

CIEN Unity 3D

중급반 (화요일 오후 7시)

고주형



목차

1. 유니티의 UI란?
 2. UI를 위한 개념들
 - 화면 해상도
 - Canvas
 - Anchor
 - Render Mode
 3. 텍스트
 4. 버튼
 5. 이미지
 - Fill을 이용한 다양한 활용 방법
 6. World 게임오브젝트의 마우스 감지
 7. 씬 이동
- 실습. 점수창 만들기



1. 유니티의 UI

UI? User Interface

유니티의 UI

- 점수, 환경설정 버튼, 중지 버튼, ... 은 어떻게 표현할까?
 - 화면에 가려지면 안 된다
 - 항상 보여야 한다
 - 월드상의 오브젝트와 상호작용하지 않는다(일반적으로)
- 2D 평면에 존재한다
- RectTransform 컴포넌트를 사용한다
- 카메라에 보이게 할지 설정할 수 있다

느낌이 다른 게임오브젝트

World

게임오브젝트

- 플레이어, 적
- 미사일
- 지형
- 2D 또는 3D

UI

게임오브젝트

- 정지/설정 버튼
- 점수 텍스트
- 입력창
- 이미지
- 2D



2. UI를 위한 개념들

먼저 해상도를 고정하자!

Unity 에디터의 화면 해상도

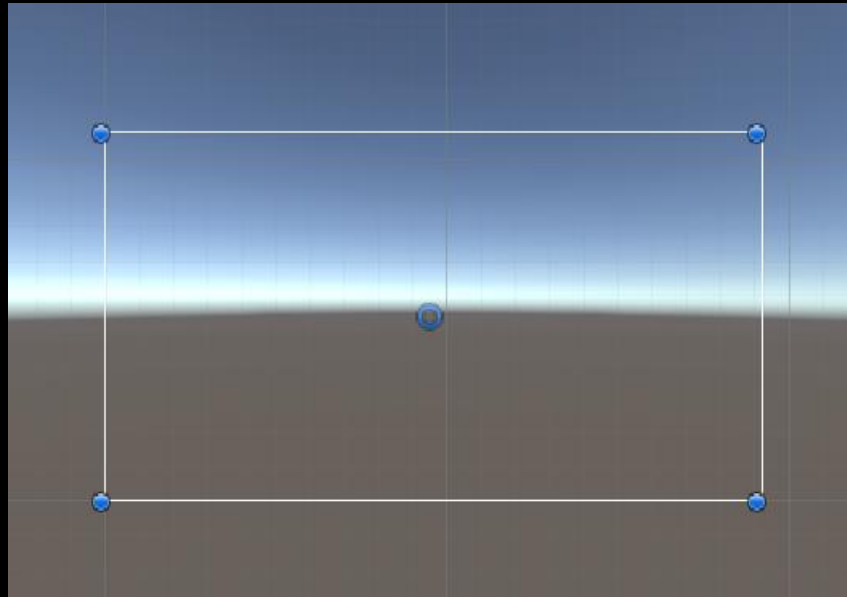
- Free Aspect
 - 내 창 크기에 맞춰서 해상도가 바뀐다
- 게임을 만들 때는 목표 해상도를 설정하고 만드는 것이 좋다.
 - Quiz. Free Aspect으로 한다면 어떤 일이 발생할 수 있을까?
 - Anchor를 잘 설정하면 ㄱ
- 1920 x 1080으로 설정해주자.
 - 또는 1280 x 720 같은 자주 쓰는 해상도로...



Canvas



- Canvas = UI 게임오브젝트를 배치하는 곳
 - 하얀선 - Game View의 크기
 - 모든 UI는 Canvas의 자식이다

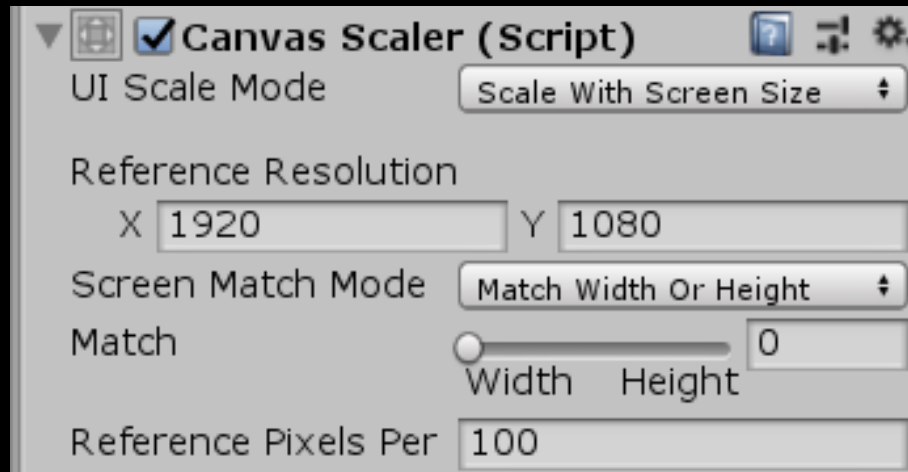


Canvas Scalar

- Constant Pixel Size
 - 실제 화면 크기와 무관하게 일정한 Pixel 크기 유지
- Scale With Screen Size
 - 실제 화면 크기에 맞춰서 UI 크기를 조정
- Constant Physical Size
 - 해상도와 실제 화면 크기와 무관하게 물리적인 크기 유지

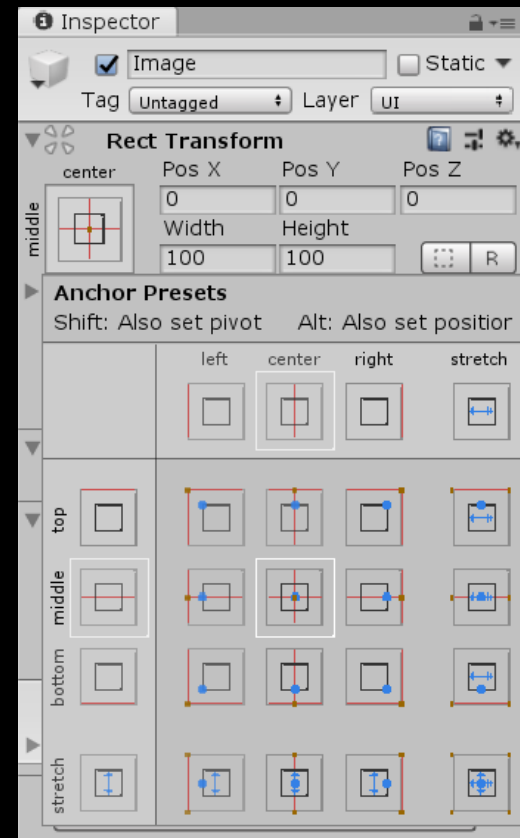
Scale With Screen Size 상세히

- Reference Resolution을 설정
 - 실제로 많이 사용하는 해상도로 하는 것이 유리하다
 - ex) 1920x1080
- 실제 화면이 Reference Resolution과 작거나 크면 Match만큼 UI를 키우거나 줄인다



Anchor

- Anchor = UI가 배치되는 기준!
 - (ex) Anchor에서 오른쪽으로 n만큼 왼쪽으로 m만큼!
- UI의 해상도 바뀌면 어디에 있을까?
 - Anchor를 기준으로 해서 위치가 정해진다.
- Quiz. Anchor가 이상하면?



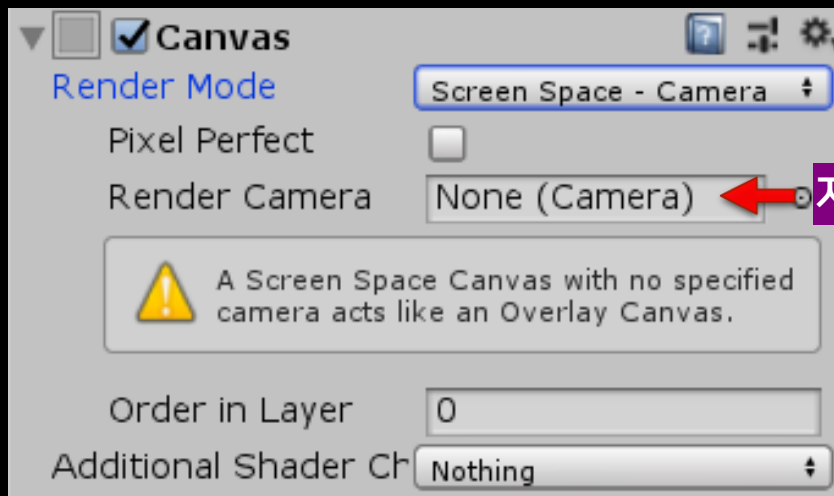
Render Mode: Overlay

- 월드와 무관하게 항상 앞에 보이는 UI
- 기본값
- ex) 설정 버튼, 점수, ...
- Camera와 분리된 Screen Space
= 카메라를 움직여도 UI는 움직이지 않는다



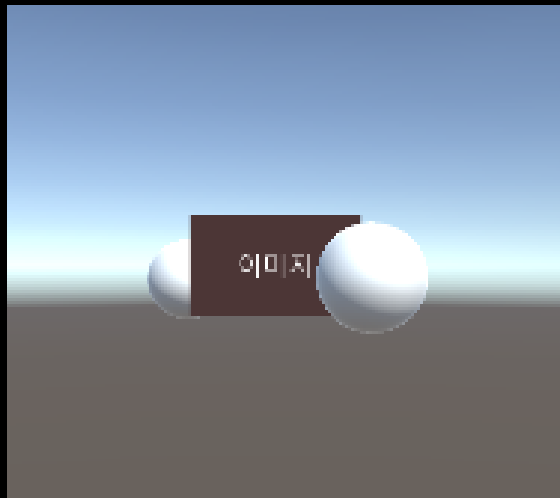
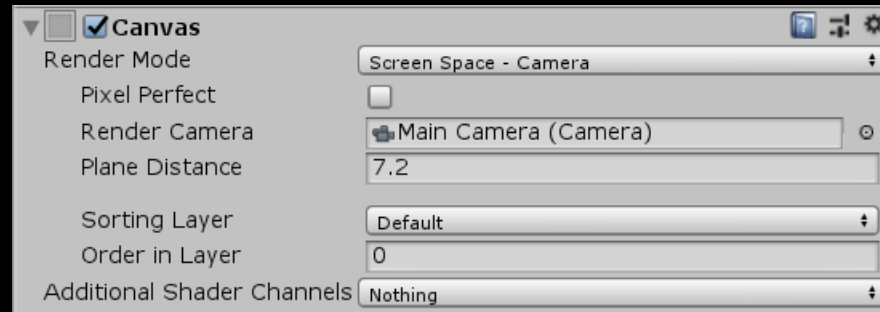
Render Mode: Camera

- World 게임오브젝트에 의해서 가려질 수 있는 UI
- 카메라와 World 앞/뒤/사이에 존재할 수 있다
- Render해줄 Camera를 지정해줘야 한다
 - 없으면 Overlay랑 동일하게 작동
- Camera와 분리된 Screen Space
= 카메라를 움직여도 UI는 움직이지 않는다



지정해야 함!

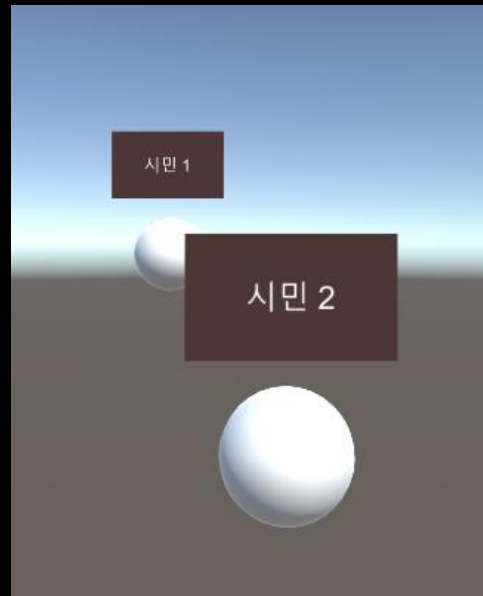
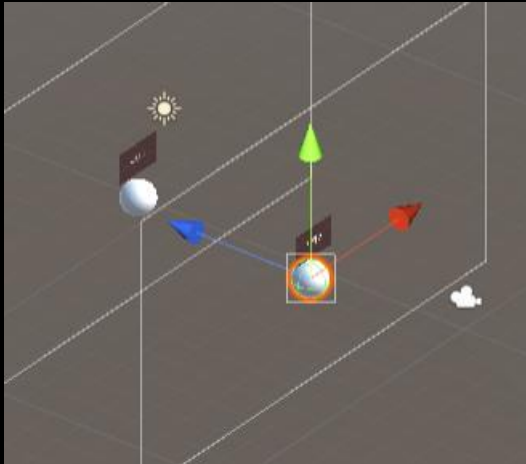
Render Mode: Camera

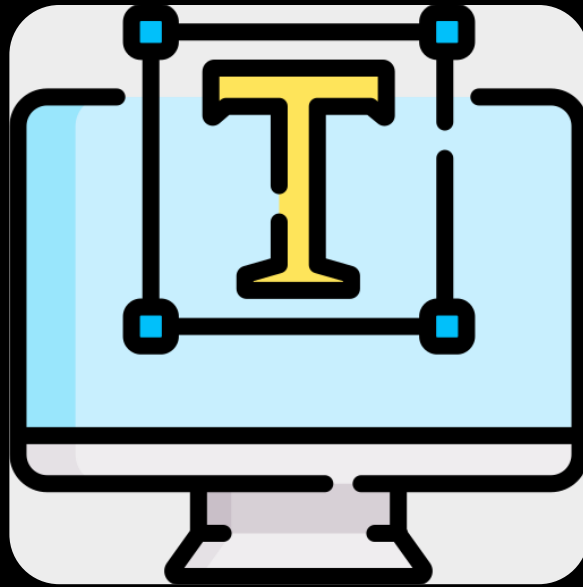


Render Mode: World Space

- 월드 오브젝트에 부착할 수 있는 UI
- AOS 게임 - 유저명, HP
- Screen Space가 아님

= Canvas를 World상 게임오브젝트에 상속 시킬 수 있다



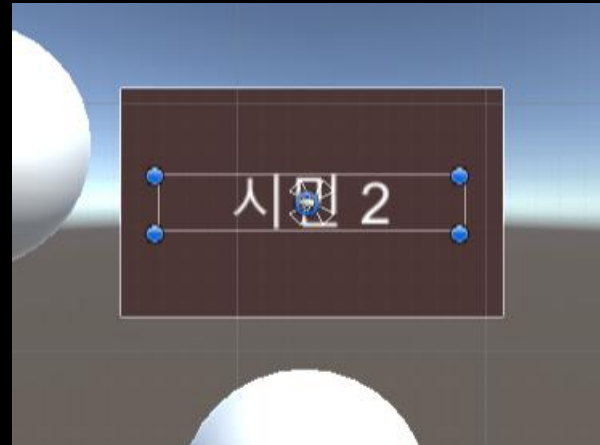
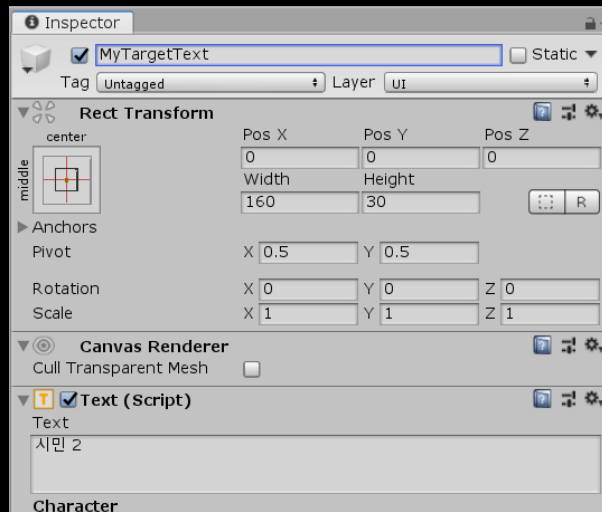


3. 텍스트

글자들

텍스트

- 텍스트를 코드로 수정해보기



- "시민 2"를 "시민 3"으로 바꾸고 싶다면?
 - UI의 Class들은 UnityEngine.UI에 있다!
- using UnityEngine.UI;**

텍스트 바꾸는 코드

- 텍스트의 text 속성을 바꾸면 된다.

```
using UnityEngine;
using UnityEngine.UI;

public class ChangeTextExample : MonoBehaviour
{
    private Text targetText;
    // Start is called before the first frame update
    void Start()
    {
        targetText = GameObject.Find("MyTargetText").GetComponent<Text>();

        if(targetText != null)
            targetText.text = "시민 3";
    }
}
```



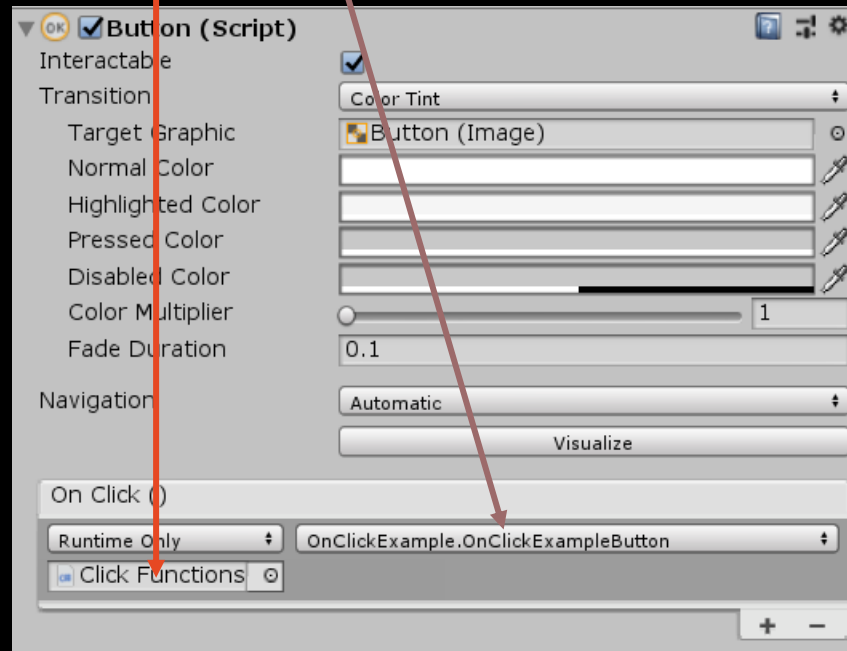
4. 버튼

버튼 클릭?

버튼

버튼을 눌렀을 때 함수를 호출해보자

1. 함수가 있는 게임오브젝트를 D&D!
2. OnClick 했을 때 실행할 함수 지정!

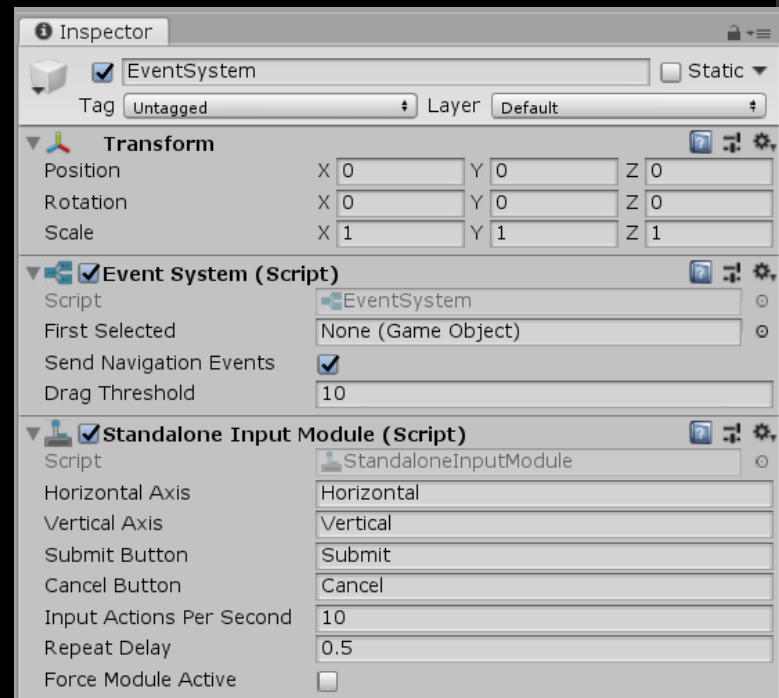


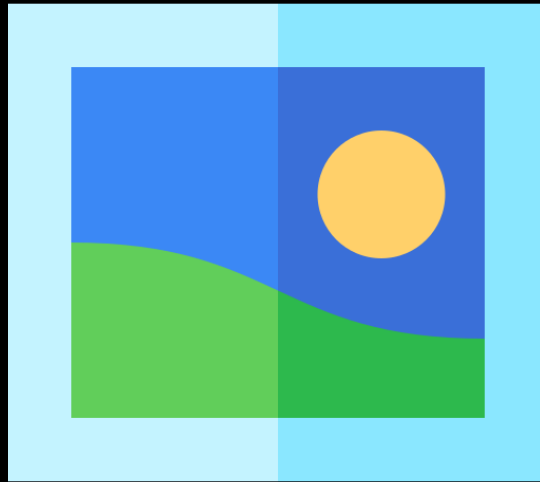
Quiz.

Drag, Click, ... 등의 Event는 누가 감지할까?

Event System이다!

- 한 씬에 **하나**만
- 없으면 이벤트를 감지하지 못한다
- UI를 생성하면 자동으로 생긴다





5. 이미지

Image 삽입, 쿨 타임, HP 바

이미지를 넣어보자

- Source가 이미지 파일
- 인터넷에서 다운 받아서 바꿔보자
 - <https://unsplash.com/s/photos/cc0> <- 저작권 없는 이미지
 - 3D 프로젝트면 바로 사용하지 못하지만 (Sprite로 바꿔주자)
 - 2D 프로젝트면 바로 사용가능한 형태로 이미지를 변환해 줌

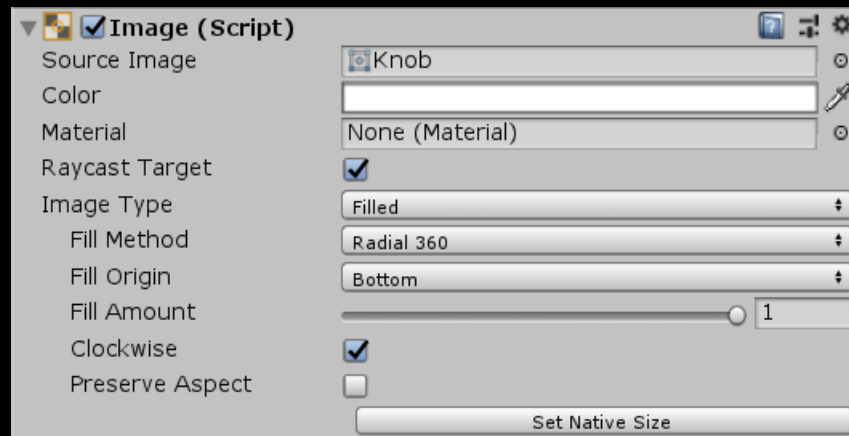
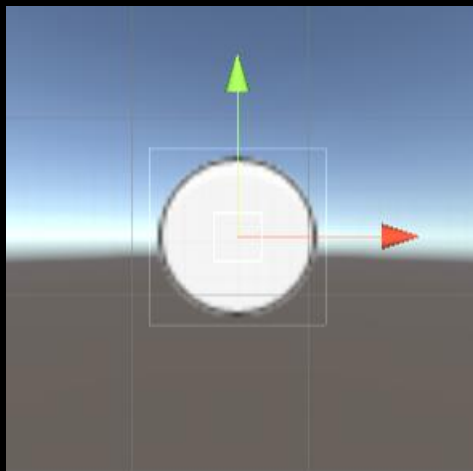
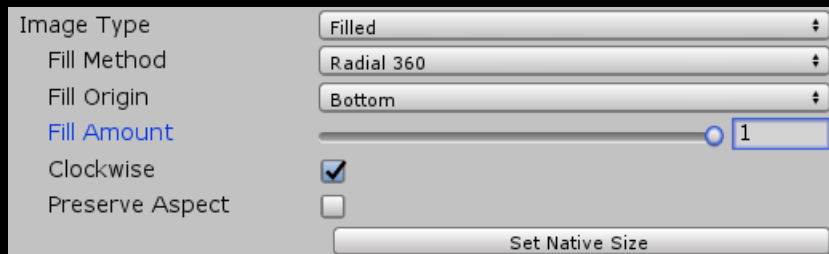


Image Type – Filled

- 쿨 타임, HP/MP 바, ... 표현하기 쉬움



Fill을 이용해보자

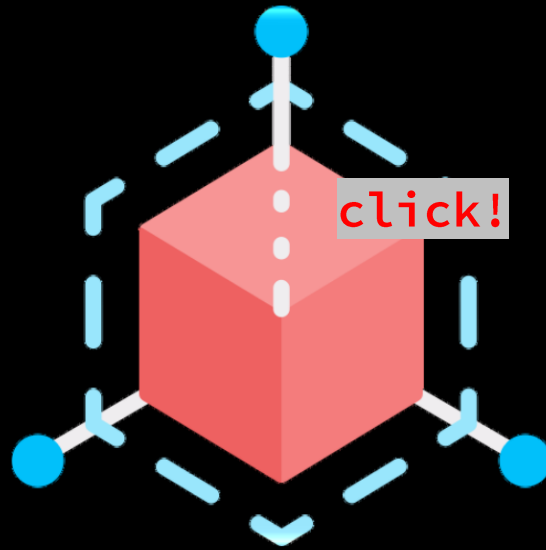
```
using System.Collections;~
using UnityEngine;~
using UnityEngine.UI;~
~
public class CoolTime : MonoBehaviour~
{~
    private Image coolTimeImage;~
    // Start is called before the first frame update~
    void Start()~
    {~
        coolTimeImage = GameObject.Find("CoolTime").GetComponent<Image>();~
        StartCoroutine(ShowCoolTime(coolTimeImage, 3f));~
    }~
~
    IEnumerator ShowCoolTime(Image img, float coolTime)~
    {~
        float delta = 1f / coolTime;~
        img.fillAmount = 1f;~
        while (img.fillAmount >= 0) {~
            yield return new WaitForSeconds(1);~
            img.fillAmount -= delta;~
        }~
    }~
}
```

이미지의 OnClick Event

- Button 컴포넌트를 달고 버튼과 같은 방법으로 사용

실습해보기

- 버튼을 클릭할 때마다 Text의 값을 1만큼 증가시켜보자

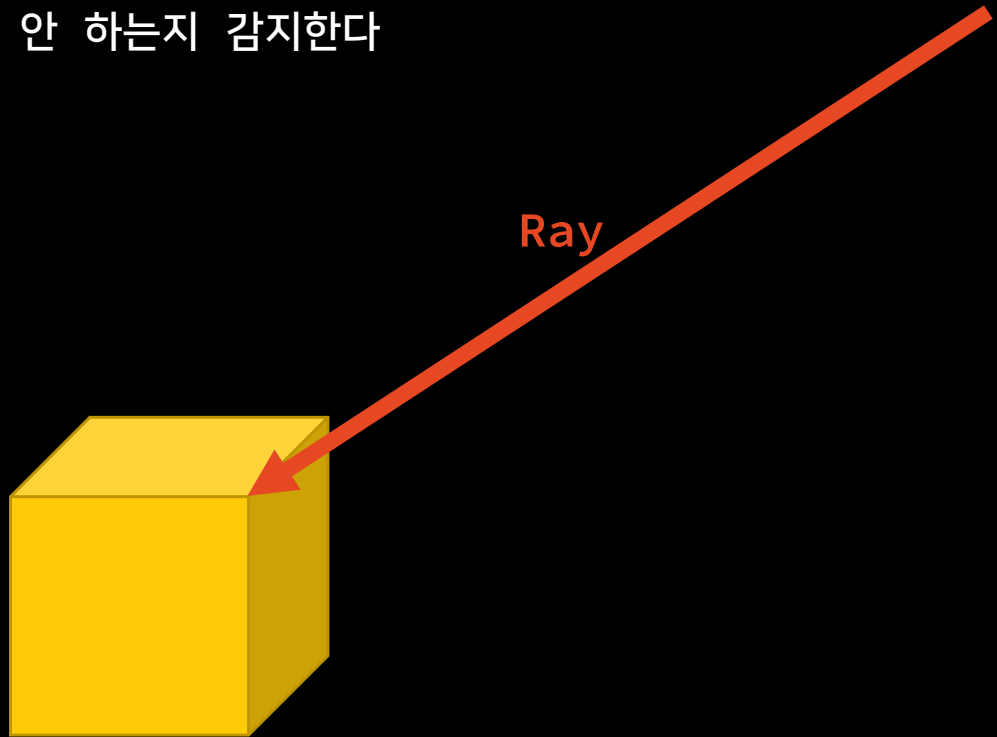


6. 월드 게임오브젝트의 이벤트

World의 게임오브젝트를 클릭하면?

World GameObject의 마우스 이벤트 감지

- Collider가 필요하다
 - Ray를 쏘고 충돌하는지 안 하는지 감지한다
- 특별한 이벤트 함수들
 - OnMouseDown
 - OnMouseUp
 - OnMouseDown



큐브 클릭을 감지해보자

```
using UnityEngine;~  
~  
public class OnCubeClick : MonoBehaviour~  
{~  
    private void OnMouseDown()~  
    {~  
        Debug.Log("Cube Click");~  
    }~  
    private void OnMouseDrag()~  
    {~  
        Debug.Log("Cube Drag");~  
    }~  
~  
    private void OnMouseExit()~  
    {~  
        Debug.Log("Cube Exit");~  
    }~  
}~
```

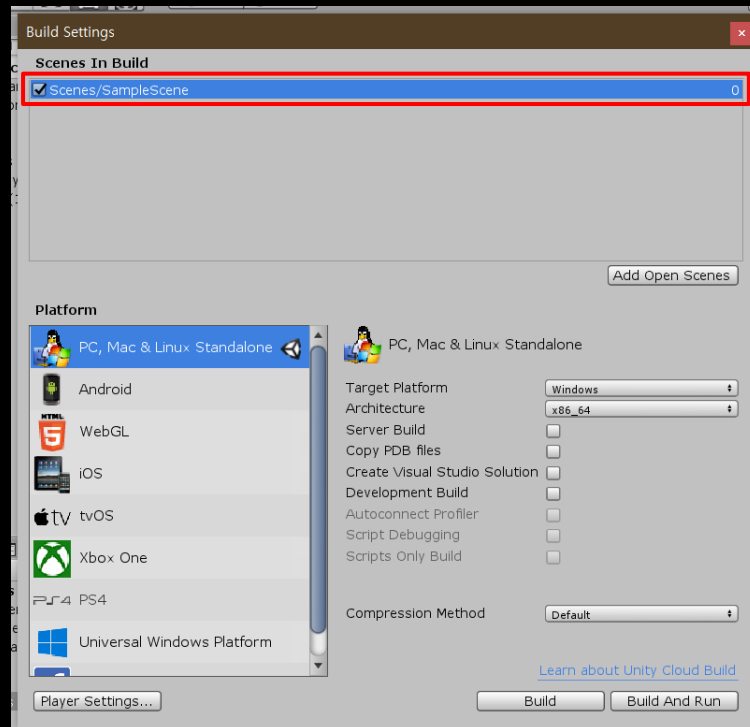
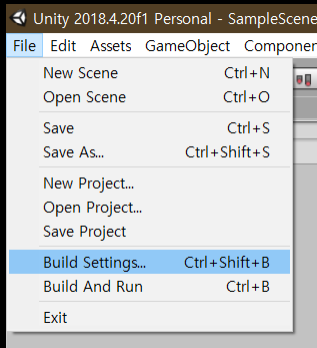


7. 썸 이동

로그인 -> 스테이지 선택 -> 레벨 1 -> ...

씬을 이동해보자

- *Edit > Build Settings > Scenes in Build*에 Scene을 추가하자
 - 이 Scene은 빌드용이라고 알려준다
 - 제일 위의 0번째 Scene = 게임을 시작했을 때 나타나는 Scene



SceneManagement

Scene을 지정하는 방법 2가지

1. 이름으로 ("Scenes/SampleScene")
2. Scenes In Build에 노출되는 숫자로 (0)

Scene을 옮길 땐 `UnityEngine.SceneManagement`를 사용한다

```
using UnityEngine.SceneManagement;
```


예제

```
using UnityEngine;~
using UnityEngine.SceneManagement;~
~
public class LoadSceneExample : MonoBehaviour~
{~
    public bool loadSceneByName = false;~
    public bool loadSceneByNumber = false;~
    ~
    // Update is called once per frame~
    void Update()~
    {~
        if (loadSceneByName)~
        {~
            SceneManager.LoadScene("Scenes/SampleScene");~
            loadSceneByName = false;~
        }~
        ~
        if (loadSceneByNumber)~
        {~
            SceneManager.LoadScene(0);~
            loadSceneByNumber = false;~
        }~
    }~
}~
~
```

버튼 클릭했을 때의 씬 이동



```
using UnityEngine;~  
using UnityEngine.SceneManagement;~  
~  
public class MyBtnClickEvents : MonoBehaviour~  
{~  
    public void OnClickStart()~  
    {~  
        SceneManager.LoadScene("Scenes/SampleScene");~  
        // 또는 SceneManager.LoadScene(0);~  
    }~  
}~
```

참고 문서

- Icons made by [Freepik](#), [wanicon](#), [Darius Dan](#), [Good Ware](#) from www.flaticon.com
- <https://docs.unity3d.com/Packages/com.unity.ugui@1.0/manual/script-CanvasScaler.html>