**수업 1.3**

-유니티는 scene별로 구성되어 있다.

-게임뷰도 scene뷰랑 비슷한데 다른 점은 scene뷰에 있는 카메라에 있는 부분만 보여준다.

-hierarchy는 모든 구성요소를 담는다.

-inspector뷰는 구성요소의 자세한 정보를 나타냄.

-project에 모든 assets을 보여줌.

-파일 생성 및 삭제는 finder에서 말고 unity editor를 통해 하기

-console창 = script할 때 메시지, 경고, 오류 띄워주는 창

단축키:  
command + 1 = scene view

command + 2 = game view

command + 3 = inspector view

command + 4 = hierarchy view

command + 5 = project view

**수업 1.4**

게임 오브젝트

-scene은 게임 오브젝트로 구성되어 있다. (hierarchy에 있는 sample scene에서 main camera가 오브젝트이다.)

-다른 오브젝트 생성은 hierarchy 뷰에서 한다.

-command-shift-n = 게임오브젝트 생성

-move tool은 축으로 움직일 수 있다.

-scale tool은 축 방향으로 크기를 조절할 수 있다.

-rect tool 네모 모형으로 크기 조절

-transform tool은 모든게 다 합쳐진 tool

-inspector로 조절하는 것을 많이 사용(position, rotation, scale(x,y,z 축))

- position에서 z값은 깊이이다.

-2d에서는 x와y축으로 회전하지는 않는다. z값은 우리가 보는 수직인 방향으로 회전한다.

-scale은 x와 y축.

**수업 1.5**

-컴포넌트(component)

-게임 오브젝트는 컴포넌트로 구성되어 있다. (inspector 뷰에 있는 transform, sprite renderer가 컴포넌트이다.

-sprite renderer는 그림을 그려주는 component

-transform는 위치, 회전, 크기를 조절할 수 있는 component

-rigidbody 2d component는 (고체) 중력을 받아서 밑으로 떨어지게 하는 component

-component는 게임 오브젝트에 기능을 갖게 해준다.

-공을 떨어뜨려서 바닥에 충돌하게 하는 유니티-

-볼이라는 게임 오브젝트는 중력을 받는 기능을 갖고 collider라는 컴포넌트를 통해 충돌하는 기능

-ground는 collider라는 컴포넌트를 통해 다른 오브젝트와 충돌하도록 기능을 받았다.

-내가 원하는 기능이 없다면 내가 만들어야 한다. (이 과정을 scripting, programming이라고 한다.) 내가 따로 만든 기능.

-다음 수업은 script를 어떻게 생성, 어떻게 생겼고. 어떻게 게임 오브젝트에 작용하는지를 배운다.

**수업 1.6**

스크립트

생성방법:

1: project 뷰에 있는 assets를 클릭하고 오른쪽 클릭하고 create 누르고 c# script 클릭해서 생성

2: menu bar에 있는 assets을 누르고 create 그리고 c# script 클릭해서 생성한다.

3: Hierarchy 뷰에 있는 게임 오브젝트 클릭하고 add component 누르고 생성한다. 그러면 component에도 생성이 된다.(inspector 뷰)

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class test : MonoBehaviour

{

// Start is called before the first frame update

void Start()

{

}

// Update is called once per frame

void Update()

{

}

}

-using분들은 namespace를 불러오는 것

ex.) using UnityEngine은 내가 UnityEngine을 불러와 사용하겠다 라는 뜻

-public 문부터 마지막 문은 가장 중요한 class이다.

-test가 class이다.

-Monobehaviour는 스크립트랑 유니티 게임 오브젝트를 연결 시켜준다. 만약 Monobehaviour가 없으면 에러가 발생한다.

-void start와 update는 메소드(method)이다.

start는 딱 한번만 호출, 실행된다.

-update는 매 프레임마다 실행된다.

start와 update에 우리가 원하는 기능을 만든다.

-Debug.Log는 콘솔 창에 괄호 안에 있는 메시지를 출력해준다.

-update는 주의해서 사용해야 한다. update는 매 프레임마다 출력하기 때문에 중간에 튕길수도 있고 멈추면 저장하지 못한채 꺼야한다.

-다음 수업은 c# 기초 문법을 배운다.

수업 1.7

C# Extension 다운로드