

3D콘텐츠 이론 및 활용

유길상

1주. OT

1차시 : 콘텐츠 개발 엔진의 이해

2차시 : 컴퓨터 언어의 이해

교과목 개요

학습 개요

- 게임엔진은 다양한 콘텐츠를 쉽게 빠르게 개발할 수 있는 개발도구로 스마트폰 기기, 웹 브라우저, 데스크톱, 스마트TV 등 기기의 구분 없이 원하는 콘텐츠를 쉽게 만들 수 있다. 이 강좌에서는 여러 전공분야에서 필요로 하는 3D콘텐츠를 누구나 쉽게 만들어 볼 수 있도록 기초 이론과 실습을 통해 가상월드를 만들고 활용할 수 있는 역량을 갖도록 한다. 이를 통해 간단한 앱 개발 뿐만 아니라 게임, 교육용 콘텐츠 개발, 미디어아트 제작, 건축(공간) 모델링, 예술작품 등에서 사용가능한 결과물을 창출하는 능력을 가질 수 있도록 한다.

평가방법

- 자율 출석 체크, 무감독 자율시험

과제	20	중간고사	30
출석	20	기말고사	30

연락처

- phd.yoo@gmail.com / ksyoo@korea.ac.kr
- 010-8995-7944 / 02-3290-1674
- 라이시움 306호

주차별 학습내용

1주	과목소개
2주	3D모델링을 위한 기초 이론
3주	유니티 조작 기초
4주	유니티 인터페이스
5주	C# 스크립트 코딩으로 객체조작하기
6주	C# 스크립트로 컴포넌트조작하기
7주	입력 처리 및 컴포넌트 활용
8주	중간고사

9주	오브젝트 충돌처리
10주	사용자 인터페이스 처리
11주	사운드 처리
12주	네비게이션 메쉬활용
13주	파티클 효과
14주	애니메이션을 위한 모션제작(상태전환 머신, 메카닉)
15주	앱, 플랫폼 패키지 만들기
16주	기말고사

1차시 학습목표

- 게임엔진에 대하여 이해한다.
- 게임엔진의 종류에 대하여 이해한다.
- 유니티 게임엔진의 특징에 대하여 이해한다.

1차시 학습내용

- 게임엔진을 이용하는 이유
- 게임엔진의 종류와 특징
- 유니티 게임엔진의 특징

생각해보기

1. 영화, 애니메이션, 게임, 공연/전시, 전자책, 만화 콘텐츠의 제작 방식의 차이점은 무엇이 있을까요?
2. 게임엔진을 사용하는 이유는 무엇일까요?
3. 다른 엔진들은 어떤 것이 있을까?

1. 게임엔진의 이해

왜 게임 엔진을 사용할까?

1) 자동차 엔진

- 특정 성능의 자동차를 빠르게 생산하기 위해 타 회사 엔진을 들여와 사용
예) 쌍용자동차 일부모델은 벤츠 엔진 탑재

2) 게임 엔진

- 특정 기능의 게임을 빠르게 생산하기 위해 타 회사 엔진을 들여와 사용
- 게임 개발용 각종 라이브러리/리소스 관리 툴
 - 적은 노력으로 고품질의 결과물 생산 가능
- 구성 요소
 - 렌더링엔진 : 3차원 공간을 2차원 화면에 그리는 역할
 - 물리엔진 : 3차원 공간 상의 물체들이 실세계의 물리법칙에 따라 움직이도록 처리 및 계산하는 역할

1. 게임엔진의 이해

영화와 게임 제작의 다른 점은?

1) 영화

- 고품질 영상을 얻기 위해 한 장의 이미지 생성에 몇 시간이 걸리더라도 고품질을 선택

2) 게임

- 초당 30장 이상의 이미지를 실시간으로 생성해야 함

2. 게임 엔진의 특징

1) Unreal Engine

- 에픽게임즈에서 FPS용으로 개발
- 한국게임의 대표작: 리니지2
- MMORPG, 레이싱 장르 개발에 활용
- PC, PS, Xbox, Mobile용 제작 가능



2. 게임 엔진의 특징

2) Cry Engine

- 독일의 게임 개발사 크라이텍에서 개발한 게임 엔진
- EA와 제휴해 <크라이시스> 출시와 함께 크라이 엔진 2를 발매
- 언리얼 엔진에 버금가는 고품질 그래픽을 구사
- 표현 가능한 가시거리가 넓고, 수면반사 효과, 지형 표현, 배경 처리 등이 뛰어나
- 다른 장르의 게임을 만들기 어려움
- 아이온, 문명 온라인

2. 게임 엔진의 특징

3) Unity

- 3D 게임을 비롯해 건축 시각화, 실시간 3D 애니메이션 같은 콘텐츠를 제작하기 위한 통합 저작 도구로 사용되고 있음
- 다른 게임 엔진보다 가격이 낮음, 직관적인 인터페이스
- 윈도우, 맥, 리눅스, 웹 브라우저와 같은 PC 기반 플랫폼부터 플레이스테이션3, 플레이스테이션4, WiiU, 엑스박스 360, 엑스박스 원과 같은 콘솔 기반 플랫폼, iOS, 안드로이드와 같은 모바일 플랫폼까지 굉장히 다양한 플랫폼을 지원.
- 에셋 스토어를 통해 소규모 개발사나 개인 개발자라도 캐릭터 모델, 소품, 자료, 질감, 풍경화 도구, 게임 제작 툴, 오디오 효과 등 게임 제작에 필요한 각종 자료를 유, 무료로 구할 수 있음

2. 게임 엔진의 특징

4) 주피터 Engine

- 모노리스 프로덕션 게임 엔진
- 당시 엔진중에서 저렴한 가격으로 좋은 품질의 게임 제작을 지원
- 낮은 PC 요구 사양으로 기동
- 서든어택
- 2005년부터 주피터 EX 엔진으로 그래픽 성능, '하복' 채용으로 성능 향상



2. 게임 엔진의 특징

5) Source Engine

- 밸브 코퍼레이션이 개발한 게임 엔진
- 셰이더에 기초를 둔 렌더러와 실사 같은 그래픽
- 페이셜처리 기술이 뛰어남
- 하복 물리 엔진을 기본 탑재해 사실적인 물리 표현이 가능하고, 자체적으로 네트워크 엔진을 지원해 온라인게임을 지원
- 또한, 게임 플랫폼 스팀에서 실행 가능한 모드 게임 '소스 엔진 모드'를 지원
- 소스 엔진을 활용한 대표적인 게임은 <하프라이프 2>, <레프트 4 데드>, <마비노기 영웅전> 등

3. 유니티의 특징

1) 개발 자원 확보의 용이함

- Asset Store
- 고품질 모델/텍스처 구매

2) 저렴한 가격

- 1500\$ (≒160만원, 무료 버전)
- Doom3: 60억원
- Source: 35억원
- Unreal: 30~60억원
- Jupiter: 25억원
- RenderWare: 25억원
- Quake3 Arena: 25억원
- Cry: 8억원

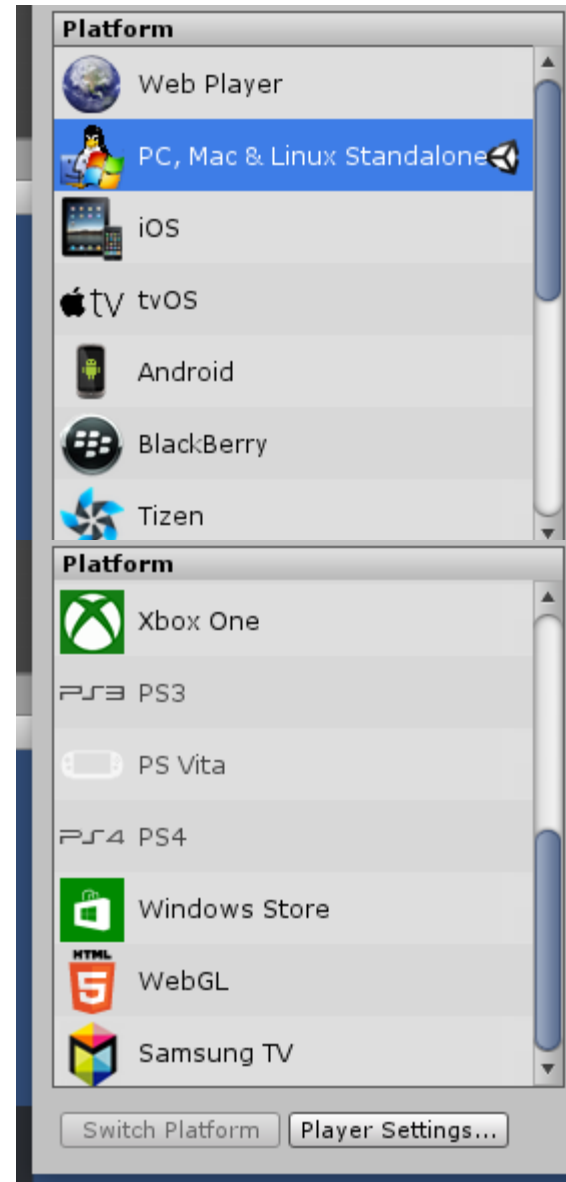
3) 개발 환경

- PC 또는 Mac

3. 유니티의 특징

4) 멀티플랫폼 지원

- 플래시 웹 플레이어
- iOS(아이폰, 아이패드)
- 안드로이드
- 닌텐도 Wii
- 소니 플레이스테이션
- 마이크로소프트 XBox360
- 맥 OS X
- 윈도우
- 리눅스
- 웹 브라우저



4. 유니티의 모토

1) 게임 개발의 민주화

- 가격적으로 부담이 적은 정책
- 제작 시간의 효율성

2) 3D콘텐츠 개발을 원하는 대부분의 사람들의 좌절

- 아이디어는 있는데 구현할 능력이 없음
- 상당한 수준의 물리학과 수학적 원리 이해 필요
 - 캐릭터를 어떻게 움직이지?
 - 아이템, 캐릭터, 배경, 객체 등은 어떻게 만들까?
 - 지형은 어떻게 만들어야 하나?
 - 포탄을 어떻게 발사하지?
 - 폭발을 어떻게 묘사하지?

5. 유니티 적용 분야

- 1) 게임
- 2) 앱
- 3) 건축
- 4) 디자인
- 5) 인테리어
- 6) 시뮬레이션
- 7) 의학
- 8) 군사 - 시뮬레이션 프로그램
- 9) 교육 - 이러닝 콘텐츠
- 10) 가상현실

6. 유니티 개발 언어

1) C#, 자바스크립트, Boo 스크립트

- 스크립트 형태로 작성
- 대부분 기본적인 문법만 사용

2) 스크립트 언어

- 소프트웨어의 동작을 대본(Script)처럼 순서대로 기술

3) 자바스크립트

- 인터프리터 언어
- 프로그램을 처음부터 문장별로 해석해 컴퓨터가 이해할 수 있는 형식으로 번역하면서 실행

6. 유니티 개발 환경

1) 통합 개발 환경 제공

- 게임의 개발과 실행 환경을 통합
- 개발 도중 언제라도 게임을 실행시켜 테스트 가능
- 테스트 도중에 각종 변수의 상태가 모니터링 될 뿐 아니라 변수의 값을 직접 바꿔 가며 게임의 상태를 관찰 가능

2) 다양한 2D 및 3D 모델링 툴 지원

- 대부분의 2D 및 3D 파일을 변환없이 사용 가능

	Meshes	Textures	Anims	Bones
Maya .mb & .ma ¹	✓	✓	✓	✓
3D Studio Max .max ¹	✓	✓	✓	✓
Cheetah 3D .jas ¹	✓	✓	✓	✓
Cinema 4D .c4d ^{1 3}	✓	✓	✓	✓
Blender .blend ¹	✓	✓	✓	✓
modo .lxo	✓	✓	✓	
Autodesk FBX	✓	✓	✓	✓
COLLADA	✓	✓	✓	✓
Carrara ¹	✓	✓	✓	✓
Lightwave ¹	✓	✓	✓	✓
XSI 5.x ¹	✓	✓	✓	✓
SketchUp Pro ¹	✓	✓		
Wings 3D ¹	✓	✓		
3D Studio .3ds	✓			
Wavefront .obj	✓			
Drawing Interchange Files .dxf	✓			

1

게임엔진이란 컴퓨터나 비디오 게임 같은 실시간 그래픽 표시기능을 갖춘 상호작용 응용 프로그램을 구현 하는 핵심 소프트웨어이다.

2

게임엔진은 게임개발에 바탕이 되는 기술을 제공하여 게임개발을 과정을 단축시켜 준다.

3

유니티 게임엔진은 개발된 게임을 다양한 플랫폼에서 실행 할 수 있도록 제공한다.

4

스크립터는 소프트웨어의 동작을 간편히 동작시킬 수 있는 언어이다.

한 학기 교과목 구성

1

유니티 다루기

2

C#언어

3

콘텐츠 제작 기초

4

2D, 3D 콘텐츠 제작실습

유니티 기초

용어, 인터페이스,
조작, 기초실습

유니티 프로그래밍
C# 스크립트

- 모델과 애니메이션
이용한 게임제작
- 메카닉 활용
- 기초 콘텐츠제작
- 3D콘텐츠 제작
- 앱개발 응용

- Unity 3.x Game Development Essentials, Will Goldstone, PACKT, 2011
- 유니티코리아 <http://unity3d.com/kr/>
- 언리얼엔진 코리아 <https://www.unrealengine.com/ko/>
- 크라이엔진 <http://cryengine.com/>

2차시

컴퓨터 언어의 이해

본 수업에 사용된 일부 자료 및 영상물은 강의 내용을 보충하기 위한 교육적 목적으로 활용하였습니다. 자료 및 영상물의 불법적 이용, 무단 전재·배포는 법적으로 금지되어 있으니, 학생 여러분께서는 학습 외 용도의 사용을 삼가 주시길 바라며, 자료 및 영상물 원본 전체를 사용하기 위해서는 별도로 정품을 구매하여 사용하시기를 바랍니다.

본 교안에는 네이버에서 제공한 나눔글꼴이 적용되어 있습니다.

1. 언어란 무엇인가?

- 1) 의사소통을 위한 약속
- 2) 다양한 언어

예) 한국인과 아마존 야노마미부족 사람과 의사소통하기

- 브라질 언어와 한국어를 구사하는 통역관 필요
- 브라질 언어(포르투갈어)와 부족언어가 가능한 통역관 필요

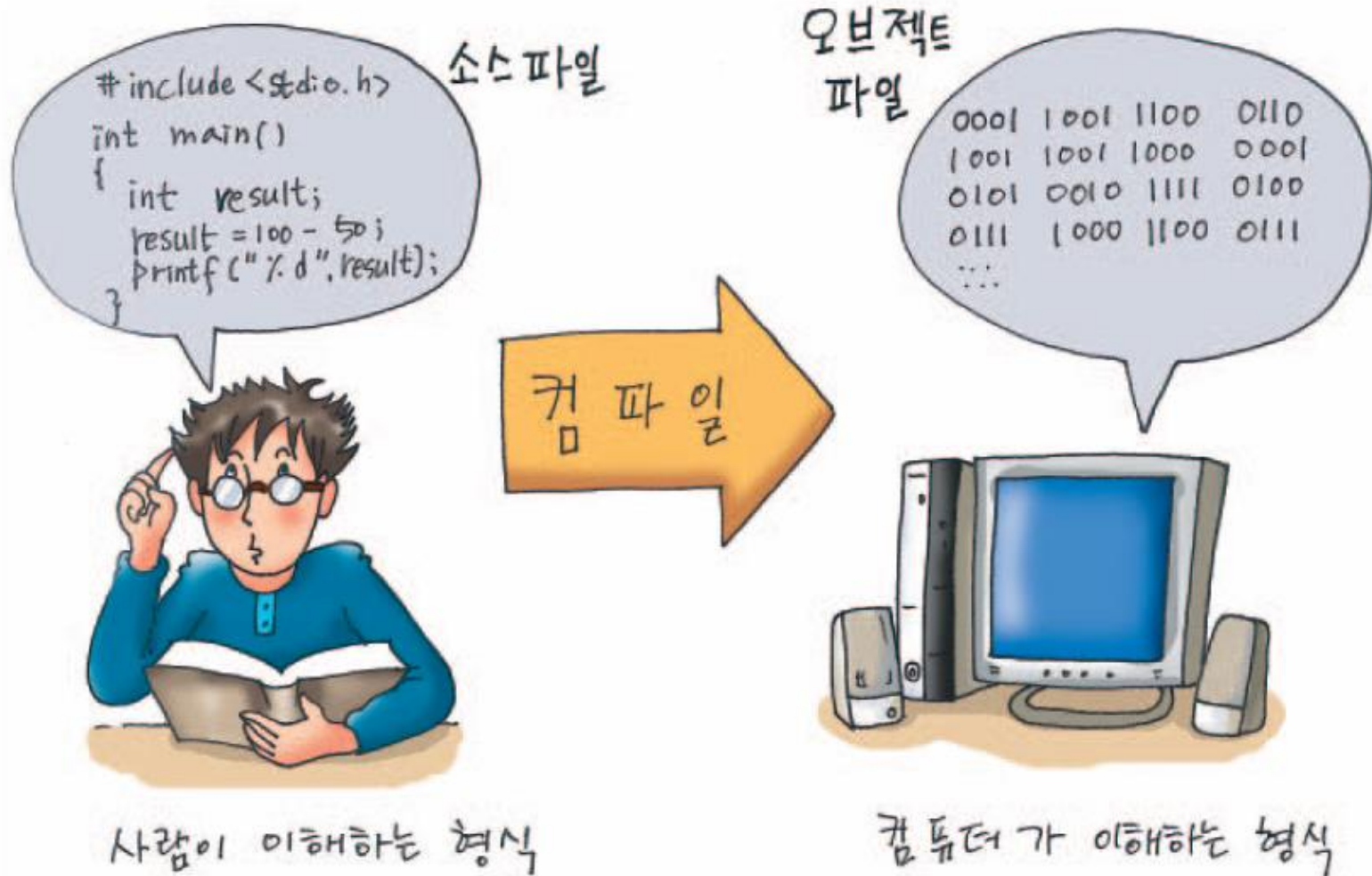


MBC, 2010 아마존의 눈물

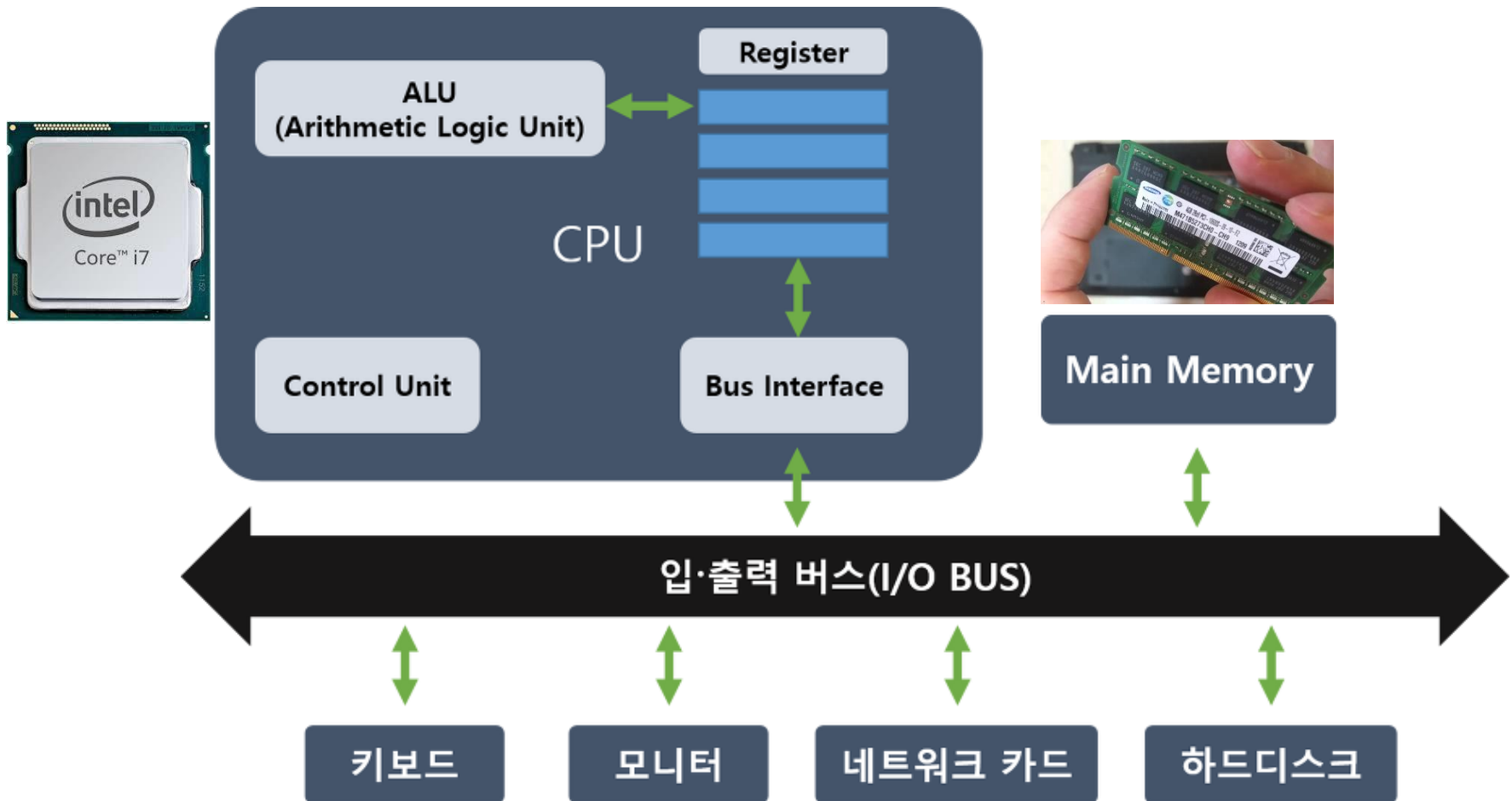
2. 컴퓨터가 이해하는 언어는?



2. 컴퓨터가 이해하는 언어는?



3. 컴퓨터가 계산하는 방식



```
studyScript.cs
```

```
void Start() {  
    int hako1;  
    hako1 = 12345;  
    Debug.Log(hako1);  
}
```

3. 컴퓨터가 계산하는 방식

1) 메모지(메모리)를 활용하여 계산

=, 오늘쪽 값을 왼쪽 메모지에 기록하라

```
memo1 = 10
```

```
memo2 = 20
```

```
memo3 = memo1 + memo2
```

```
Print(memo3)
```

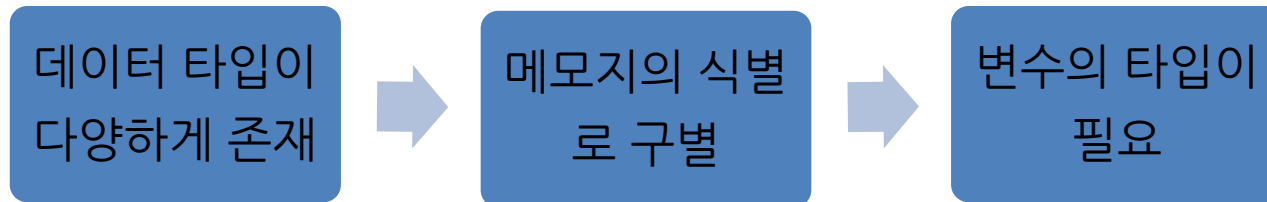
3. 컴퓨터가 계산하는 방식

2) 메모지의 적힌 뜻은?



3. 컴퓨터가 계산하는 방식

2) 메모지의 적힌 뜻은?



- 자료형 : 메모지(변수)의 형태
- ex) int - 정수형, float - 실수형, string - 문자열, bool- 참/거짓 판정(깃발)



- 컴퓨터는 2진수를 사용하기 때문에 사람이 사용하는 10진수로 표현하는 방식과 다르다.

3. 컴퓨터가 계산하는 방식

3) 변수(메모지)의 기능

- 변수의 내용은 원하는 대로 바꿀 수 있다.

```

int memo1;
memo1 = 1;
print(memo1);
memo1 = 2;
print(memo1);
    
```

- 위 코드를 실행했을 때 출력되는 값은?

3. 컴퓨터가 계산하는 방식

4) 변수(메모지)의 값을 비교하는 기능

- if ~ else

```
int memo1 = 1;
```

```
if(memo1 == 1) {  
    print("1입니다.");  
} else {  
    print("1이 아니야");  
}
```

3. 컴퓨터가 계산하는 방식

5) 중복되는 계산을 반복하는 기능

- for문 – 같은 처리를 반복할 때 사용

- 기본형 `for(변수의 초기값; 실행조건; 변화값) {`
 실행될 처리
 `}`

- 예제

```
for(int i = 0; i<10; i++) {  
    print(i);  
}
```

4. 통역관(컴파일러)은 모든 내용을 통역할 수 있을까?

- 유니티 C# 프로그램의 기본 형태

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class NewBehaviourScript : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {

    }

}
    
```

1

컴퓨터 언어와 스크립트

2

데이터의 다양한 종류, 변수의 종류, 사용을 위한 선언

3

컴파일러의 역할과 선언(using)

4

대입문, 조건문