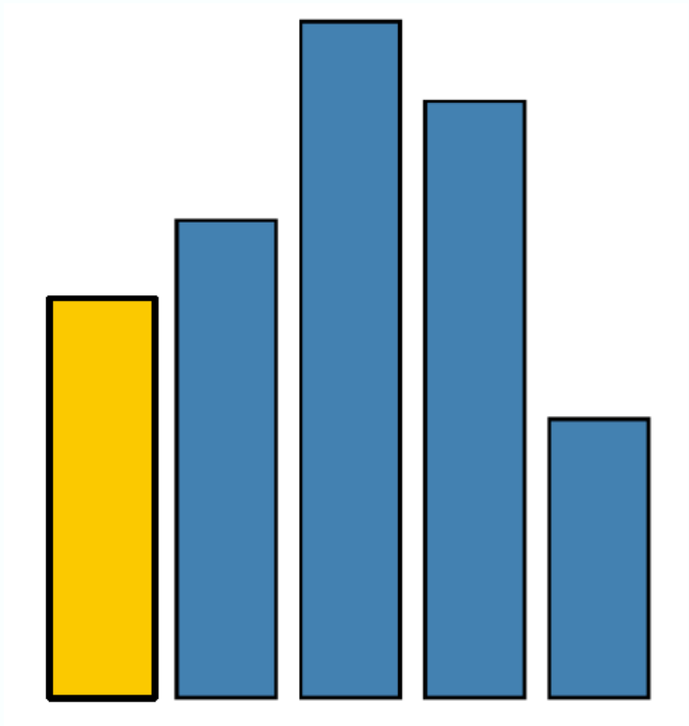
data	1	1.2	1.7	1.5	0.7
visible	true				
left	λ: index * 25				
bottom	0				
width	20				
height	λ: datum * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

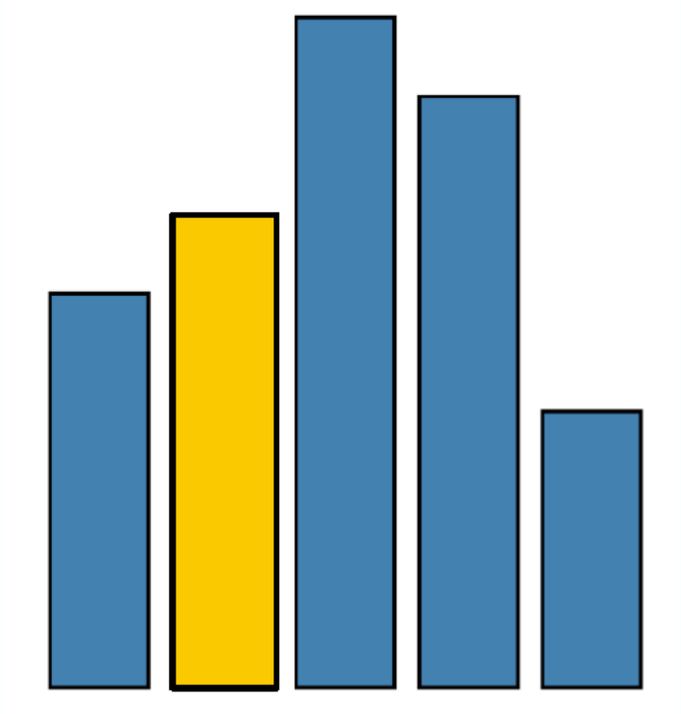
data	1	1.2	1.7	1.5	0.7
visible	true				
left	0 * 25				
bottom	0				
width	20				
height	1 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

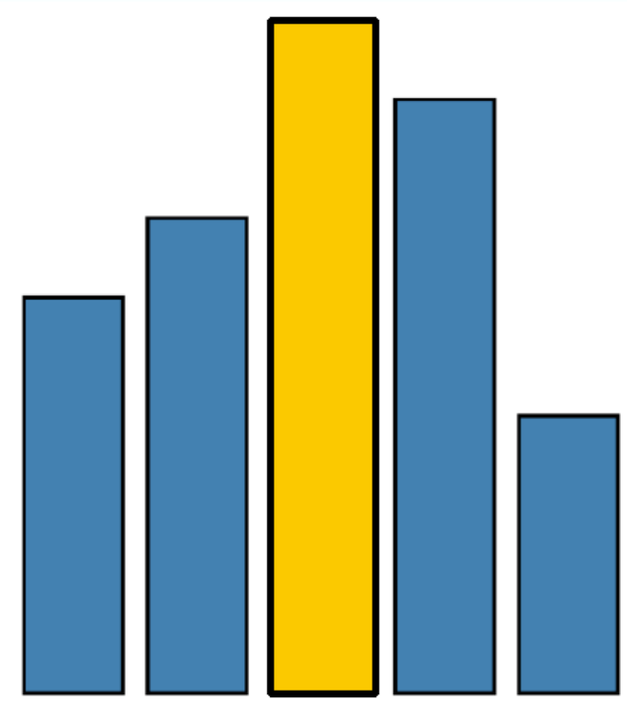
data	1	1.2	1.7	1.5	0.7
visible	true				
left	1 * 25				
bottom	0				
width	20				
height	1.2 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

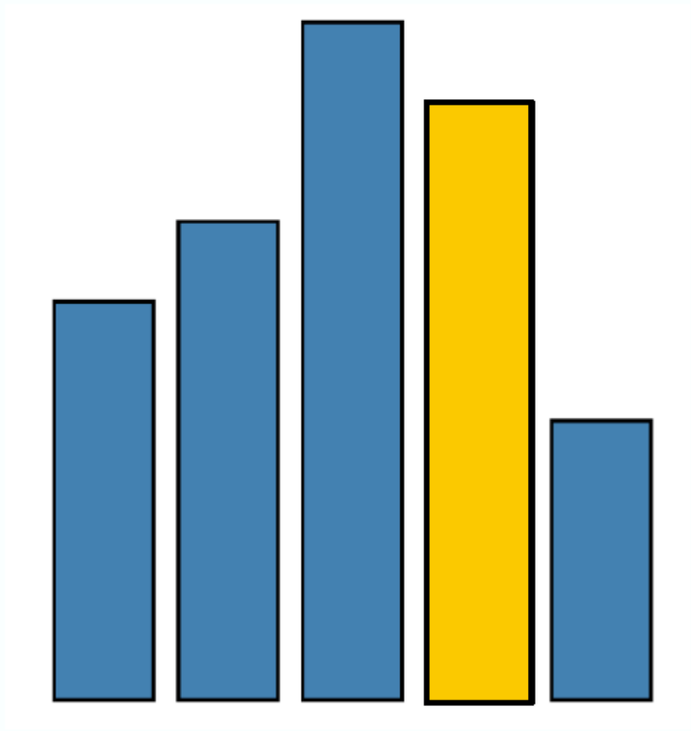
data	1	1.2	1.7	1.5	0.7
visible	true				
left	2 * 25				
bottom	0				
width	20				
height	1.7 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

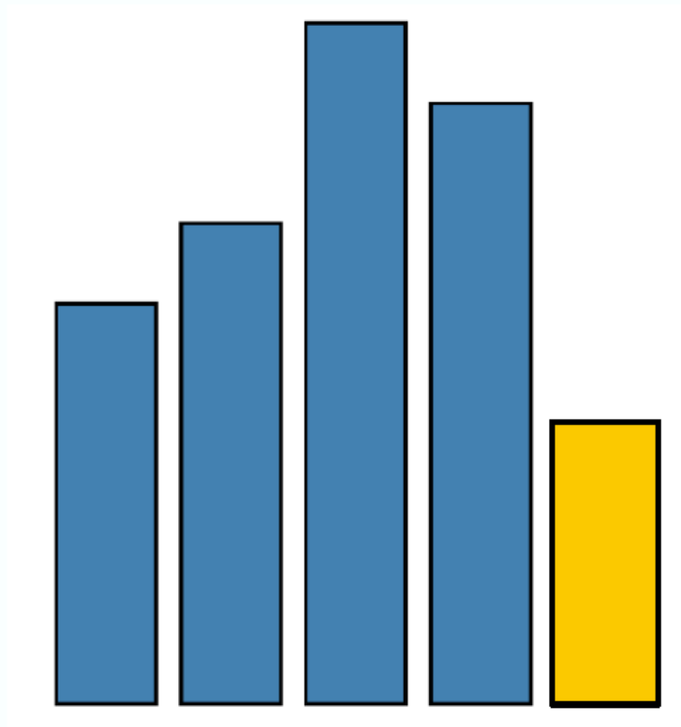
data	1	1.2	1.7	1.5	0.7
visible	true				
left	3 * 25				
bottom	0				
width	20				
height	1.5 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

RECT $\lambda : D \rightarrow R$

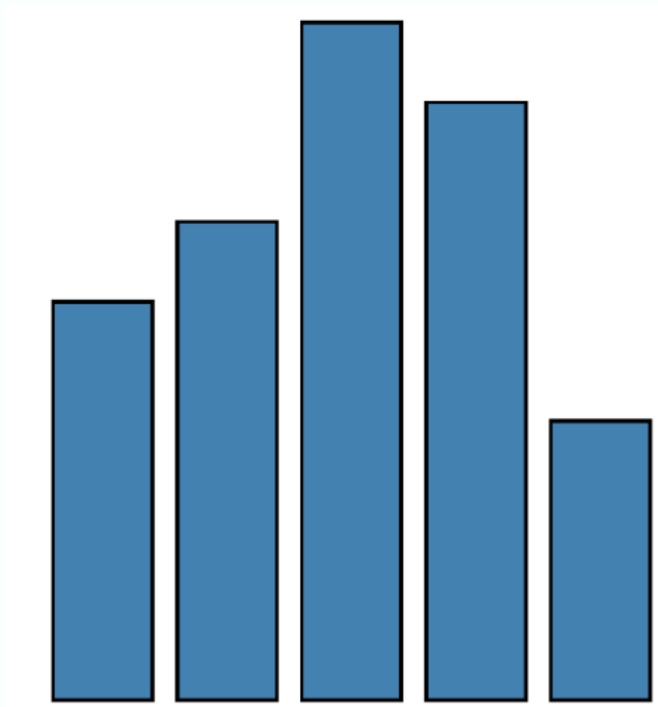
data	1	1.2	1.7	1.5	0.7
visible	true				
left	4 * 25				
bottom	0				
width	20				
height	0.7 * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

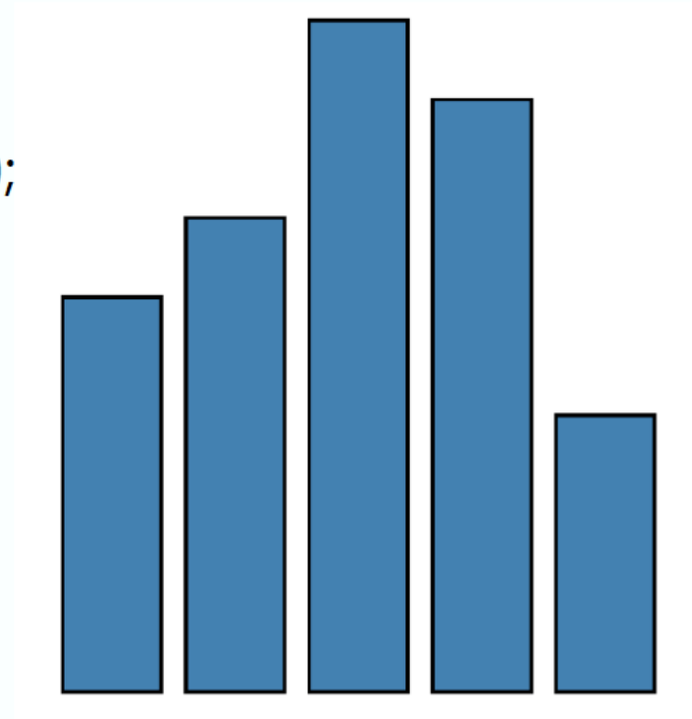
RECT $\lambda : D \rightarrow R$

data	1	1.2	1.7	1.5	0.7
visible	true				
left	λ: index * 25				
bottom	0				
width	20				
height	λ: datum * 80				
fillStyle	blue				
strokeStyle	black				
lineWidth	1.5				
...	...				



from CSE512

```
var vis = new pv.Panel();
vis.add(pv.Bar)
  .data([1, 1.2, 1.7, 1.5, 0.7])
  .visible(true)
  .left(function(d) this.index * 25);
  .bottom(0)
  .width(20)
  .height(function(d) d * 80)
  .fillStyle("blue")
  .strokeStyle("black")
  .lineWidth(1.5);
vis.render();
```



from CSE512

SVG의 세계로

Protovis to D3.js

Mick Bostock (1)

The Grammar of Graphics was also an influence, in terms of avoiding chart typologies while trying to be more declarative about the visual encoding, though certainly D3 ended up being much lower level and very different.

Reddit - Hi! I'm Mike Bostock ...

Mick Bostock (1)

시각적 인코딩에 대해 더 선언적인 라이브러리를 만드는 데 있어서 차트 유형학을 피한다는 점에서 *The Grammar of Graphics*의 영향도 받았다. 하지만 *D3*는 결과적으로 좀 더 저수준으로 만들어졌으며 많이 다르다.

Reddit - Hi! I'm Mike Bostock ...

Mick Bostock (2)

I wanted something that gave the designer greater control over the output—the kind of control that the early practitioners like Minard, Playfair and Bertin had because they did things by hand. Even within Protovis I felt like I was limited by its mark types; I wanted something that could use all of the DOM and SVG.

Reddit - Hi! I'm Mike Bostock ...

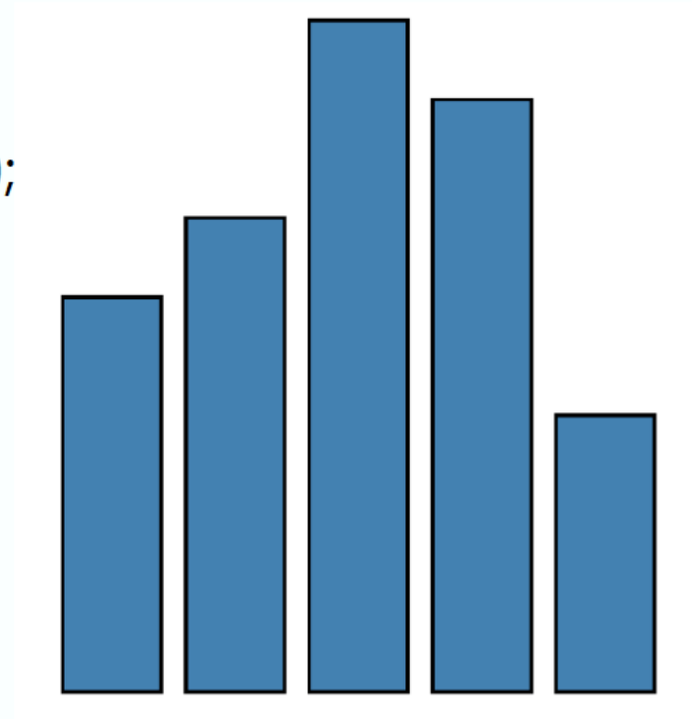
Mick Bostock (2)

찰스 미나드, 윌리엄 플레이페어, 자크 버틴이 손으로 시각화 작업을 하면서 더 많은 제어권을 가지고 있었듯이, 나는 디자이너들에게 출력에 대해 더 많은 제어권을 주고 싶었다. *Protovis*에서도 여전히 마크(*mark*) 타입들의 제약들을 느껴왔다. 그래서 나는 *DOM*과 *SVG*르 최대한 활용할 수 있는 방법을 원했다.

Reddit - Hi! I'm Mike Bostock ...

다시, 막대그래프

```
var vis = new pv.Panel();
vis.add(pv.Bar)
  .data([1, 1.2, 1.7, 1.5, 0.7])
  .visible(true)
  .left(function(d) this.index * 25);
  .bottom(0)
  .width(20)
  .height(function(d) d * 80)
  .fillStyle("blue")
  .strokeStyle("black")
  .lineWidth(1.5);
vis.render();
```



from CSE512

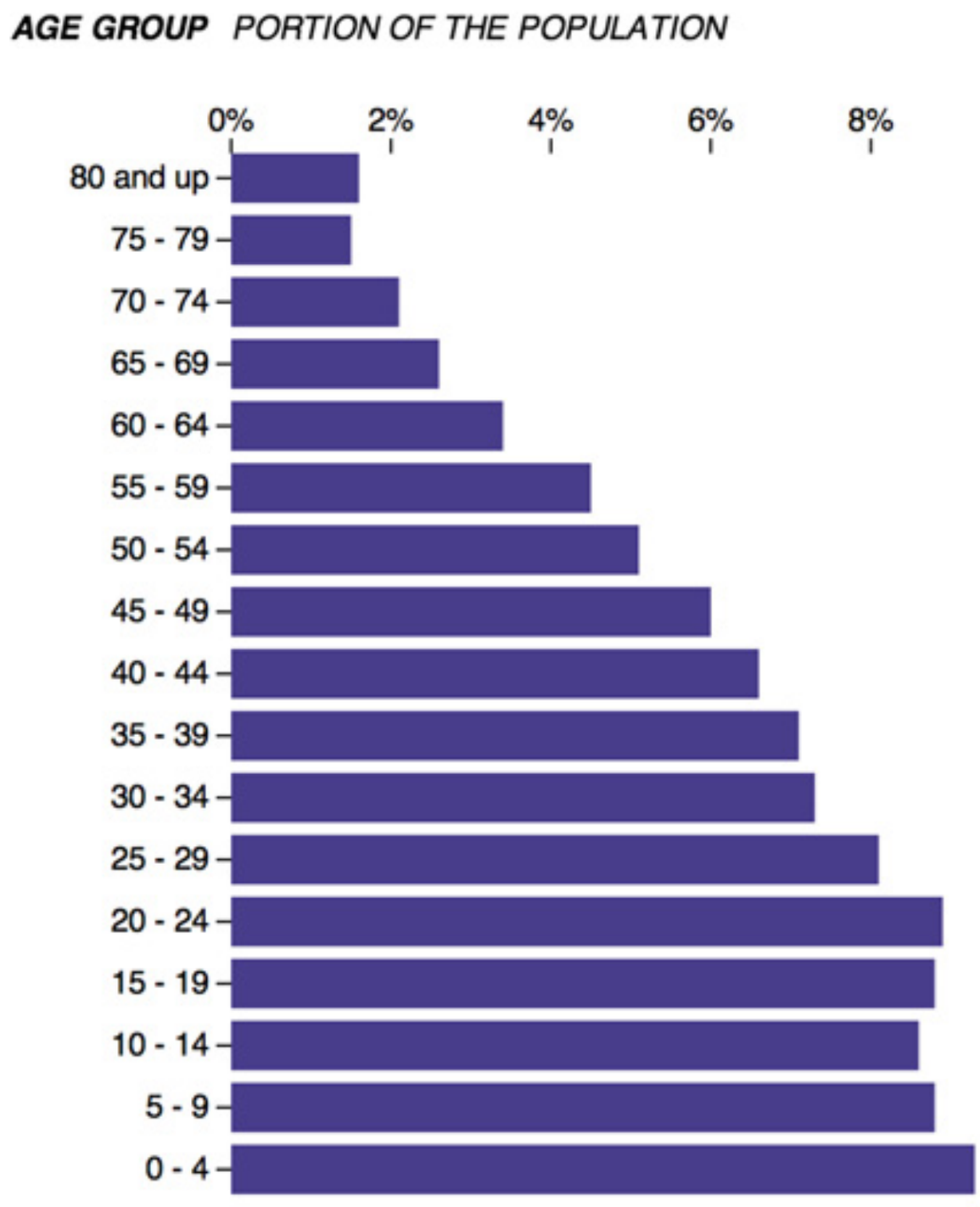
마가복음 1장

Data -> Svg 속성

```
barGroup.selectAll("rect")
    .data(popData)
    .enter()
    .append("rect")
    .attr("x", 0)
    .attr("y", function(d) {return y(d.age)})
    .attr("width", function(d) {return x(d.value)})
    .attr("height", y.rangeBand());
```

Visual Storytelling with D3 6장 예제

Age distribution of the world, 2010



D3.js의 한계 = SVG의 한계

거꾸로 말하면 DOM과 SVG를 잘 알아야함

다시, 정보시각화란 무엇인가?

규칙(집합)을 디자인하기
Declarative Language

시각화의 본질
규칙을 디자인하는 것

규칙들을 모으면

기성적인 차트
기성적인 차트의 변주
완전히 새로운 표현 방식

규칙이 정해지고 나면 디자이너는 결과물에 개입할 수 없다

시각화 디자이너란?

규칙을 정의하는 사람

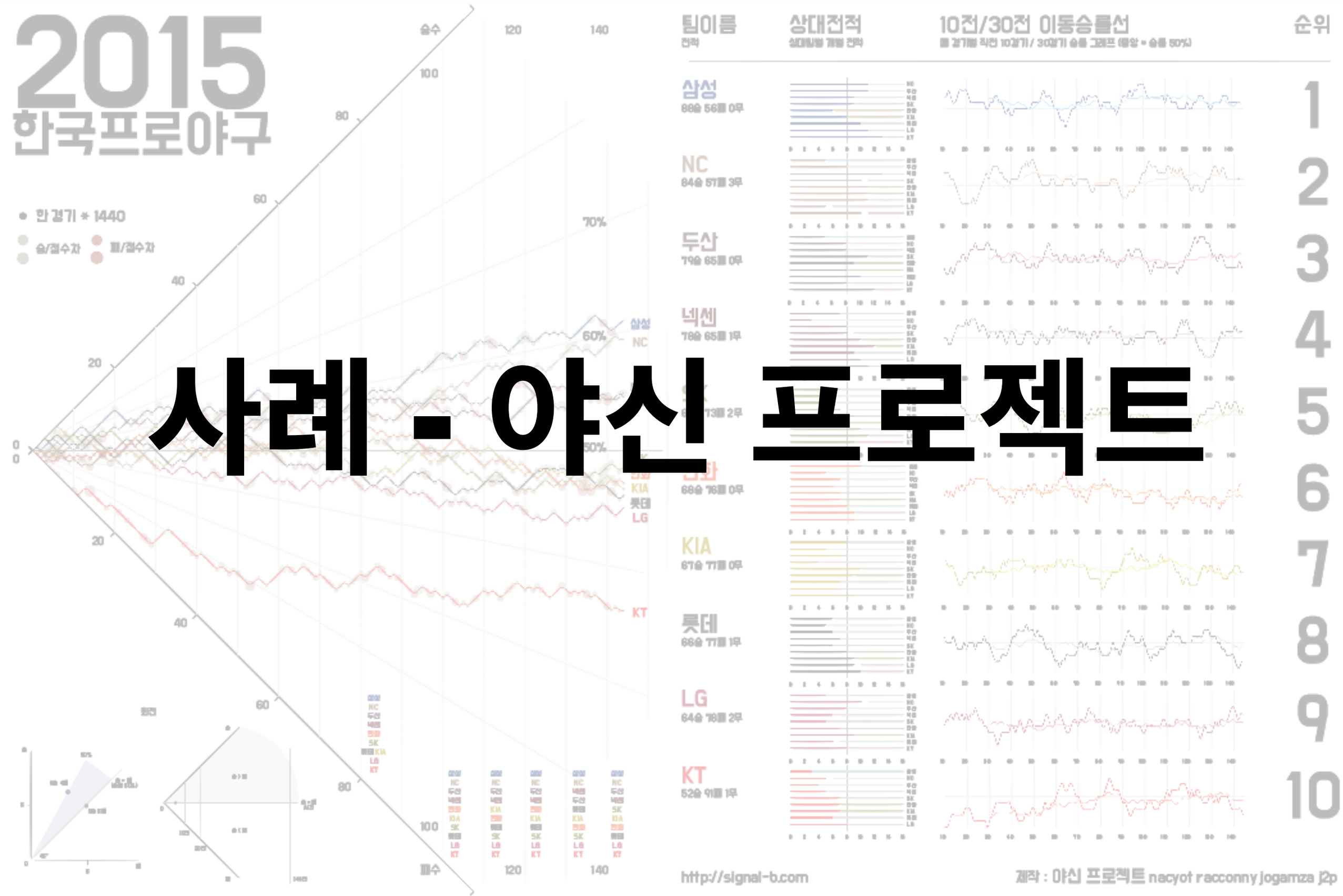
D3.js는 정보시각화에서 왜 중요한가?

D3

Data Driven Document

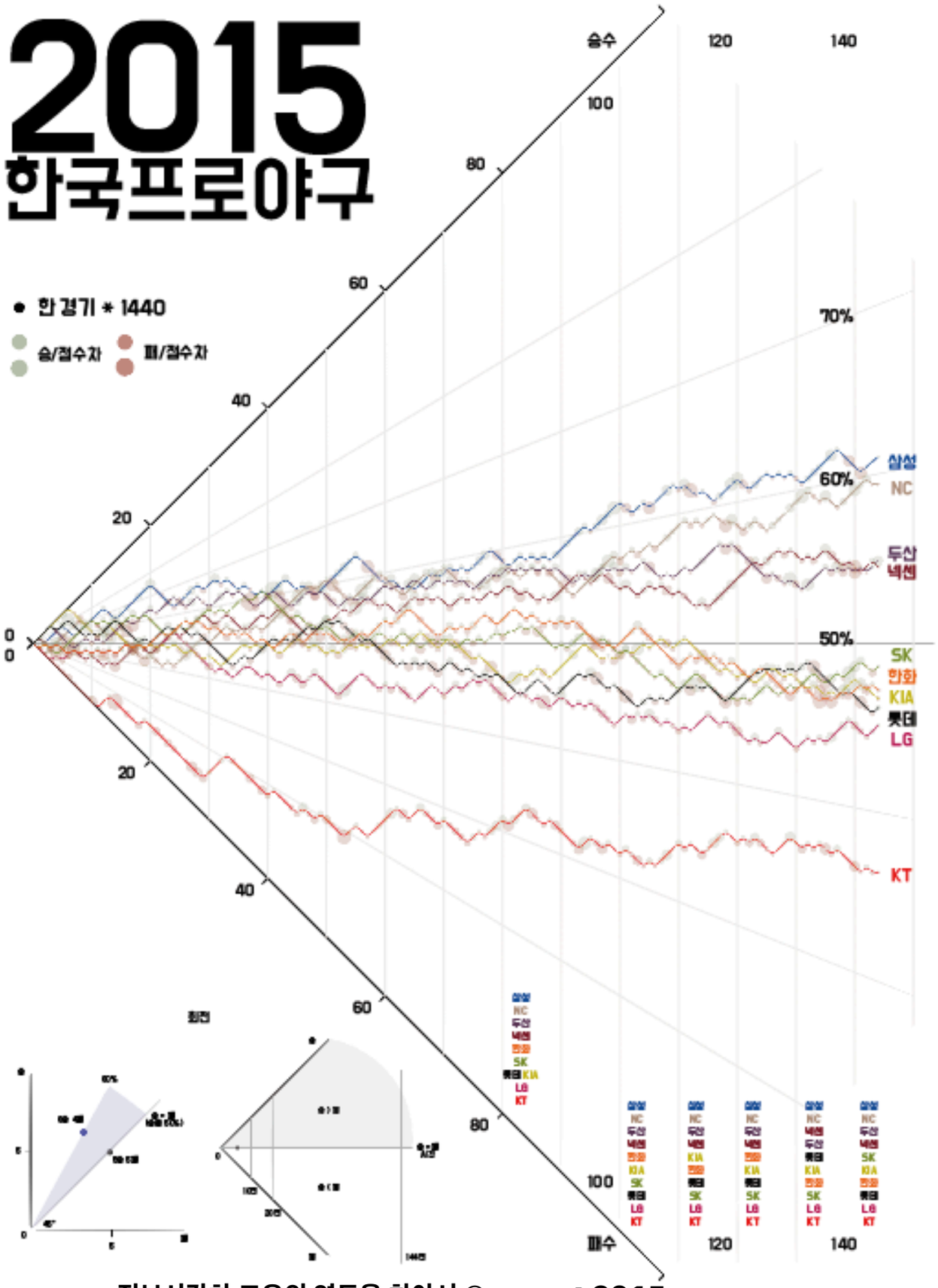
Declarative
Declarative
Declarative

시각화의 본질에 가장 적합
정보시각화 고유의 영토 위에서
규칙을 디자인하는 도구


2015
한국프로야구
팀이름
상대전적
10전/30전 이동승률선
순위
삼성
NC
두산
넥센
KIA
롯데
LG
KT
http://signal-b.com
제작 : 야신 프로젝트 nacyot racconny jogamza j2p

사례 - 야신 프로젝트

2015 한국프로야구

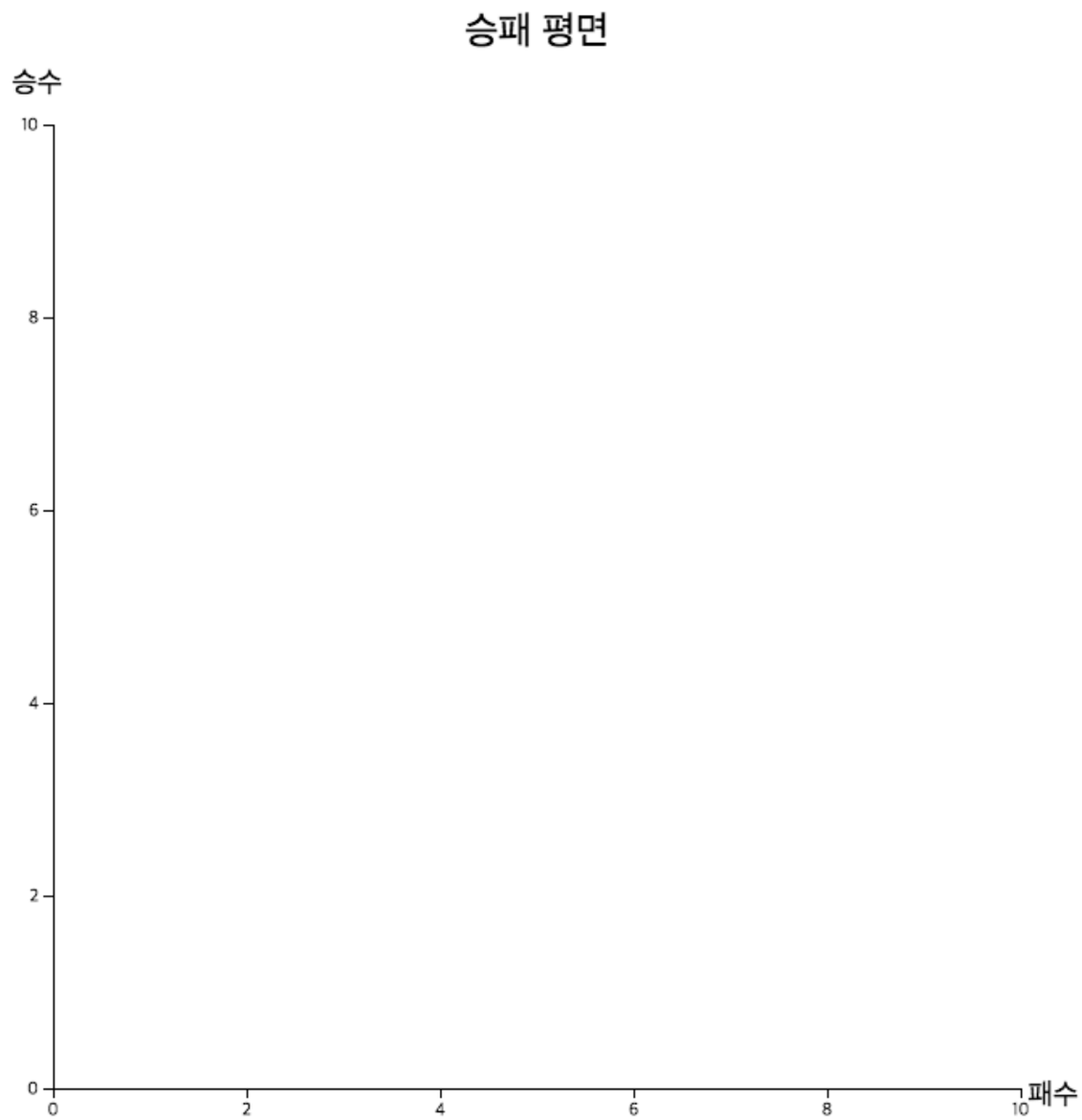


팀이름 / 전적	상대전적 / 상대팀별 개별 전적	10전/30전 이동승률선 / 매 경기별 직전 10경기 / 30경기 승률 그래프 (평균 = 승률 50%)	순위
삼성 88승 56패 0무			1
NC 84승 57패 3무			2
두산 79승 65패 0무			3
넥센 78승 65패 1무			4
SK 69승 73패 2무			5
한화 68승 76패 0무			6
KIA 67승 77패 0무			7
롯데 66승 77패 1무			8
LG 64승 78패 2무			9
KT 52승 91패 1무			10

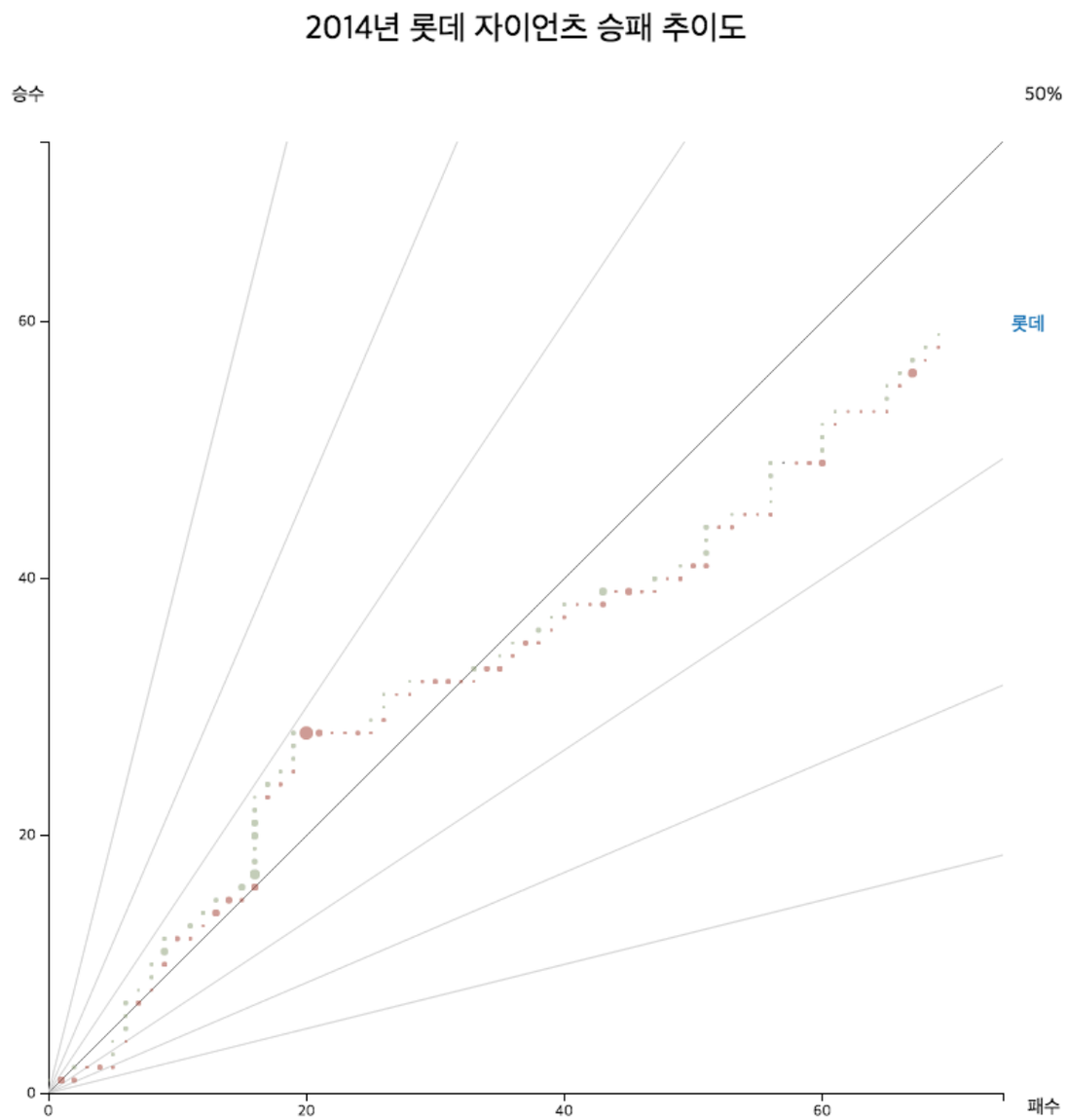
http://signal-b.com

제작 : 야신 프로젝트 nacyot racconny jogamza j2p

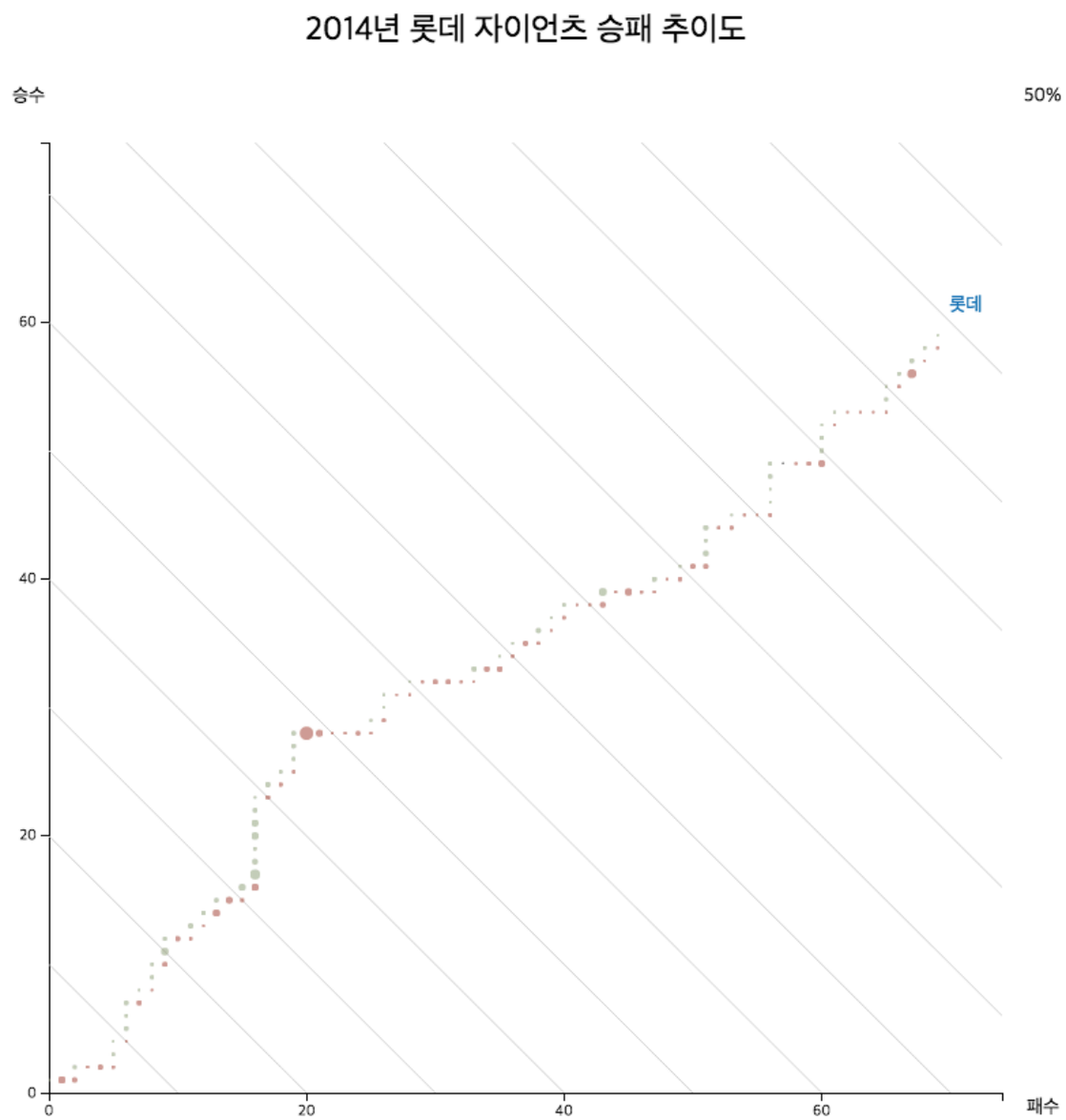
패승 평면



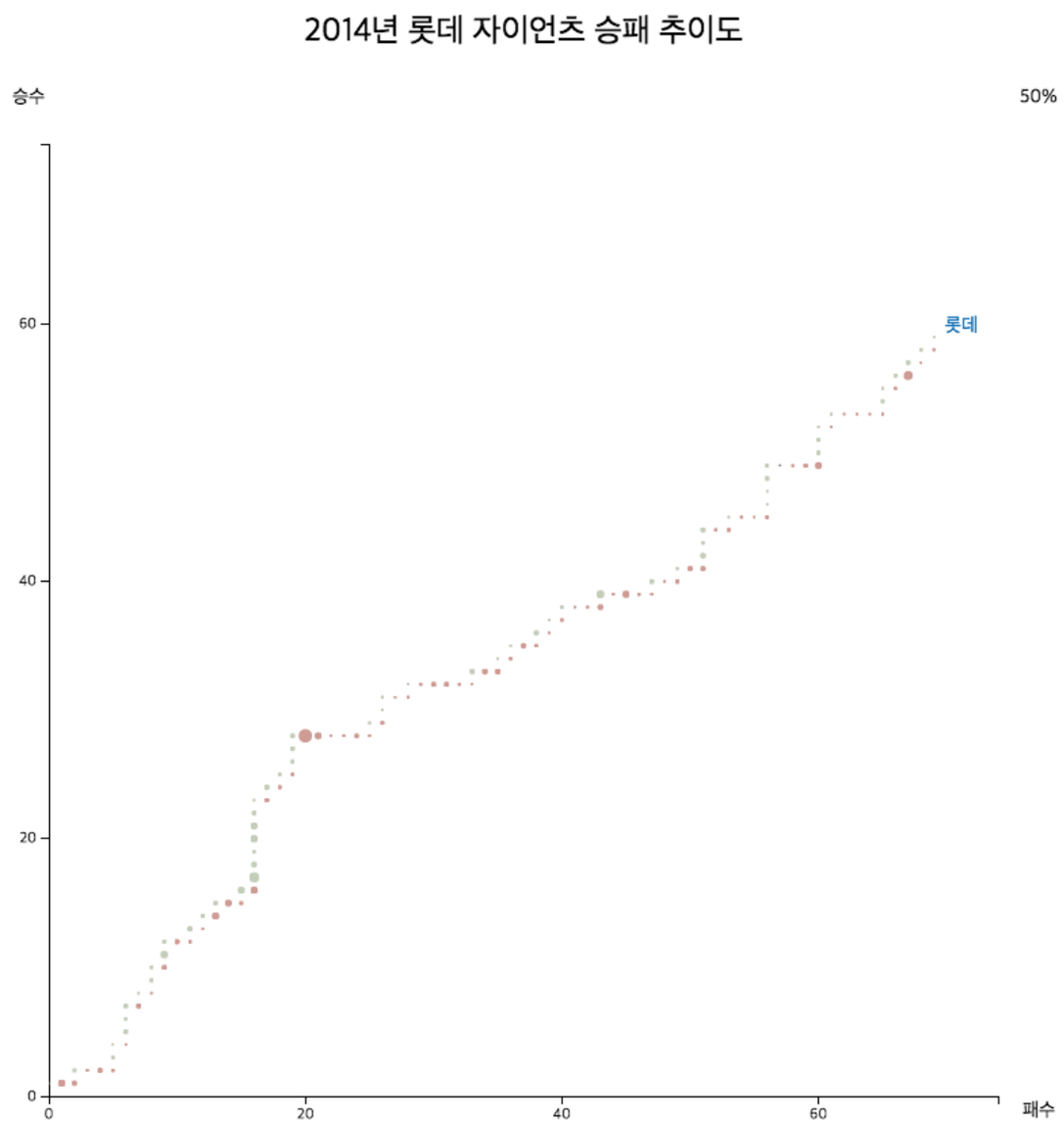
승률 공간



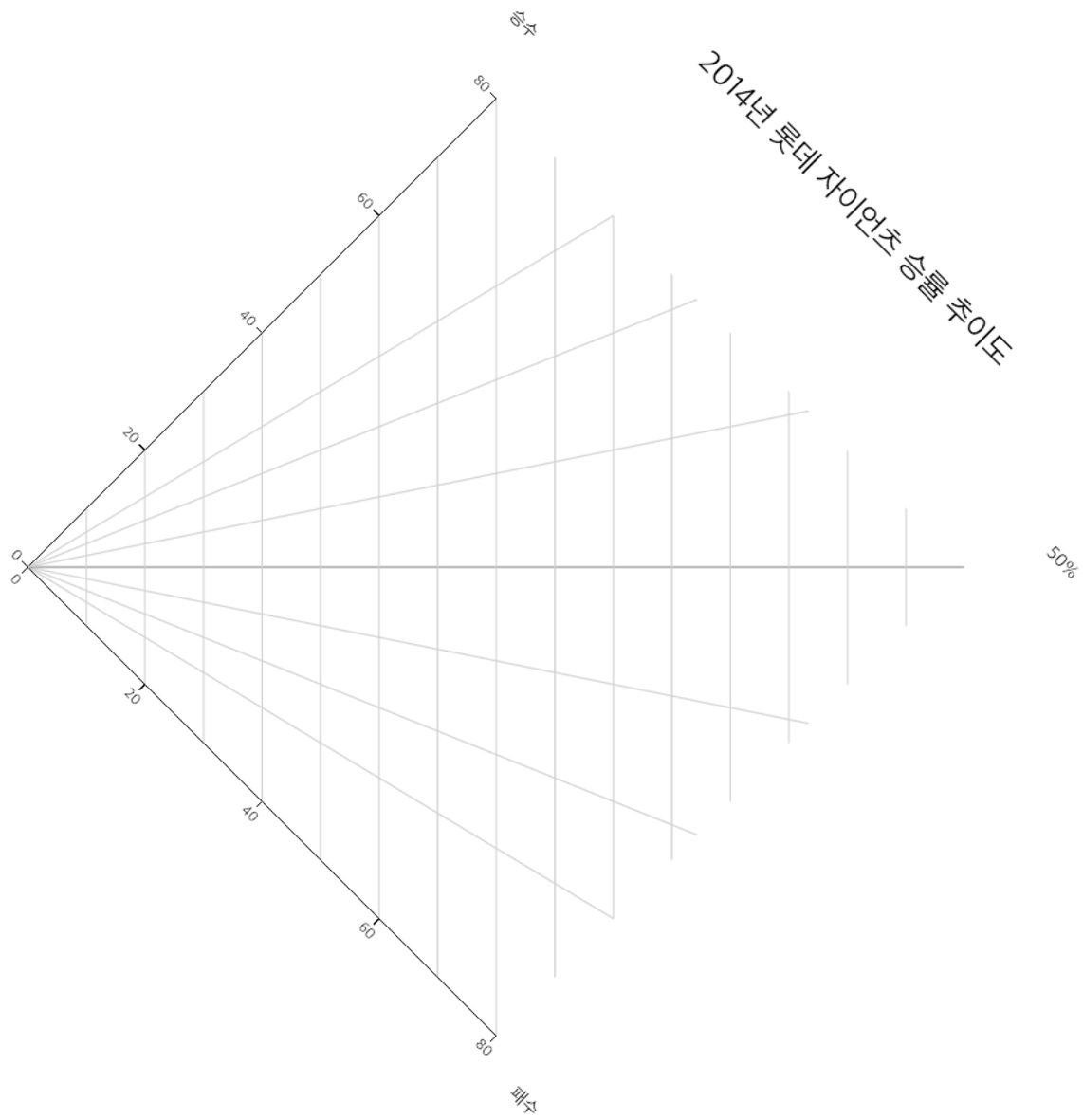
전적 가이드



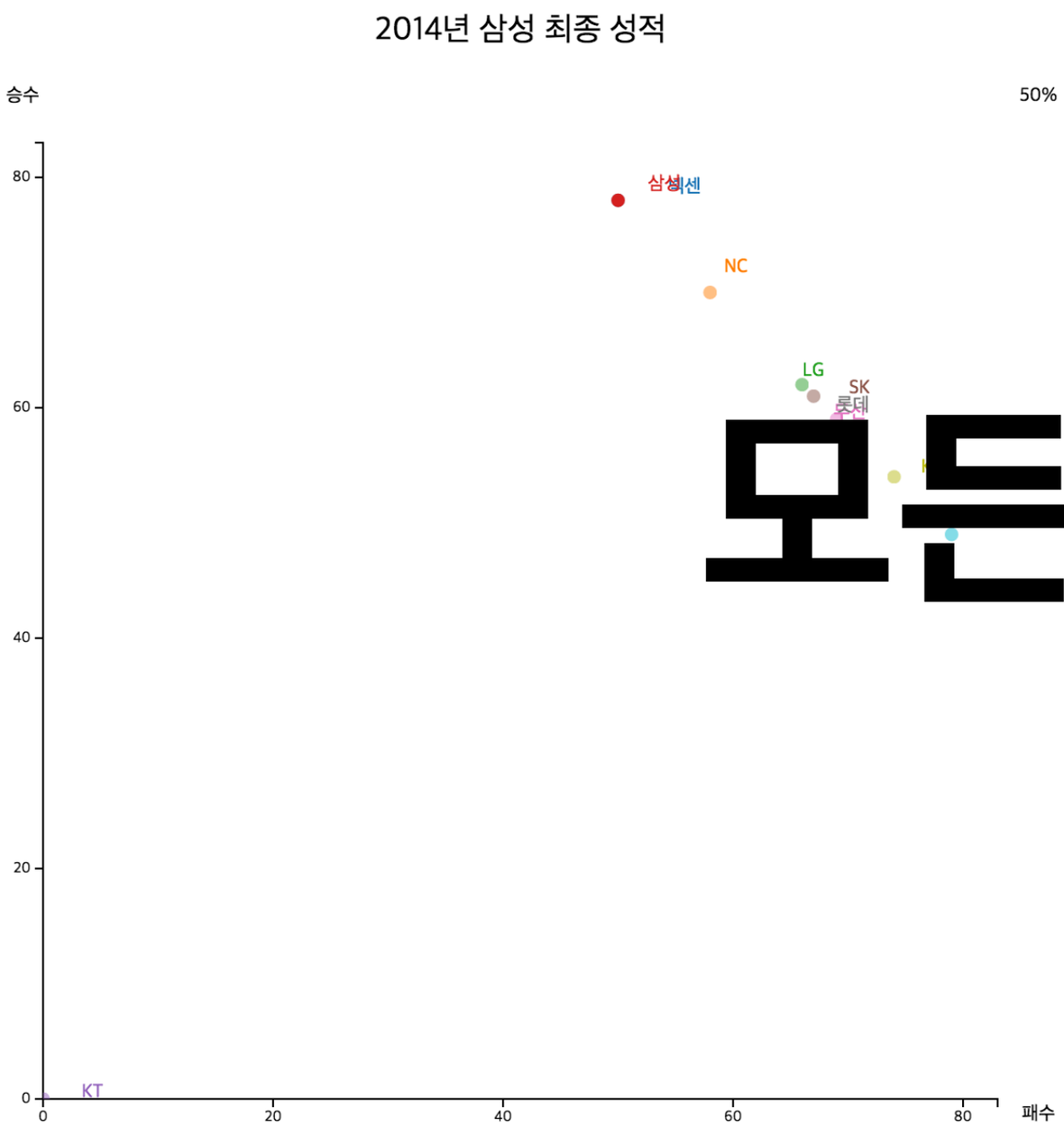
점의 크기

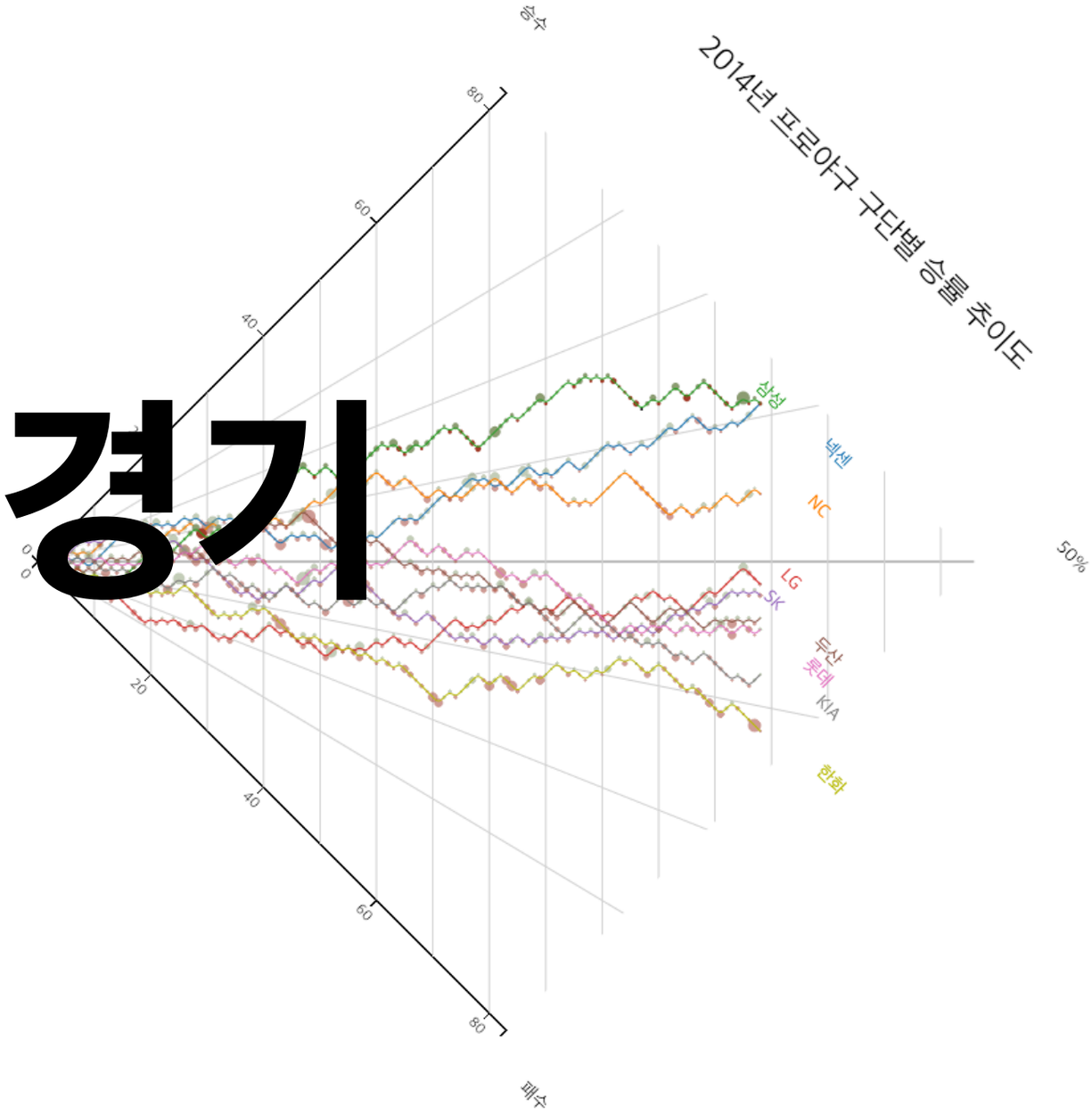


회전

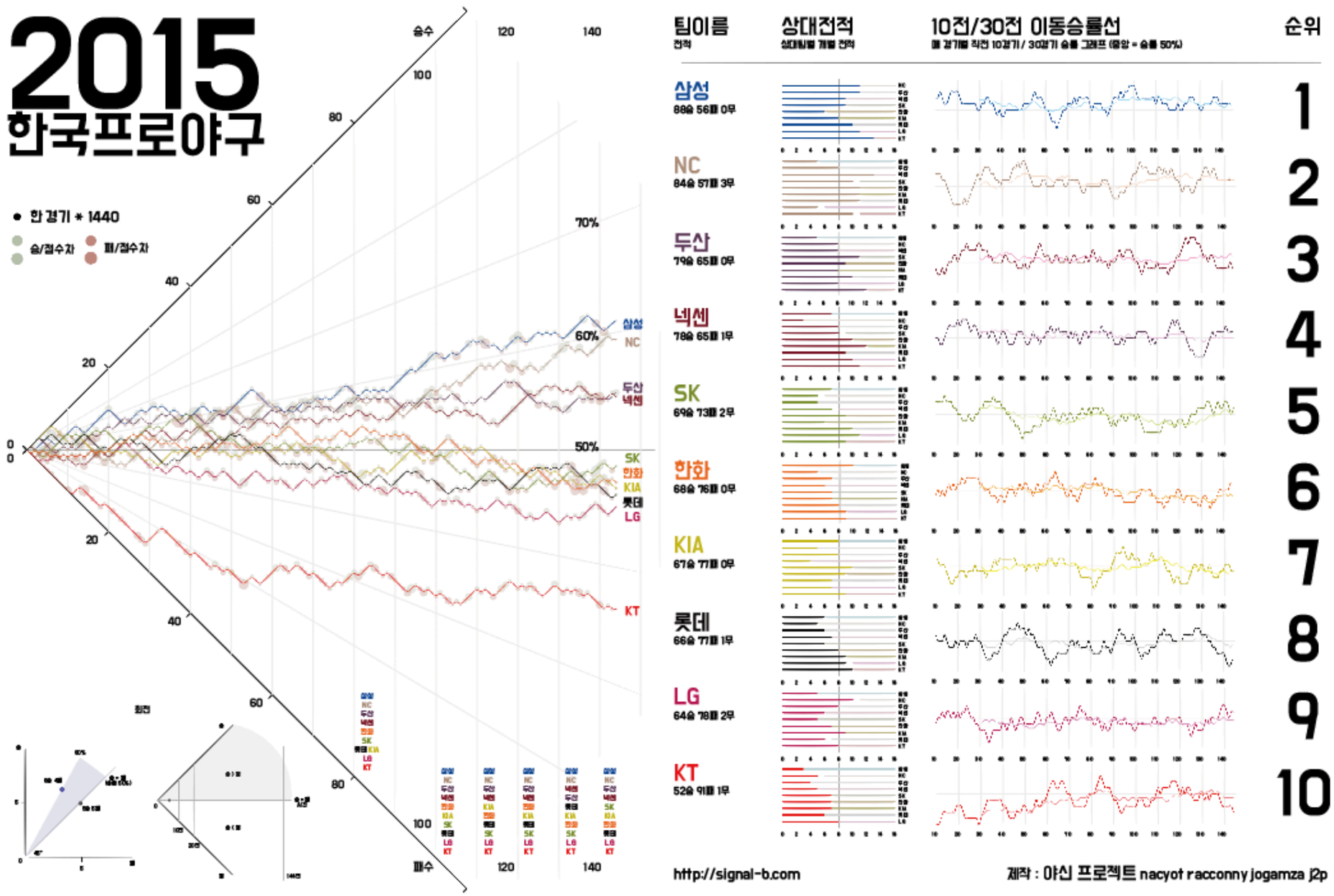


모든 경기


2014년 삼성 최종 성적
승수
50%
80
60
40
20
0
삼성
넥센
NC
LG
SK
롯데
KT
0
20
40
60
80
패수


2014년 프로야구 구단별 승률 추이도
승수
80
60
40
20
0
50%
LG
넥센
NC
SK
두산
롯데
KIA
한화
20
40
60
80
패수

This isn't a scatter plot



새로운 규칙 집합

$$Infinite * Infinite$$

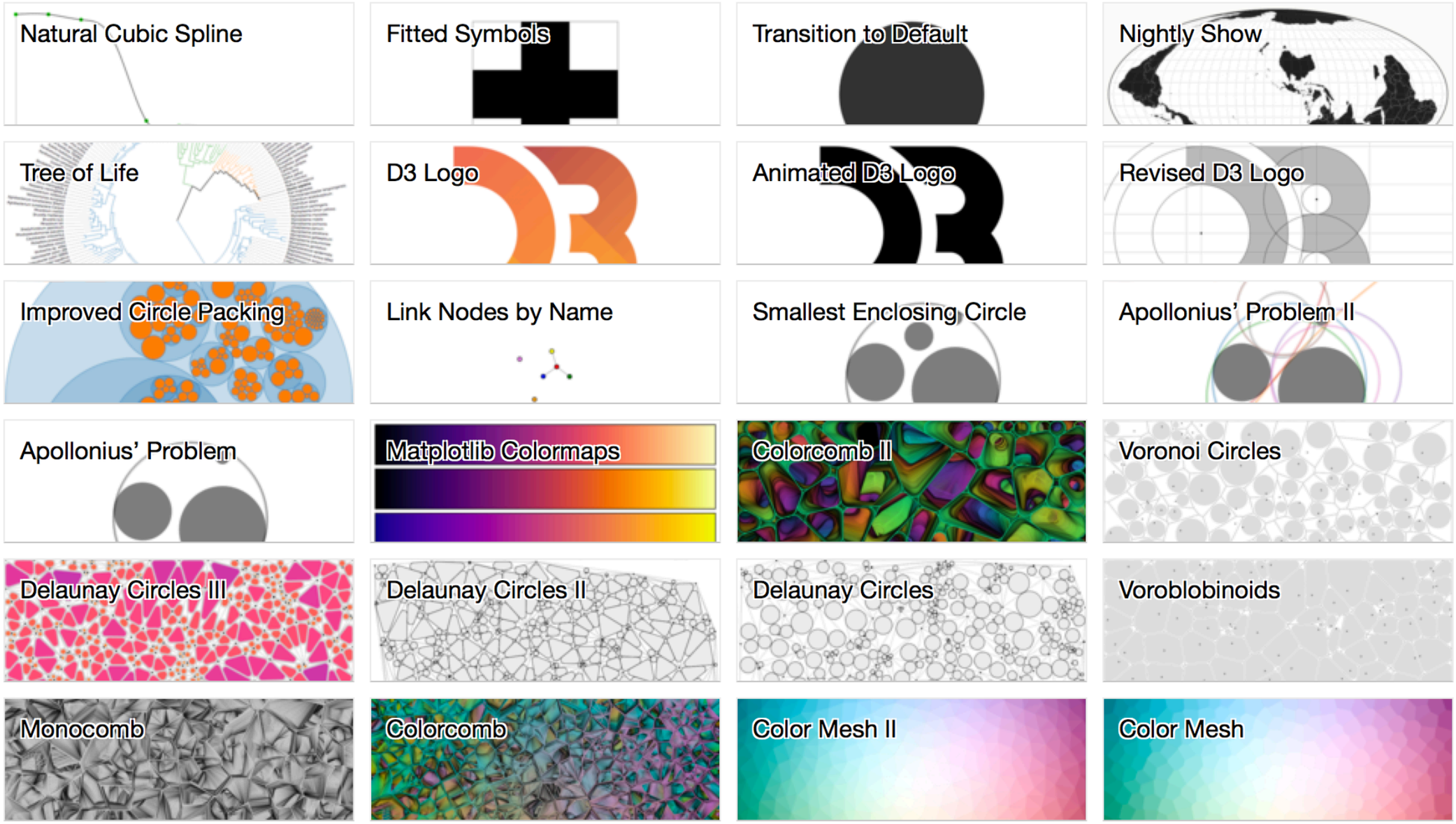
무한히 다양한 데이터셋
무한히 다양한 규칙 집합

WES ANDERSON
Obama
McCain
Kerry
Bush
Gore
30 sec

무명의 세계

규칙 집합만이 있을 뿐

mbostock's blocks

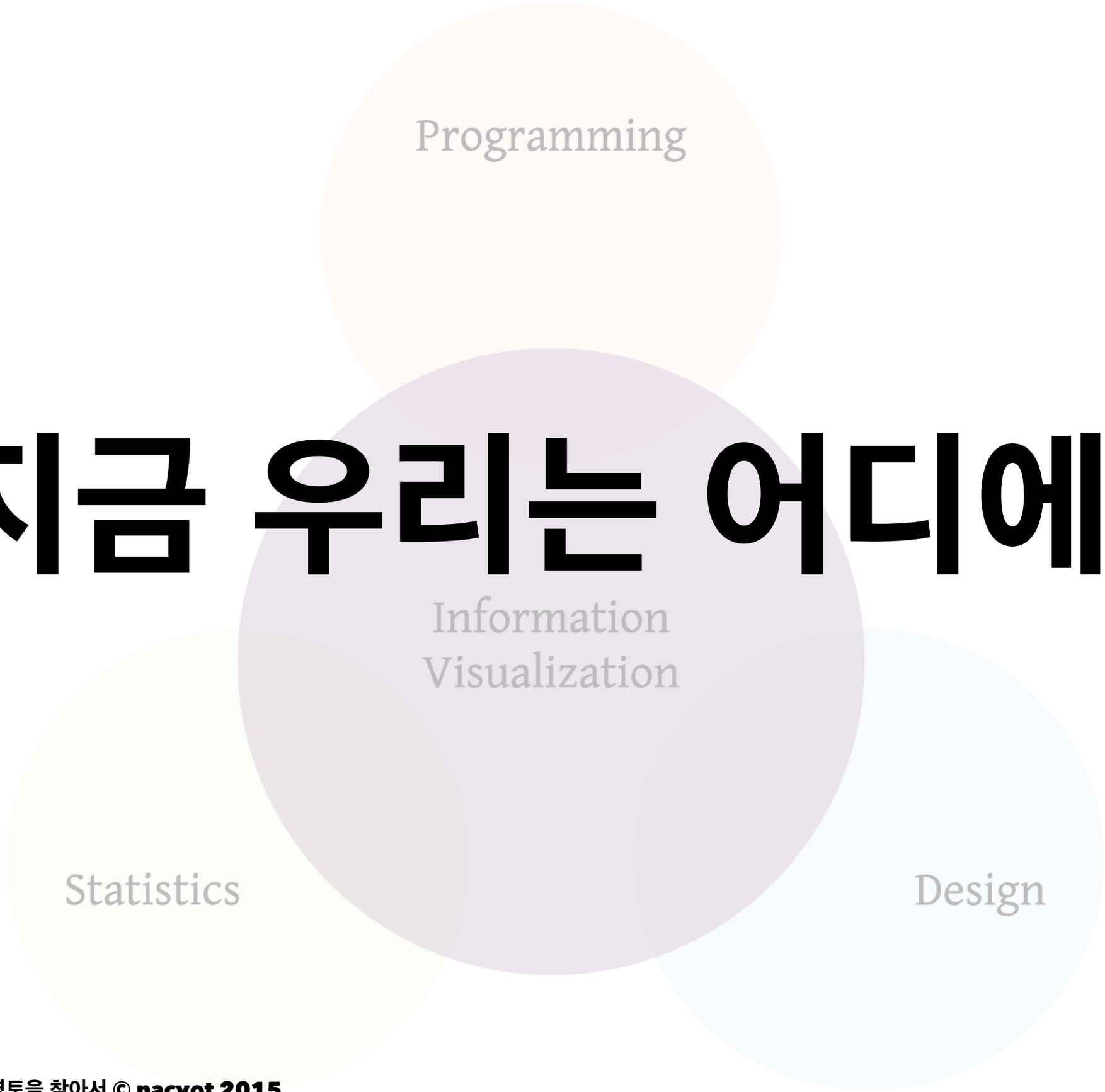


mbostock's blocks

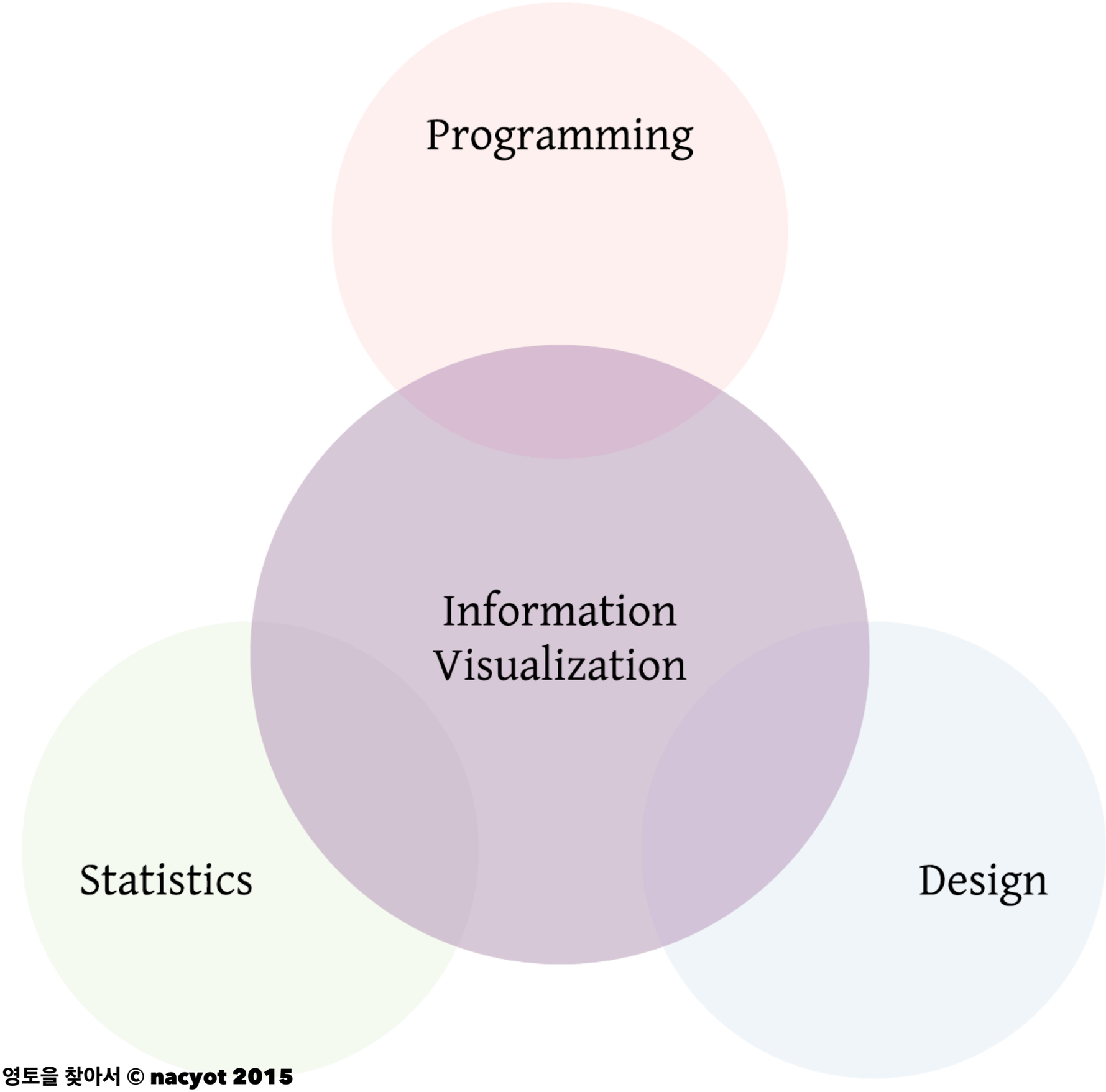
eyeo 2013 - For Example

좋은 시각화 사례는 다시 유형화

- Treemap
- Bubble Chart
- Chord Diagram
- Sunburst
- Calendar View
- Choropleth


Programming
Information
Visualization
Statistics
Design

지금 우리는 어디에


Programming
Information
Visualization
Statistics
Design

제프리 히어의 분류

Chart Typologies

Excel, Many Eyes, Google Charts

Charting Tools

Visual Analysis Grammars

VizQL, ggplot2, ***Vegalite***

Visualization Grammars

Protovis, D3.js, ***Vega***

Declarative Languages

Component Architectures

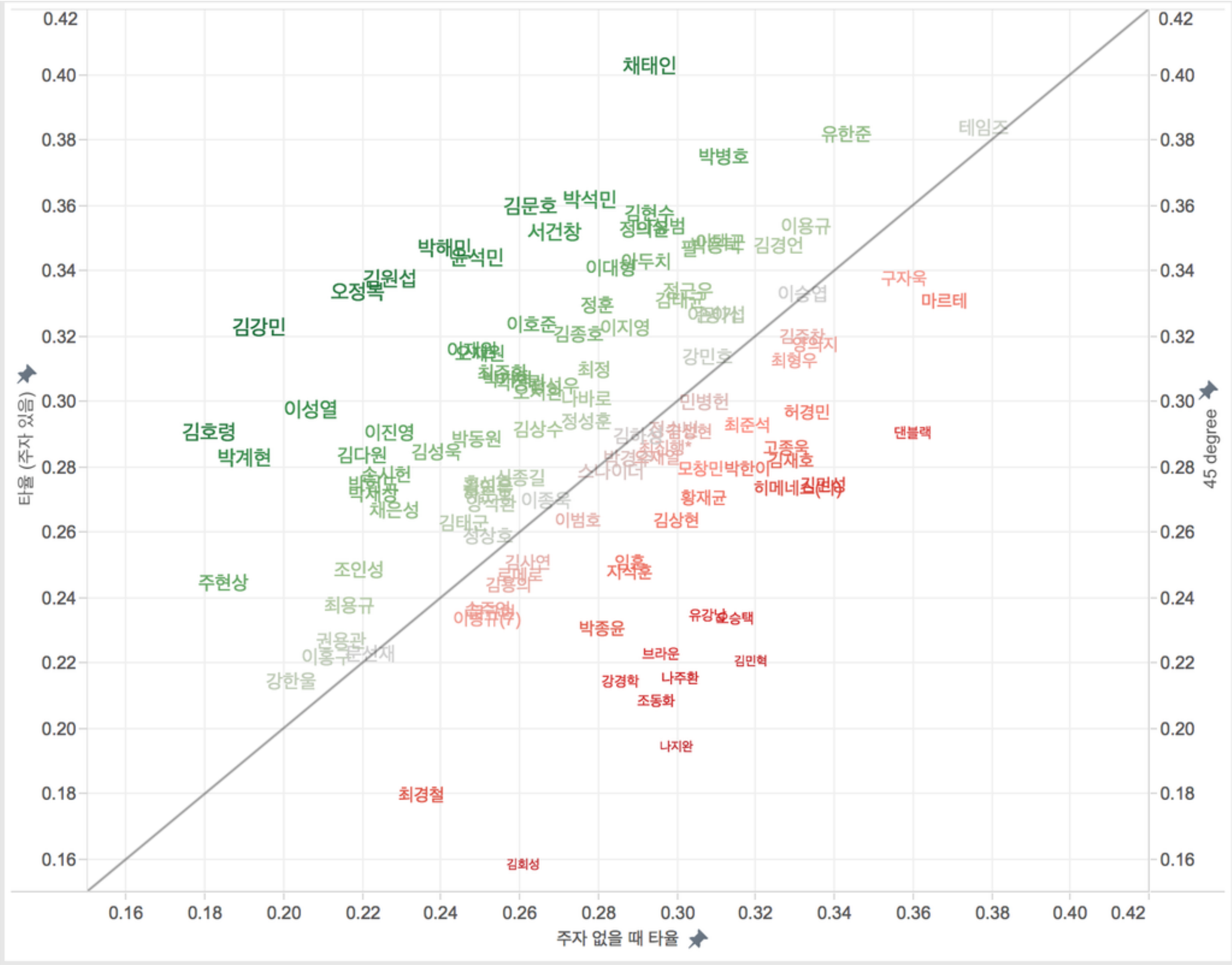
Prefuse, Flare, Improvise, VTK

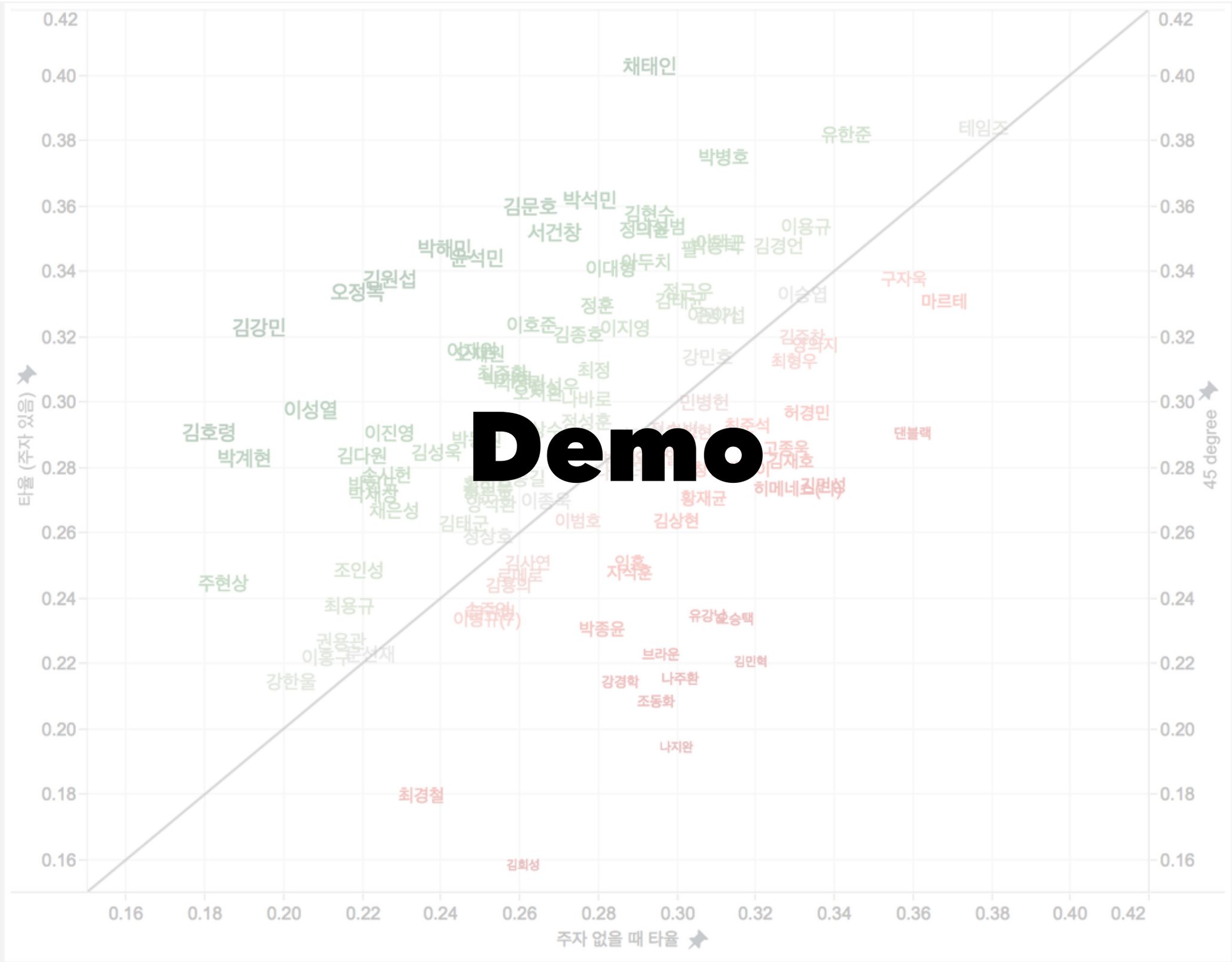
Programming Toolkits

Graphics APIs

Processing, OpenGL, Java2D

Tableau


타율 (주자 있음)
45 degree
주자 없을 때 타율
0.16
0.18
0.20
0.22
0.24
0.26
0.28
0.30
0.32
0.34
0.36
0.38
0.40
0.42
채태인
유한준
테임즈
박병호
김문호
박석민
서건창
이용규
김경언
박해민
윤석민
이대형
김원섭
오정복
구자욱
마르테
김강민
정훈
이호준
김종호
이지영
최정
강민호
최형우
이성열
민병헌
허경민
김호령
이진영
김상수
정성훈
최준석
박계현
김다원
김성욱
박동원
고종욱
김재호
모창민
박한이
황재균
김상현
이범호
주현상
조인성
최용규
박종윤
유강남
김민혁
브라운
강경학
나주환
조동화
나지완
최경철
김회성


Demo
주자 없을 때 타율
타율 (주자 있음)
45 degree
0.16
0.18
0.20
0.22
0.24
0.26
0.28
0.30
0.32
0.34
0.36
0.38
0.40
0.42
채태인
유한준
테임즈
박병호
김문호
박석민
서건창
이용규
김경언
구자욱
마르테
김강민
이호준
이지영
최정
이성열
김호령
이진영
박계현
김다원
김성욱
채은성
이범호
김상현
황재균
허경민
댄블랙
주현상
조인성
최용규
박종윤
브라운
김민혁
강경학
나주환
조동화
나지완
최경철
김회성

Interactive Data Exploration

Tableau, ***Lyra, Polestar, Voyager***

Graphical Interfaces

Visual Analysis Grammars

VizQL, ggplot2, ***Vegalite***

Visualization Grammars

Protovis, D3.js, ***Vega***

Declarative Languages

Component Architectures

Prefuse, Flare, Improvise, VTK

Programming Toolkits

Graphics APIs

Processing, OpenGL, Java2D

Thank you!

@nacyo_t