

# 언리얼 프로그래밍 Part2-7

## 제목: 캐릭터 스탯과 위젯

\*\*강의 내용

### 강의 내용

캐릭터 스탯 컴포넌트의 구현과 HP를 표시하는 위젯의 구현



\*\*강의에서 다루는 게임 프레임워크 요소

### 강의에서 다루는 게임프레임워크 요소

게임	월드	모드	상태	
기믹	트리거	스폰	물리	
플레이어	입력	카메라	HUD	상태
폰	이동	모션	액션	위젯
데이터	애셋	테이블	설정	저장
인공지능	길찾기	BT		

이번 강의에서 다루는 게임프레임워크 요소를 정리해보았습니다

## \*\*강의 목표

### 강의 목표

---

- 액터 컴포넌트를 활용한 액터 기능의 확장 방법의 이해
- 언리얼 델리게이트를 활용한 발행 구독 모델의 학습
- 액터의 초기화 단계와 위젯 초기화 과정의 이해

inflearn  
1138208  
이득우의 언리얼 프로그래밍 Part2 - 언리얼 게임 프레임워크의 이해  
이득우

액터 기능을 확장하는 방법을 이해하는 것이 첫 번째 목표고요

### \*\*\*캐릭터 스탯의 설정

#### 액터 컴포넌트를 활용한 스탯의 설계

- 액터에 부착할 수 있는 컴포넌트 중 트랜스폼이 없는 컴포넌트
- 액터의 기능을 확장할 때 컴포넌트로 분리해 모듈화할 수 있음
- 스탯 데이터를 담당하는 컴포넌트와 UI 위젯을 담당하는 컴포넌트로 분리
- 액터는 두 컴포넌트가 서로 통신하도록 중개하는 역할로 지정



캐릭터가 가지고 있는 Hp와 같은 스탯 정보를 관리하기 위해서는

-캐릭터가 가지고 있는 HP와 같은 스탯 정보를 관리하기 위해서는 캐릭터 액터내에 구현할 수도 있지만, 별도로 액터 컴포넌트를 활용하여 해당기능을 분리하여 구현이 가능하다.

-액터 컴포넌트는 액터에 부착할수 있는 컴포넌트중에서 트랜스폼이 없는 컴포넌트를 의미한다.

-우리는 앞으로 캐릭터에 두 가지 컴포넌트를 부착해볼 것이다.

#### ■ 스탯 데이터를 담당하는 스탯 컴포넌트

:스탯 컴포넌트는 기능만 담당하기 때문에 액터 컴포넌트로 구현이 가능하다

외부로부터 이벤트가 발생하면 다른캐릭터로부터 데미지를 받는 경우에는 HP값인 스탯이 갱신 된다.

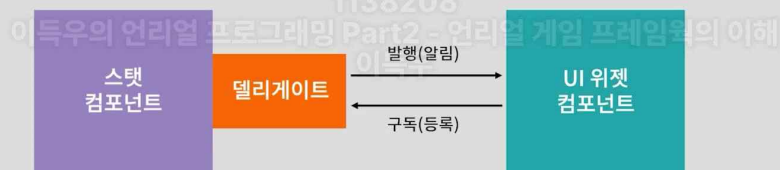
#### ■ UI를 담당하는 UI 위젯 컴포넌트

:데미지를 받아 HP값인 스탯이 갱신될 때 UI 위젯 컴포넌트가 자동으로 갱신 된다.

**\*\*언리얼 델리게이트를 활용한 발행 구독 모델의 구현**

## 언리얼 델리게이트를 활용한 발행 구독 모델의 구현

- 푸시(Push)형태의 알림(Notification)을 구현하는데 적합한 디자인 패턴
- 스텟이 변경되면 델리게이트에 연결된 컴포넌트에 알림을 보내 데이터를 갱신
- 스텟 컴포넌트와 UI 컴포넌트사이의 느슨한 결합의 생성



### 우리가 구현할 스텟 컴포넌트와

-우리가 구현할 스텟 컴포넌트와 UI 위젯 컴포넌트는 데이터를 주고받지만 서로를 참조할 이유가 없다.

-각각은 자신에게 부여된 기능을 묵묵히 수행만 하면 된다.

-이때 언리얼 엔진의 델리게이트를 활용하여 서로의 존재를 몰라도 데이터를 편리하게 주고받을 수가 있다.

-델리게이트를 사용하여 데이터를 주고받는 것을 발행 구독 모델이라고 한다.

-발행 구독 모델은 푸시 형태의 알림을 구현하는 데 적합한 디자인 패턴이다.

-스텟 컴포넌트에 델리게이트를 선언해주고, 스텟이 변경되면 직접적으로 위젯 컴포넌트에 연결하는 것이 아니라 위젯 컴포넌트가 델리게이트에 미리 등록을 해두고 델리게이트로부터 알림을 받아 해당 내용을 업데이트 하도록 설계해주면 된다.

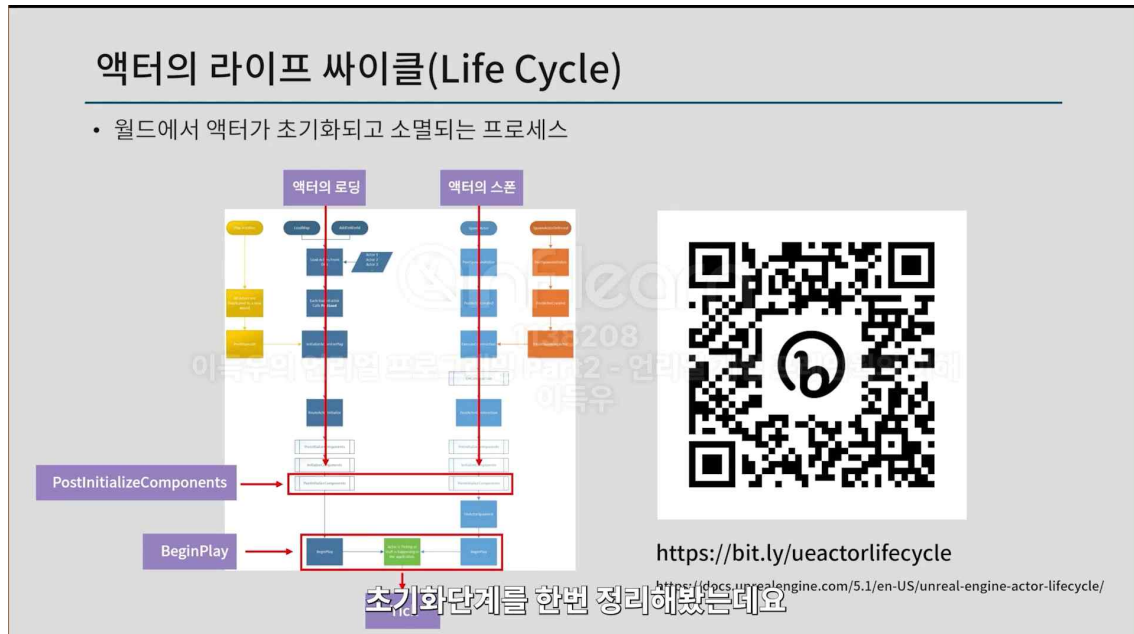
-이러한 구조를 느슨한 결합이라고 한다.

-먼저 스텟 컴포넌트와 UI 위젯 컴포넌트를 구현해보도록 하자.

-위젯을 캐릭터에 부착할 때 위젯 자체로는 부착할 수 없기 때문에 위젯 컴포넌트라고 하는 언리얼 엔진이 제공하는 컴포넌트를 활용해야 한다.

\*\*\*액터의 초기화 과정

\*\*액터의 라이프 싸이클



공식 사이트 주소 : [https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-actor-lifecycle?application\\_version=5.1](https://dev.epicgames.com/documentation/en-us/unreal-engine/unreal-engine-actor-lifecycle?application_version=5.1)

-액터가 초기화 될 때는 디스크에 저장된 레벨 정보가 로딩이 되면서 초기화 되는 과정이 있고, 아니면 스크립트를 사용해서 런타임에서 생성하는 스폰과정 두가지가 있다.

-이 두가지 과정이 각각 조금씩 다르게 초기화가 진행이 된다.

-어쨌든 우리가 알아야 하는 것은 거의 마지막 단계에서는 PostInitializeComponents란 함수가 호출이 된다는 것이다. 이 함수는 말 그대로 모든 컴포넌트들이 초기화가 완료된 뒤에 호출이 되는 함수이다. 이 함수가 실행이 되면 그 다음으로는 BeginPlay함수가 실행이 되는데 이때부터는 Tick이 발동이 된다.

-따라서 우리가 액터를 최종적으로 마무리 하고자 할 때는 PostInitializeComponents함수를 통하여 마무리를 진행하고 시작할 때 먼가 초기화를 진행하고자 할때는 BeginPlay에서 작업을 해주면 된다.

-한끝 차이이지만 Tick이 시작이 된다 안된다 라는 점에서 사용 방법에 있어서 차이가 있다.

## \*\*위젯 컴포넌트와 위젯

### 위젯 컴포넌트와 위젯

- 위젯 컴포넌트는 액터 위에 UI 위젯을 띄우는 컴포넌트
- 3차원 모드와 2차원 모드를 지원함
- 위젯 컴포넌트는 컨테이너 역할만 할 뿐, 둘은 서로 독립적으로 동작함



여기에서 위젯 컴포넌트와 위젯의 관계에 대해서 한번 정리해봤습니다

-위젯 컴포넌트는 액터 위에 UI위젯을 띄워주는 컴포넌트에 불과

## \*\*위젯 컴포넌트의 초기화 과정

### 위젯 컴포넌트의 초기화 과정

- 발행 구독 모델의 구현을 위해 위젯 컴포넌트의 초기화 단계를 파악할 필요가 있음.
- UI 관련 컴포넌트는 액터의 BeginPlay 이후에 호출되고 있음.
- 생성시 InitWidget 함수와 NativeConstruct 함수를 호출
- 차후에 변경될 가능성도 염두해 두어야 함.



-앞서서 발행구독 모델을 구현하여 스탯 컴포넌트의 데이터가 업데이트 될 때 자동으로 위젯이 갱신되도록 기능을 구현해야 하는데 이것을 위해서는 스탯 컴포넌트의 존재를 위젯이 알아야 한다.

-스탯 컴포넌트의 경우에는 PostInitializeComponents에서 모든 세팅이 종료가 되는데 우리가 HpBar를 구현한 유저 위젯의 경우에는 BeginPlay 이후에 생성이 되기 때문에 유저 위젯과 스탯 컴포넌트를 연결 시켜줄 수 있는 적당한 타이밍이 필요하다 이때 사용할 수 있는 것이 위젯 컴포넌트의 InitWidget함수와 유저 위젯의 NativeConstruct함수가 있다.

-문제가 있는데 유저 위젯의 경우에 내가 등록을 하기 위해 액터에 속한 스탯 컴포넌트의 존재를 어떻게든 가져와야 하는데 이 정보를 얻기 위해서는 자신을 소유하고 있는 위젯 컴포넌트들로부터 액터의 정보를 얻어와야 하는데 이것을 알수가 없다. (왜인지는 모르나 현재 언리얼 엔진 버전에서는 그 정보를 알수 없도록 설계가 되어 있음)

-그래서 이것을 구현하기 위해 위젯 컴포넌트와 유저 위젯을 확장하여 해당 정보를 받아올 수 있도록 클래스를 확장할 필요가 있다.

-그래서 현재 사용하고 있는 위젯 컴포넌트를 확장한 ABWidgetComponent 클래스를 선언해주고 유저 위젯도 ABUserWidget이라는 클래스로 확장해야 한다.

-이렇게 해서 유저 위젯이 액터의 정보를 얻어오면 현재 유저 위젯이 가지고 있는 정보를 등록할 수 있도록 새로운 인터페이스를 만들고 인터페이스에서 제공하는 SetupCharacterWidget()이라는 함수를 호출하여 자신의 멤버함수를 스탯의 델리게이트에 등록하자.

-실습 진행중 Physics나 UI폴더는 우리가 만든 기본 컴포넌트(엔진)들이라 생각하자.

**\*\*정리**

## 액터 컴포넌트를 활용한 기능 모듈화

---

1. 액터 컴포넌트를 사용해 캐릭터가 가진 기능을 분산
2. 언리얼 델리게이트를 활용한 발행 구독 모델의 구현
3. 위젯 컴포넌트 초기화 시점을 파악하기 위한 기존 클래스 구조의 확장 설계



1138208

이득우의 언리얼 프로그래밍 Part2 - 언리얼 게임 프레임워크의 이해  
이득우

이번 강의에서는 액터 컴포넌트를 활용한