

# 셰이더 프로그래밍

Lecture 1

이택희

# 수업에서 다룰 내용

- ◆ 렌더링을 위해 필요한 기본적인 요소들에 대해 다시 상기해보자
- ◆ 그래픽스 파이프라인에 대해 다시 상기해보자
  - ◆ 그래픽스 파이프라인이란?
  - ◆ 파이프라인의 각 단계에서 수행하는 작업

# 수업에서 다룰 내용

- ◇ 개발 환경 구축
  - ◇ Visual studio Community
  - ◇ Sample project 빌드 및 실행 확인

렌더링에 필요한 기본 정보

# 렌더링에 필요한 기본 정보

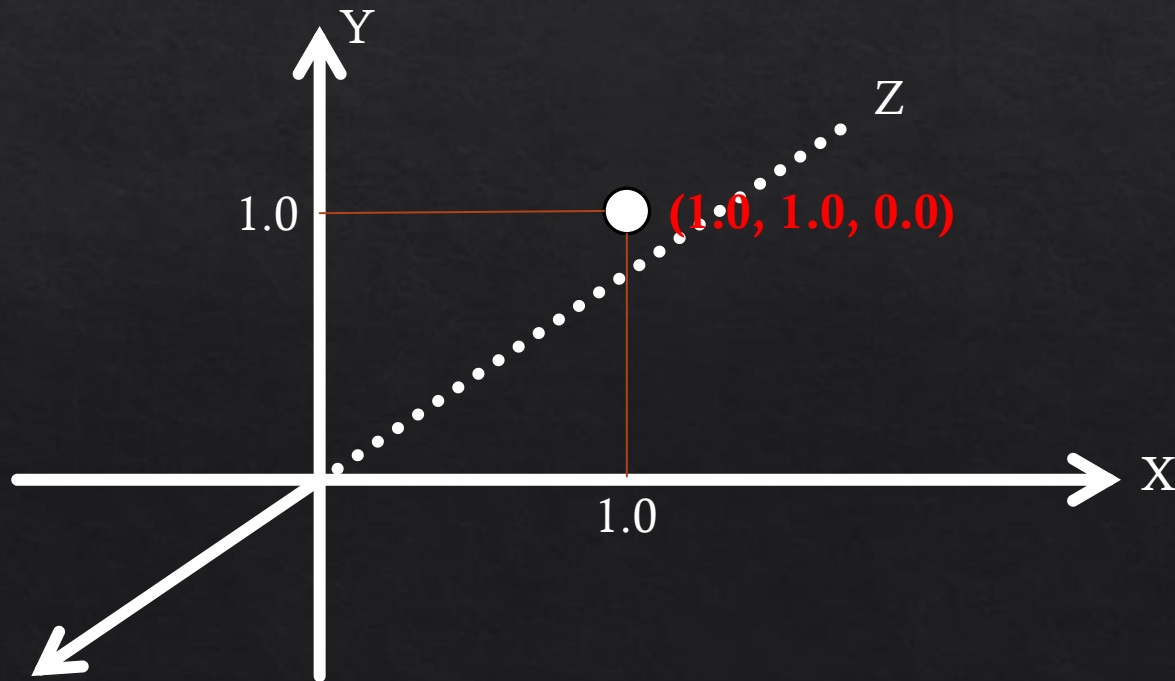
◆ 쉽지만 중요한 기본 지식

◆ 렌더링을 위해서 기본적으로 필요한 것들이 무엇일까?  
Vertex(정점) 컴, 선, 면

<u>point</u>	vs	<u>Vertex</u>
위치점 또는 개선점		모든의 한 점으로서 위치뿐만 아니라 다양한 정보를 가진다.

# 렌더링에 필요한 기본 정보

- ◆ 점 (하나의 정점으로 이루어짐)
  - ◆ 삼차원 공간상에 정의되는 점.
  - ◆ x, y, z 값을 가진다.





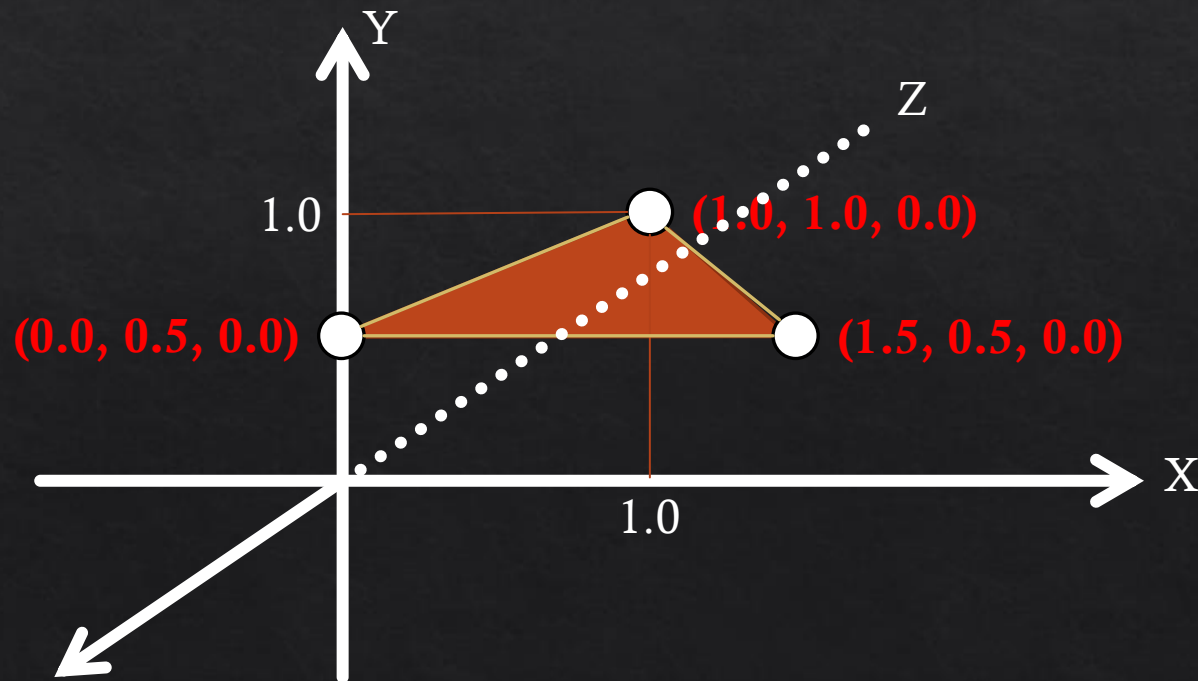
# 렌더링에 필요한 기본 정보

- ◆ 선 (두개의 정점으로 이루어짐)
  - ◆ 삼차원 공간상에 정의되는 선
  - ◆ 최소 두 개의 점들로 이루어 짐



# 렌더링에 필요한 기본 정보

- ◇ 삼각형 (세개의 정점으로 이루어짐)
  - ◇ 삼차원 공간상에 정의되는 면
  - ◇ 세 개의 점들로 이루어 짐





# 렌더링에 필요한 기본 정보

◇ 점 → Point

`glDrawArrays(GL_POINTS, 0, 1)`

◇ 선 → Line

`glDrawArrays(GL_LINES, 0, 2)`

◇ 삼각형 → Triangle

`glDrawArrays(GL_TRIANGLES, 0, 3)`

◇ 정점 → Vertex

◇ 요소 → Primitive

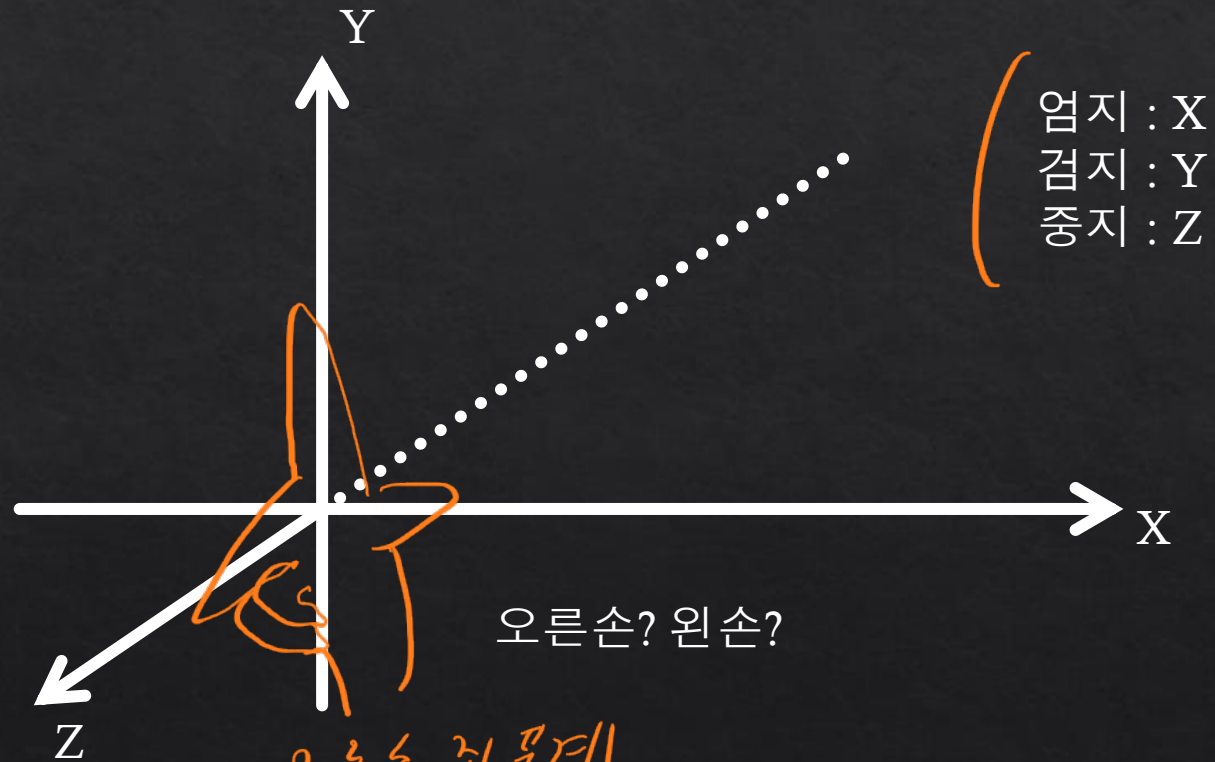


OpenGL APIs

OpenGL API 는 영어 기반이므로  
영어로 알아두는 게 훨씬 편함

# 렌더링에 필요한 기본 정보

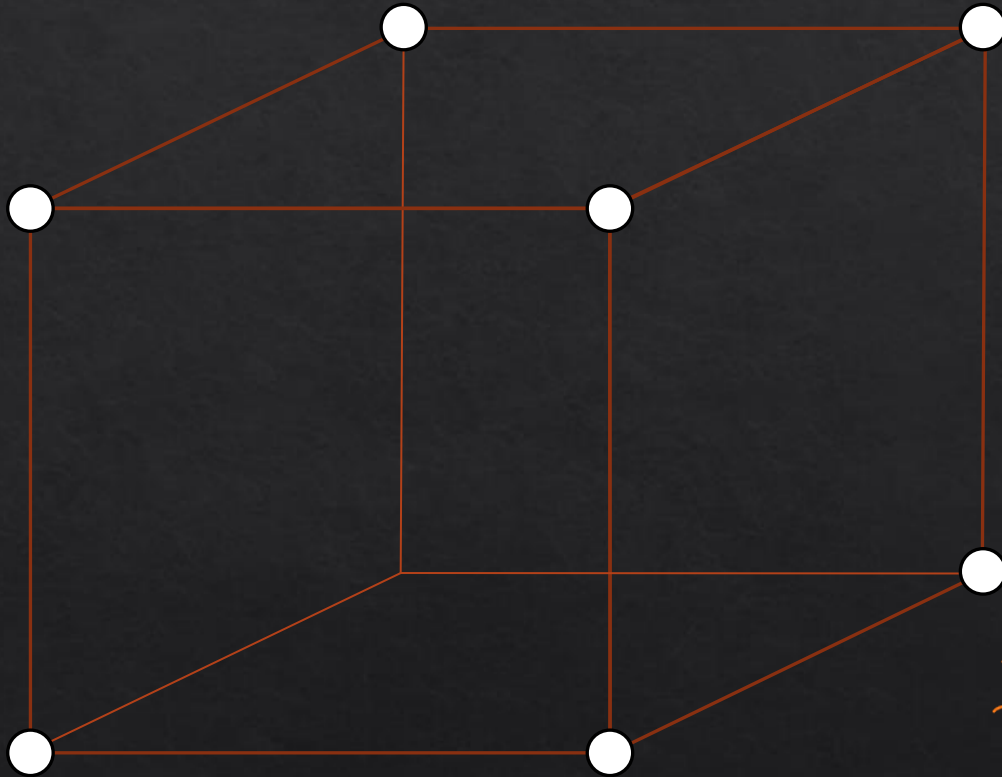
## ◇ 좌표계



오른손 좌표계

Open GL은 오른손 좌표계

# 질문



정육면체를 그리기 위해선  
몇 개의 정점이 필요할까?

정육면체를 그리기 위해선  
몇 개의 삼각형이 필요할까?

그냥 볼다면 정은 8개 이지않,

사실 12개의 삼각형을 그려야 하므로

36개의 정점이 필요하다

# Graphics Pipeline

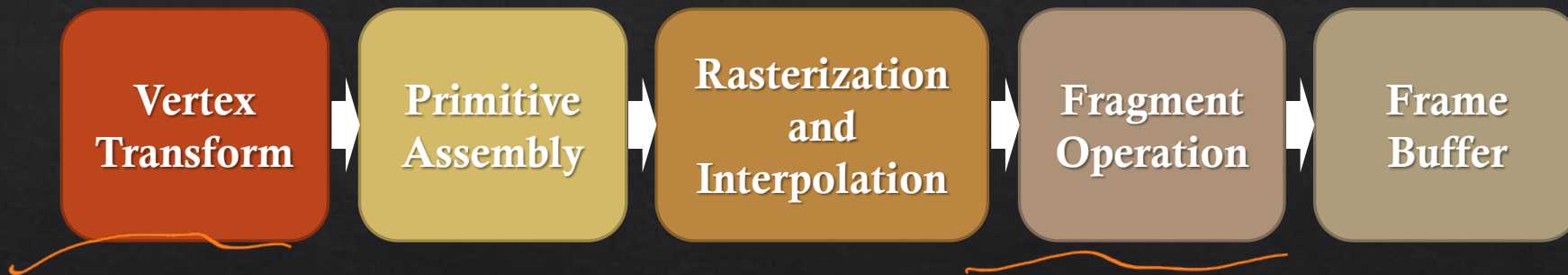
# Graphics Pipeline

- ◆ 그래픽스 파이프라인은 여러 단계에 걸쳐 진화를 해 왔음
- ◆ 가장 큰 변화는 고정 그래픽스 파이프라인에서 프로그램 가능한 그래픽스 파이프라인으로 변화한 것



# Graphics Pipeline

고정 그래픽스 파이프라인이란?  
(fixed-function pipeline)

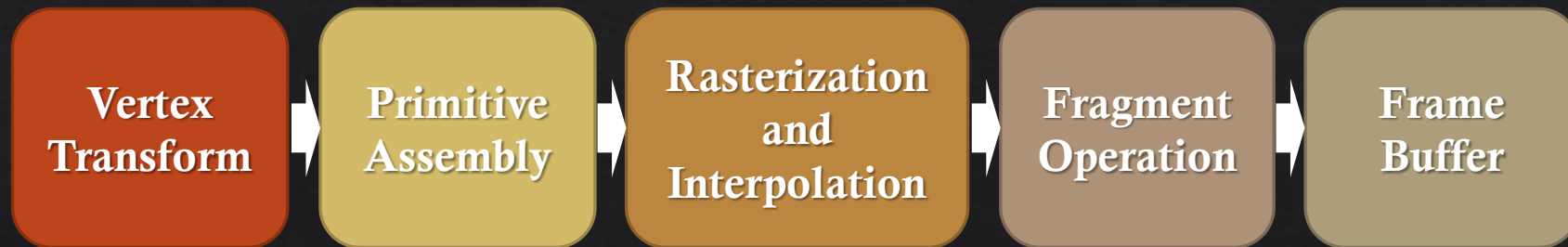


그래픽스 파이프라인의 각 단계는 미리 주어진 연산만 수행이 가능함  
필수적인 기능이 필요할 경우 하드웨어 및 드라이버 둘 다 업데이트가 필요  
↳ 시스템 소프트웨어



# 여기서 잠깐..

그래픽스 파이프라인의 각 단계가 수행하는 일에 대해 알고 있는지 확인 필요



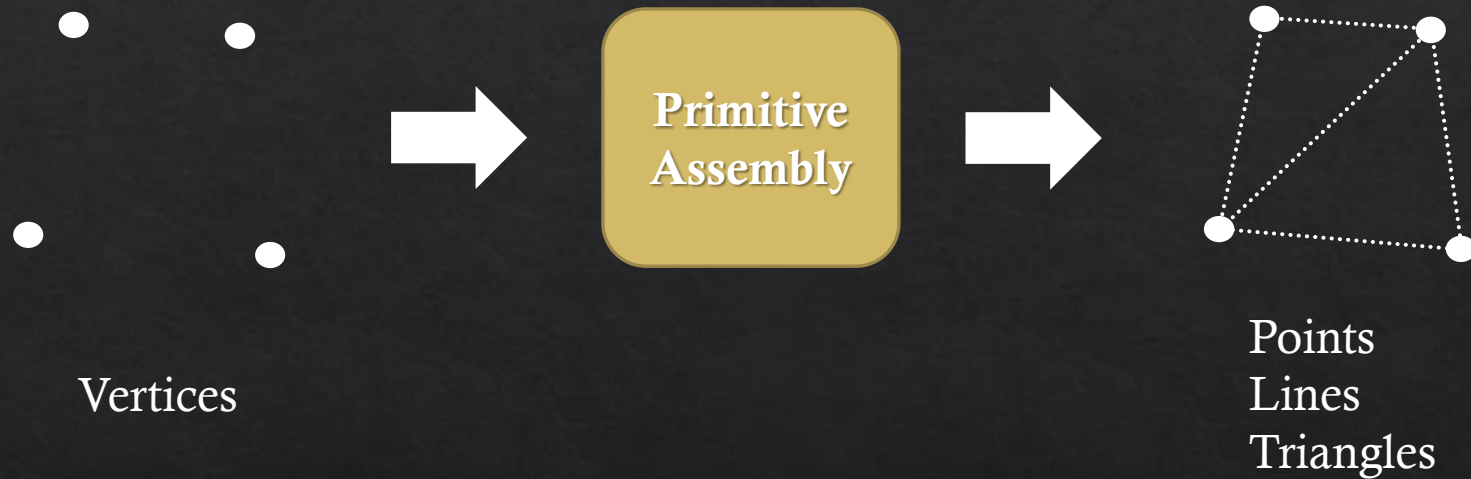
# Graphics Pipeline



점을 변환해 준다

입력값 출력값의 개수가 같음

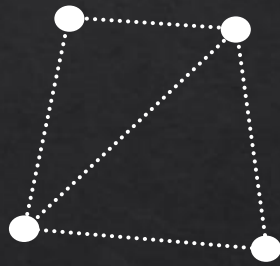
# Graphics Pipeline



점, 선, 면을 정한다

즉, 어떻게 보실 것인지 어떻게 서연리/를 정한다

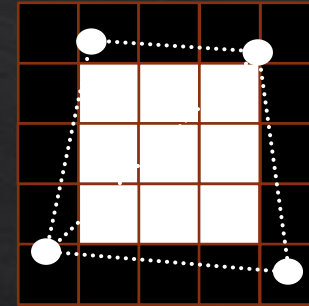
# Graphics Pipeline



Points  
Lines  
Triangles



Rasterization  
and  
Interpolation



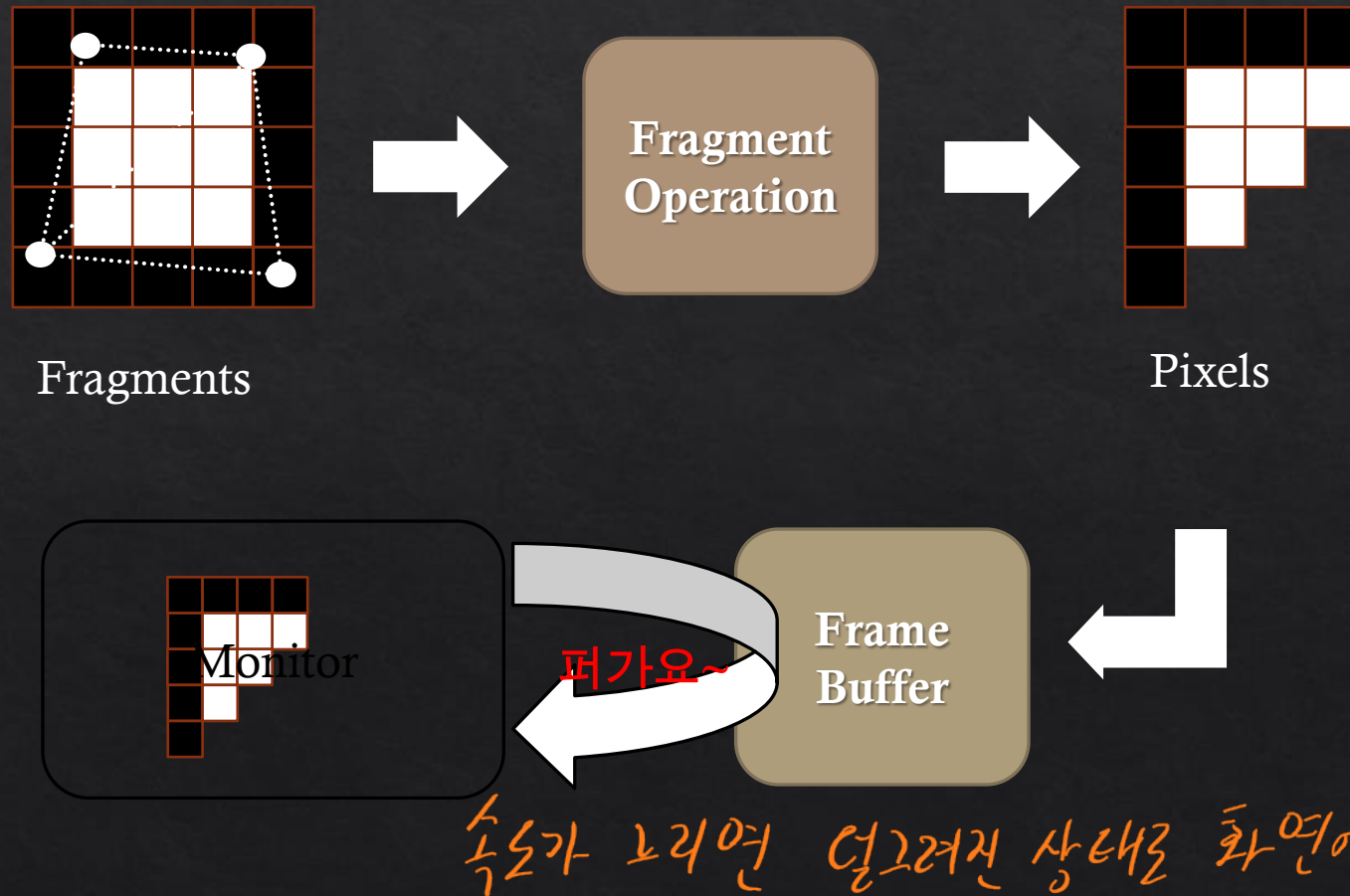
Fragments

이 단계를 지나면

입력된 점들이 픽셀들로 바뀌어 출력된다

즉 입력과 출력의 개수가 달라진다

# Graphics Pipeline



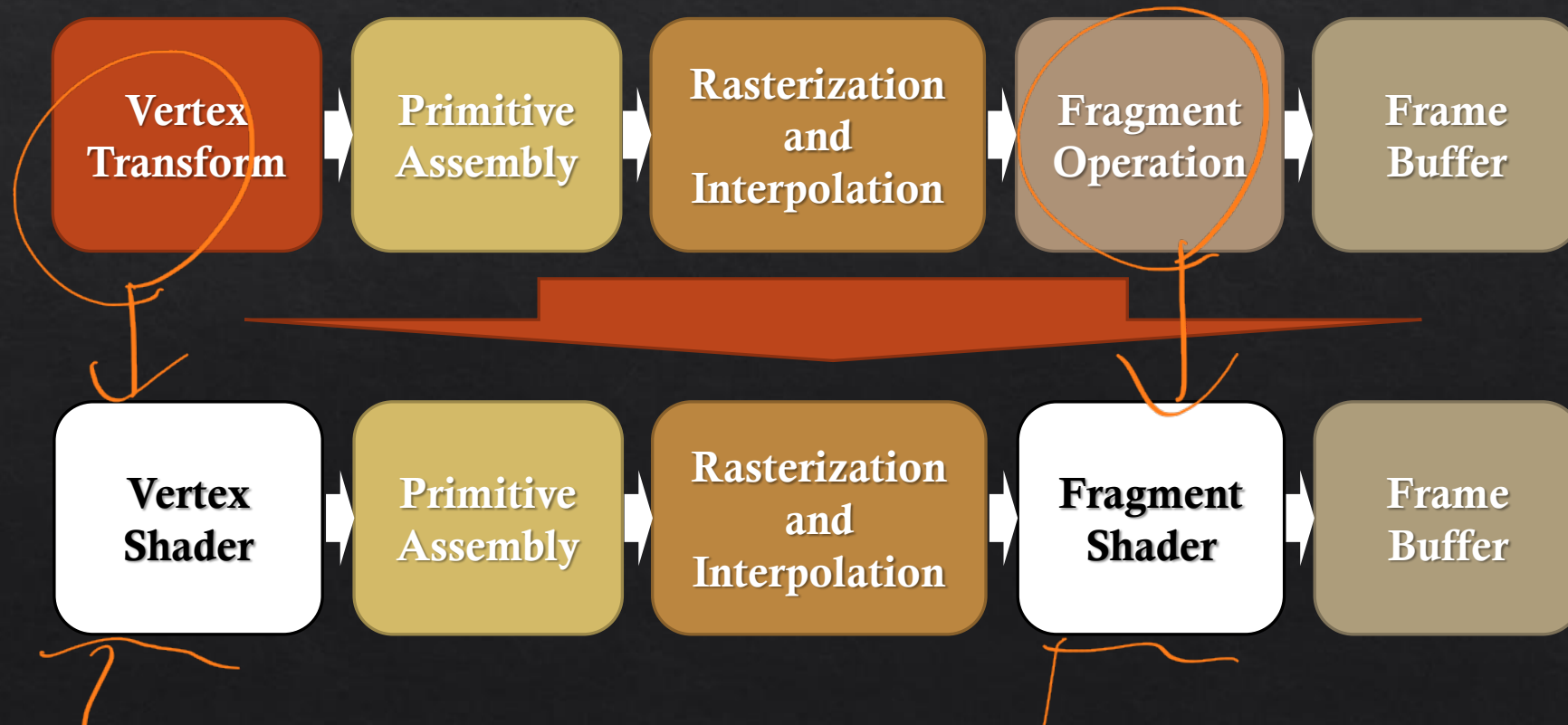


# PROGRAMMABLE Graphics Pipeline



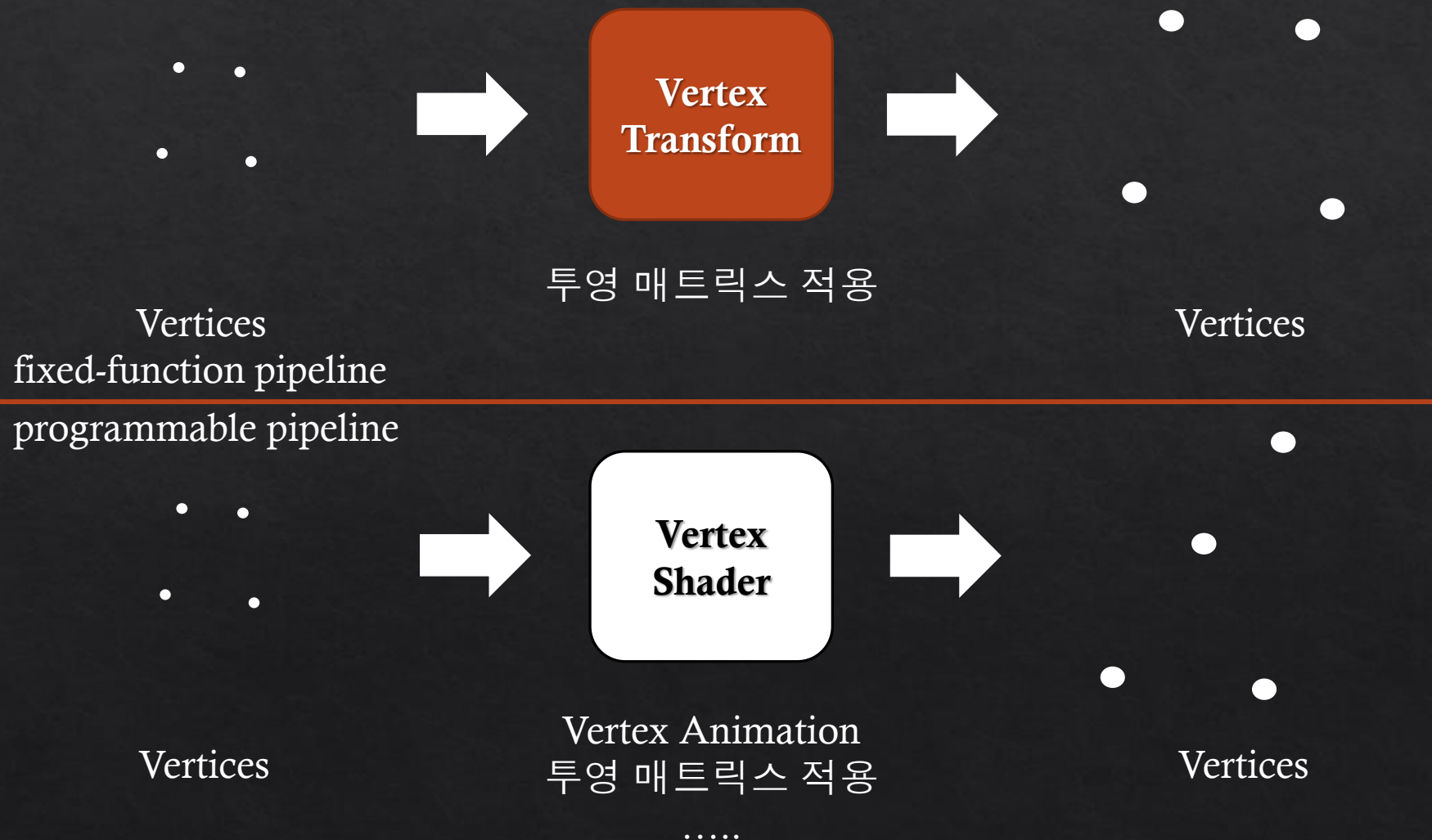
# Programmable Graphics Pipeline

프로그래밍 가능한 그래픽스 파이프라인이란?  
(programmable pipeline)

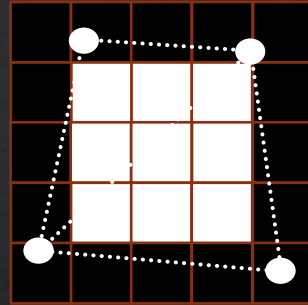


↳ 자극 바뀌는 것보다 프로그래밍 가능하도록 변경됨

# Programmable Graphics Pipeline



# Programmable Graphics Pipeline



Fragments



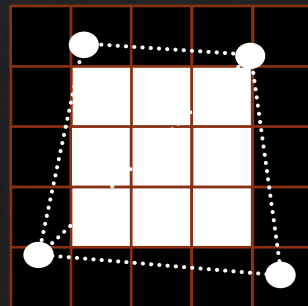
고정된 몇 가지  
광원효과



Pixels

fixed-function pipeline

programmable pipeline



Fragments



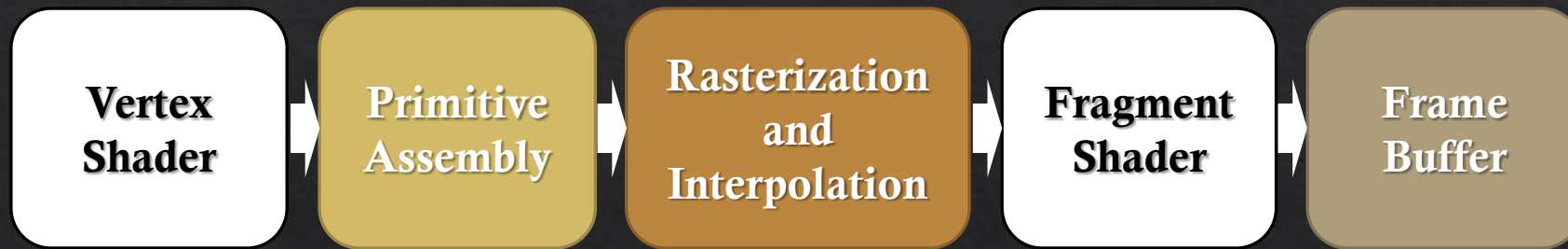
다양한 필터링  
NPR...



Pixels

이런 필터 이런 필터

# Programmable Graphics Pipeline

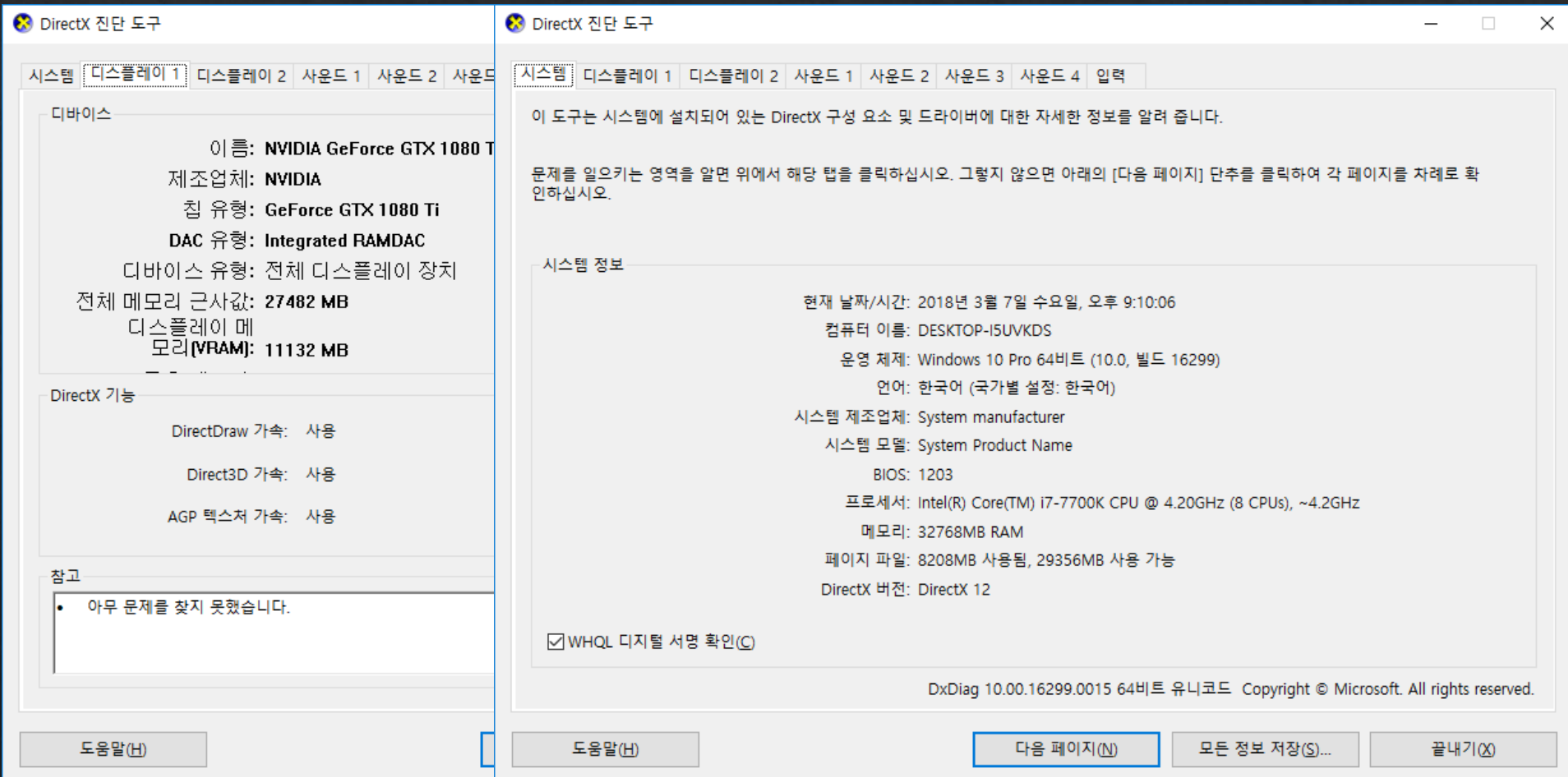


실습



# 실습

## ◆ 컴퓨터 사양 체크 (dxdiag 실행)





# 실습

## ◆ 컴퓨터 사양 체크

OpenGL Hardware Capability Viewer 1.1 - © 2011-2016 by Sascha Willems (www.saschawillems.de)

Refresh Upload Save xml Device Database Settings About Exit

OpenGL

NVIDIA Corporation GeForce GTX 1080 Ti/PCIe/SSE2 4.6.0 NVIDIA 388.13 (Windows 10)  
Device not yet present in database

Your device Database

Implementation Extensions (407) Compressed formats (51) Internal formats

Filter :

Key	Value
Implementation details	
OpenGL version	4.6.0 NVIDIA 388.13
Operating system	Windows 10
Renderer	GeForce GTX 1080 Ti/PCIe/SSE2
Shading language version	4.60 NVIDIA
Vendor	NVIDIA Corporation
OpenGL Version	2
GL_MAJOR_VERSION	4
GL_MINOR_VERSION	6
OpenGL 1.2	3
GL_MAX_3D_TEXTURE_SIZE	16384
GL_MAX_ELEMENTS_INDICES	1048576
GL_MAX_ELEMENTS_VERTICES	1048576
OpenGL 1.3	2
GL_MAX_CUBE_MAP_TEXTURE_SIZE	32768
GL_MAX_TEXTURE_UNITS	4
OpenGL 1.4	1
GL_MAX_TEXTURE_LOD_BIAS	15
OpenGL 2.0	9
GL_MAX_COMBINED_TEXTURE_IMAGE_UNITS	192
GL_MAX_DRAW_BUFFERS	8
GL_MAX_FRAGMENT_UNIFORM_COMPONENTS	4096

<http://opengl.gpuinfo.org/download.php>

# 실습

- ◆ 수업용 visual studio project 다운로드 후 컴파일 및 실행 확인
  - ◆ E-class 강의 자료에 올려져 있음