

CIEN Unity 3D

중급반 (화요일 오후 7시)

고주형



목차

1. 물리엔진 개념
2. 강체(Rigidbody)
 - 뜻
 - velocity AddForce()
3. 충돌
 - Collider
 - Collision / Trigger
 - Enter / Stay / Exit

실습. Flappy Bird



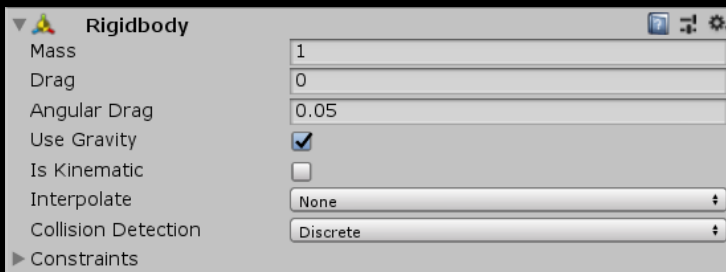
1. 유니티 물리엔진

물리 편하게 쓰자!

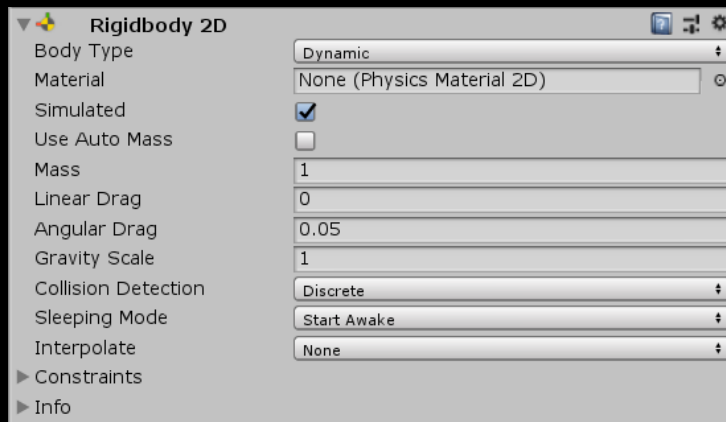
유니티의 물리엔진

- 유니티는 물리엔진이 내장되어 컴포넌트의 형태로 제공된다
- 우리는 물리를 구현하지 않고 컴포넌트를 사용한다
- 3D와 2D는 **다른** 물리엔진을 사용하기 때문에 주의하자
 - 주의해야할 차이점: 다른 컴포넌트를 사용한다

3D 물리엔진 컴포넌트



2D 물리엔진 컴포넌트



이벤트용으로 많이 사용

충돌 감지 (Enter? Stay? Exit?)



- 총알이 맞았을 때
 - 소리
 - 이펙트
 - HP--
- 특정 영역에 들어왔을 때
 - 이동
 - 디버프

물리엔진의 계산 비용

- 계산 비용이 높다
- 일반적인 FPS보다 낮게 설정되어 있음

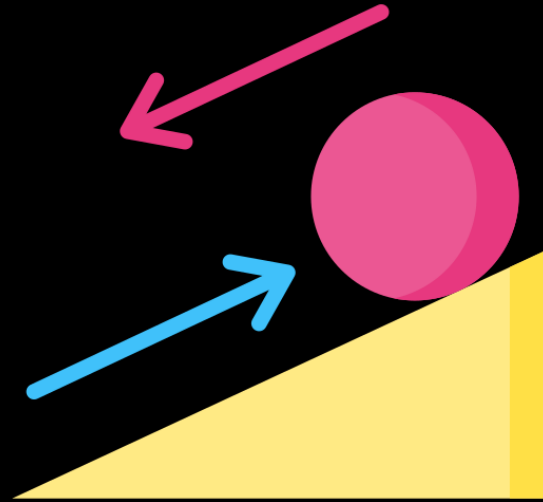
Description

Frame-rate independent [MonoBehaviour.FixedUpdate](#) message for physics calculations.

[MonoBehaviour.FixedUpdate](#) has the frequency of the physics system; it is called every fixed frame-rate frame. Compute [Physics](#) system calculations after [FixedUpdate](#). [0.02](#) seconds (50 calls per second) is the default time between calls. Use [Time.fixedDeltaTime](#) to access this value. Alter it by setting it to your preferred value within a script, or, navigate to Edit > Settings > Time > Fixed Timestep and set it there. The [FixedUpdate](#) frequency is more or less than [Update](#). If the application runs at 25 frames per second (fps), Unity calls it approximately twice per frame. Alternatively, 100 fps causes approximately two rendering frames with one [FixedUpdate](#). Control the required frame rate and [Fixed Timestep](#) rate from [Time](#) settings. Use [Application.targetFrameRate](#) to set the frame rate.

Use [FixedUpdate](#) when using [Rigidbody](#). Set a force to a [Rigidbody](#) and it applies each fixed frame. [FixedUpdate](#) occurs at a measured time step that typically does not coincide with [MonoBehaviour.Update](#).

- 고정된 시간 0.02(1초에 50번) 호출함
 - 물리엔진을 사용할 때는 Update가 아닌 FixedUpdate에서 호출해주자
- 물리 연산이 중요하지 않으면 조정해서 최적화 할 수 있다
 - Edit > Settings > Time > Fixed Timestep



2. 강체 (Rigidbody)

물리

강체 (Rigidbody) ?

(물리) 뜻:

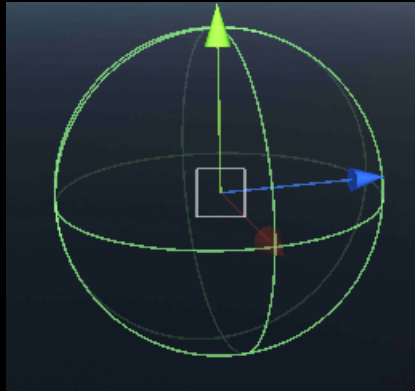
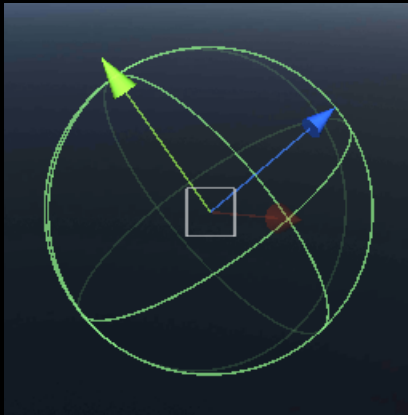
- 외력을 가해도 크기나 형태가 변하지 않는 이상적인 물체
- 고등학교 물리에서 모든 물체를 강체라고 가정하고 역학을 배움
- 외력을 가할 때 = 움직일 때

(유니티) 뜻:

- Rigidbody Component가 있다 = 움직이는 물체이다
- Rigidbody Component가 없다 = 움직이지 않는 물체이다

강체 (Rigidbody)를 쓰는 이유

- 물리엔진에서 이동/회전을 표현하기 위해



- 정적인지 / 동적인지 구분하기 위해
 - 정적 -> 연산량↓ (미리 계산함)

물체 이동하기

- Transform과 달리 컴포넌트를 우리가 직접 가져와야 한다

```
rb = GetComponent<Rigidbody>()
```

- Quiz. 왜 유니티가 자동으로 가져와주지 않을까?

- `rb.velocity = new Vector(1, 0, 0);`

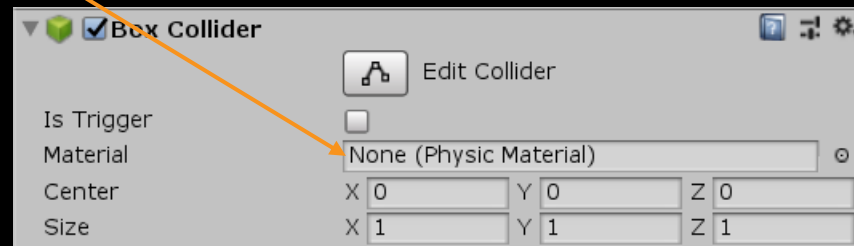
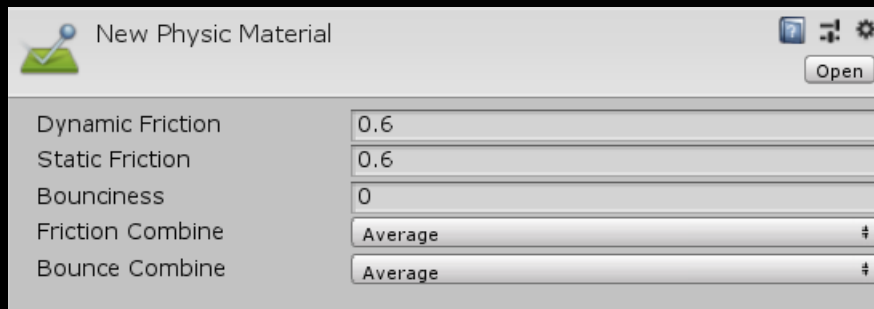
- `rb.AddForce(1,1,1);`

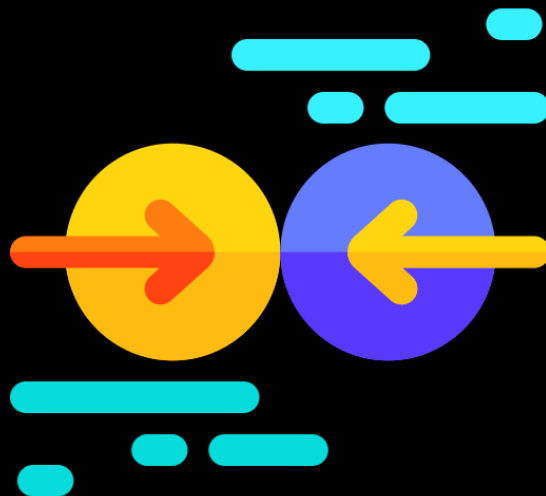
```
void Start()
{
    GetComponent<Rigidbody>().AddForce(1, 1, 1);
}
// Update is called once per frame
참조 0개
void Update()
```

Visual Studio Tip.
Ctrl + Shift + Space

더 많은 조정을 원한다면?

- Physic Material이란 것이 있다.



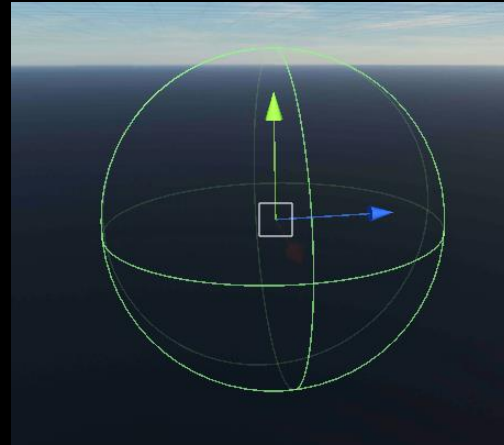
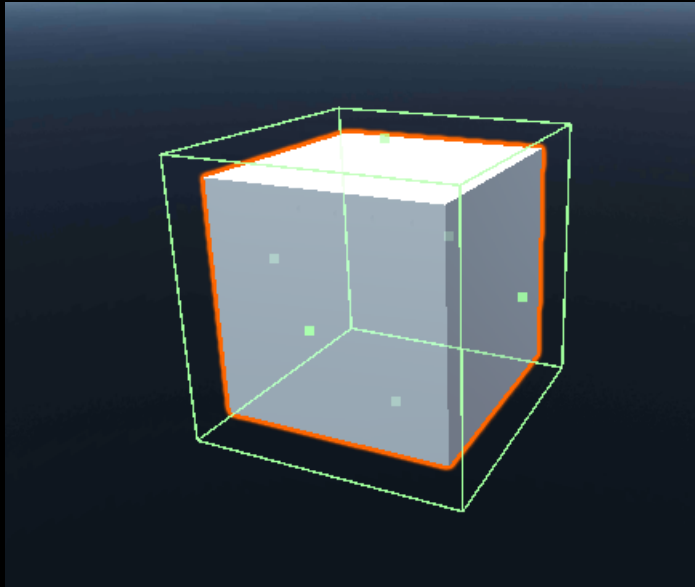


3. 충돌

충돌하는 물체 - Collision vs 감지하는 물체 - Trigger

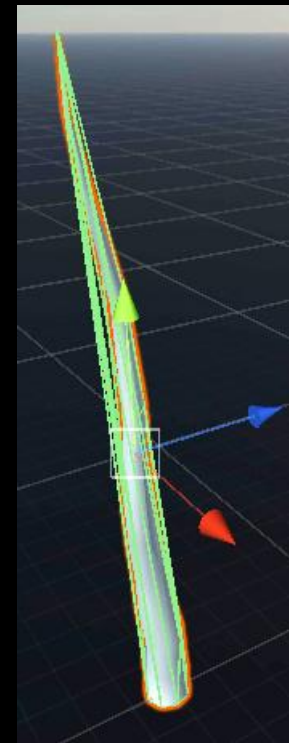
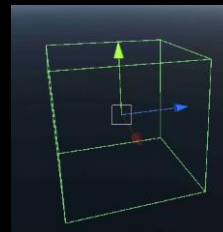
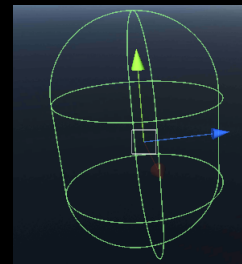
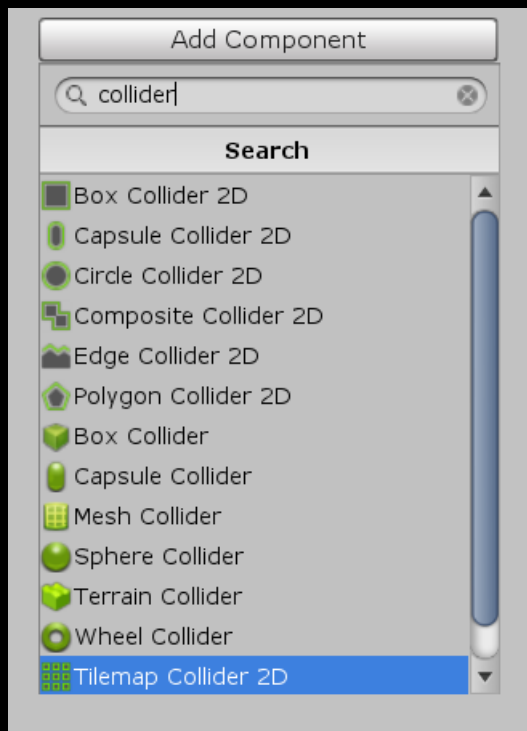
Collider 컴포넌트

- 충돌체라는 뜻
- 실제로 충돌할 부분 (초록색 선)



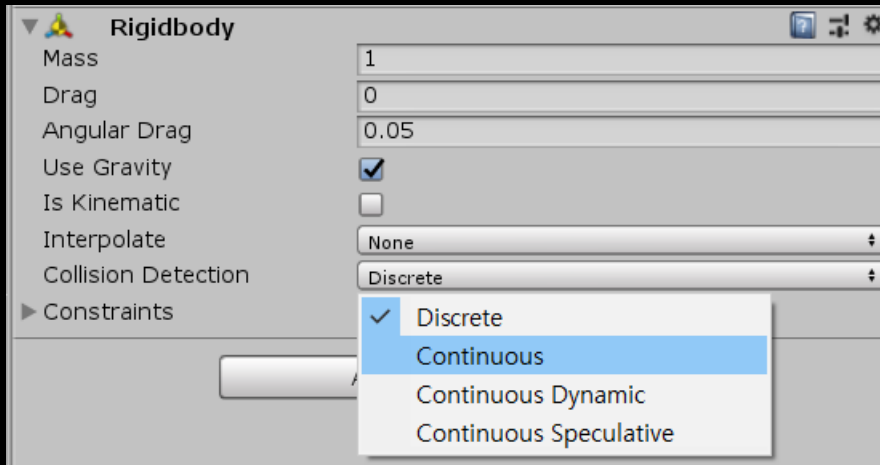
Collider의 종류

- 연산량? 구 < 캡슐 < 박스 <<< 메쉬



속도가 너무 빠를 경우 Collider Discrete vs Continuous

1. Discrete: 이산적으로 이동
 2. Continuous: Dynamic → Static
 3. Continuous Dynamic: Dynamic ↔ Dynamic
- 하지만... Continuous하게 계산하기 보단
 - Collider를 크게 잡자



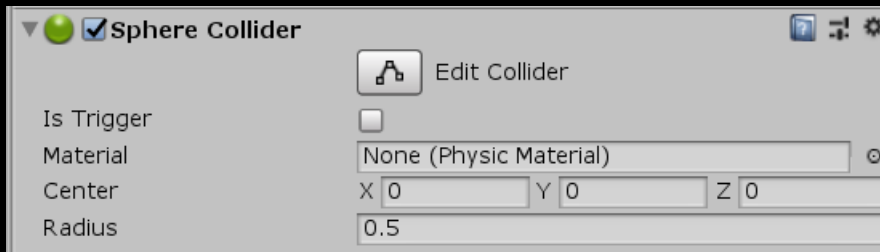
캐릭터용 Collider

- Mesh보단 Capsule



Collision

- Collider의 "Is Trigger"를 사용하지 않는 경우



- Collision을 함
- 물리적인 충돌을 한다



Trigger

- Collider의 "Is Trigger"를 사용하는 경우



- Trigger를 함
- 들어왔는지 나갔는지 감지만 한다



Enter / Stay / Exit

- Enter: 충돌을 시작할 때
- Stay: 충돌 중일 때
- Exit: 충돌이 끝날 때

- OnTriggerEnter(충돌되는 물체)
- OnTriggerStay(충돌되는 물체)
- OnTriggerExit(충돌되는 물체)

→ **Collision Object**

- OnCollisionEnter(충돌되는 물체)
- OnCollisionStay(충돌되는 물체)
- OnCollisionExit(충돌되는 물체)

→ **Collider Object**

실습해보자

! [06:51:57] Hi, Cube (1)
UnityEngine.Debug:Log(Object)
! [06:51:58] Stay, Cube (1)
UnityEngine.Debug:Log(Object)
! [06:51:58] Bye, Cube (1)
UnityEngine.Debug:Log(Object)

```
private void OnTriggerEnter(Collider other)
{
    Debug.Log("Hi, " + other.gameObject.name);
}

private void OnTriggerStay(Collider other)
{
    Debug.Log("Stay, " + other.gameObject.name);
}

private void OnTriggerExit(Collider other)
{
    Debug.Log("Bye, " + other.gameObject.name);
}
```

충돌 이벤트가 발생하지 않을 때

- Rigidbody를 제대로 달았는지 확인한다
- Quiz. Rigidbody가 없이 Collider만 있으면?

Flappy Bird 점프 구현하기

- Collider로 못 나가도록 설정
- Player가 Collider에 충돌했을 때 "충돌함" 출력
- Collider에 충돌해도 이상하게 돌아가거나 튕기지 않음
- GitHub에 gitignore까지 잘 설정해서 올린다

참고 문서

- Icons made by [Freepik](#), [photo3idea_studio](#) from www.flaticon.com
- <https://docs.unity3d.com/ScriptReference/MonoBehaviour.FixedUpdate.html>
- <https://docs.unity3d.com/ScriptReference/CollisionDetectionMode.Continuous.html>