



유니티 2D 게임개발 프로젝트

# 유석의 유니티 모락모락 1

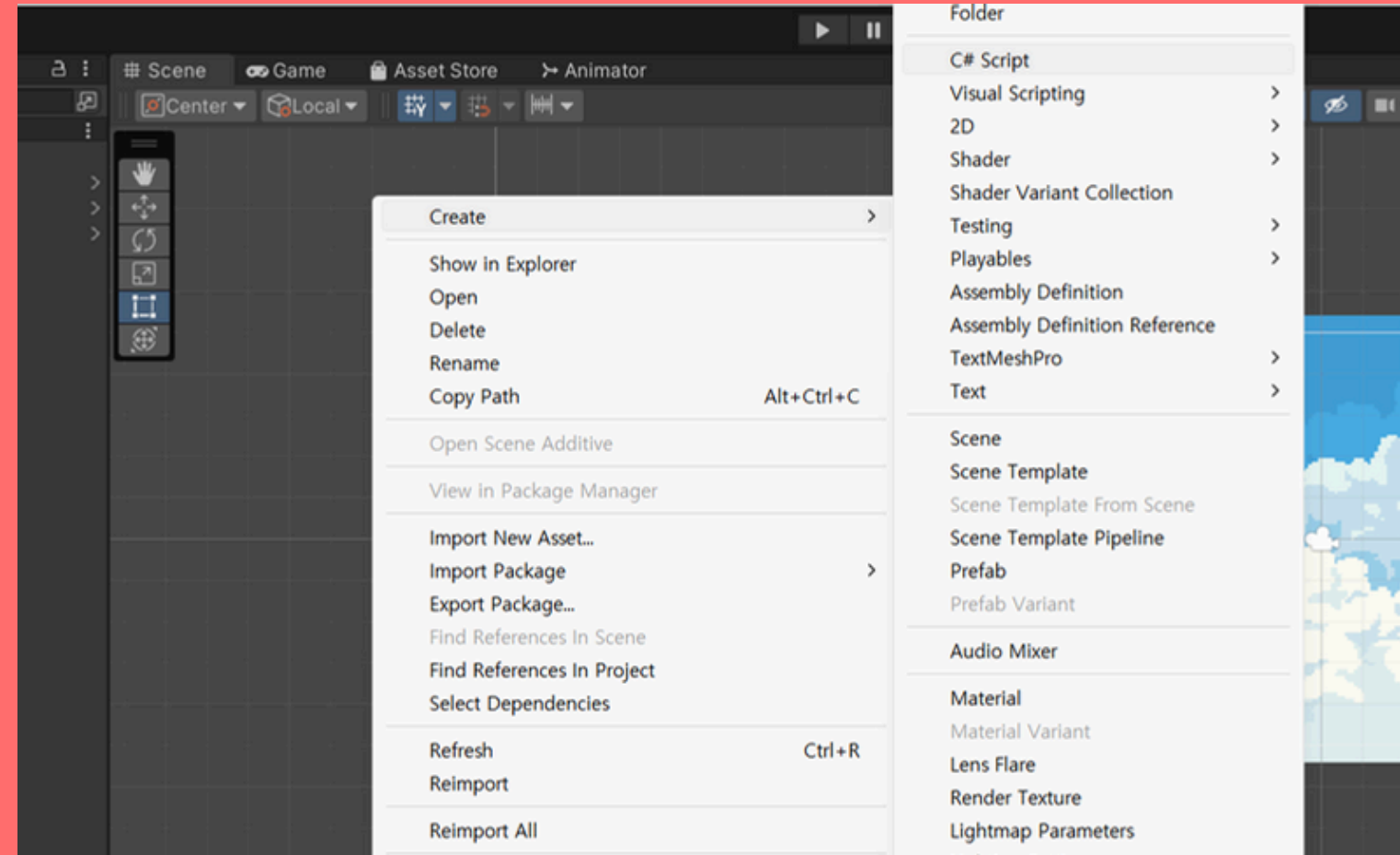
단국대 컴퓨터 그래픽 중앙동아리 CAGI



빠르게 배우는  
유니티 코딩

# 튜토리얼

## 새 유니티 스크립트 만들기



프로젝트 내 Asset폴더 안에 Scripts 폴더 생성 후  
우클릭 -> Create -> C# Script로 생성!

목차 페이지로 돌아가기



# 네임 스페이스

## using

- ① namespace - 미리 완성되어 있는 클래스나 메서드가 포함된 **그룹**
- ② using UnityEngine; using UnityEditor  
using UnityEngine.UI;
- ③ GameObject, Transform, MonoBehaviour 등 유니티에서 제공하는 기능들을 구현안해도 사용가능!
- ④ 내가 만든 다른 네임스페이스에서 동일한 이름의 클래스나 메서드를 또 생성할 수 있음

# MONO BEHAVIOR

## 생명 주기 함수

- ① 게임 오브젝트의 실행 순서를 제어하는 데 사용
- ② Awake() -> Start() : 각각 게임시작 전 후에서 호출
- ③ Update, FixedUpdate() -> LateUpdate() :  
일반과 fixed는 매 프레임단위로 실행됨! fixed는 물리 연산을  
진행함. LateUpdate는 Update 이후에 진행, 카메라 같은거
- ④ OnDestroy() : 스크립트를 넣은 오브젝트가 파괴될 때 수행

# MONO BEHAVIOR

## 코루틴(Coroutine)

```
IEnumerator MyCoroutine() {  
    yield return new WaitForSeconds(2.0f);  
    Debug.Log("2초 후에 실행됩니다.");  
}  
void Start() {  
    StartCoroutine(MyCoroutine());  
}
```

코루틴은 IEnumerator로 선언하고, yield로 특정 시간까지  
괄호 내 작업을 중단시킵니다.

다른 작업들을 하다가 의도한 시간에 이벤트 발생시킴!

\*Debug.Log는 유니티판 Console.WriteLine으로, 유니티 안에서  
개발자가 테스트를 위해 값 출력할때 사용



# 컴포넌트

## GetComponent<>

- ① 게임 오브젝트에 붙어있는 컴포넌트를 가져오는 방법
- ② `private Rigidbody2D rb; rb = GetComponent<Rigidbody2D>`
- ③ GetComponent해온 객체는 해당 컴포넌트의 전용 속성과 메소드에 접근 가능! Ex) `gravityscale`, `mass`, `AddForce` 등등..
- ④ `OnDestroy()` : 스크립트를 넣은 오브젝트가 파괴될 때 수행

## 충돌

### OnCollisionEnter

- ① Unity에서 물리 충돌이 발생했을 때 호출되는 함수
- ② 

```
void OnCollisionEnter2D(Collision2D collision){  
    Debug.Log(collision.gameObject.name);}

```
- ③ 자매품으로 OnTriggerEnter도 있음. 사용방법은 동일하나  
Collision은 뚫지 못하고 부딪히는거  
Trigger은 부딪히면 뚫고 지나친다 생각하면 편함
- ④ `collision.gameObject.CompareTag("Enemy")` 와 같은 방식으로 특정 태그를 가진 오브젝트와의 충돌 처리에 많이 사용

# 벡터

## Vector3

- ① 공간을 다루기 위한 기본 데이터 구조로, 세 개의 실수 (float)로 이루어진 벡터
- ② `Vector3 position = new Vector3(1.0f, 2.0f, 3.0f);`
- ③ 많이 사용되는 사용처로는 이동 위치 크기 조절이 있다.  
`GameObject.transform.position += new Vector3(1,0,0);`
- ④ Vector상수 개념도 존재한다. `Vector3.up`, `down`, `left`, `right` 등이 있고 해당 방향으로 직선이동 한다고 보면 됨

