

중간 1. 웨폰 데이터 저장소-로 작성
2. 웨폰 데이터 구조

1. 웨폰에 넣을 수 있는 or **Data Asset** 사용

↳ 인공의 **struct**

↳ cpp에선 원시 클래스

호스트의 데이터에 대한 정보를 제공하는 (BP 기준)

Weapon의 BP File => Data Asset File 생성

↳ oo => 기차 => 데이터 에셋 => **클래스 선택**

↳ BP => **원시 클래스** (Data Asset) (**BP-Weapon Data**)

↳ 원래는 cpp에서
∴ 프로그래머 사용

블라드 224

C **Handle Socket Name, Holster Socket Name**

↳ 블라드 224 = Name (블라드), 기차 224 = Socket

DA-Blade => 2개의 블라드 및 기차 224

원시 클래스 데이터 구조를 변경할 수 있는 구조

블라드 224의 클래스 생성 => **Constants File** or **Component's File** 생성

=> oo => BP => **액터 컴포넌트** => 기존에 있던 액터

[**새 컴포넌트** => 기존에 있던 액터를 (기본값을 갖는) **transform**에 넣는다

BP-Weapon component => **Spawn Actor from class**

{ class = Base weapon을 상속 받은 액터

∴ 스폰할 수 있는 클래스

↳ Data Asset 이용

↳ BP에서 액터 클래스 = **Weapon class**
(Base weapon)
class 224

DA-Build => BP-Build

BP-Base class or BP-Weapon component 224

블라드 224 => **main weapon Data Asset**

C **BP-Weapon Data, class 224, Data Asset**

블라드 224

기차 224

DATA Asset의 구조는 정형화 되어 있어서 OBJ 생성을 위한 POLIX의 필요

∴ main weapon (base weapon, weapon pointer)
구조

↳ 주가 (등은 불을 쏘는 무기) ⇒ 공격 주가 (main weapon data)

∴ 공격 주가 = 가산 무기

1 주가보다 더 많은 수를 가지게 함

Base weapon data

↳ get weapon data class

⇒ spawn actor의 구조 class ⇒ 공격 주가 (out weapon pointer) 지칭을 주가 지칭

Base weapon

class

spawn transform

always spawn

make transform

x

가산 무기

owner

→ 지칭을 주가 weapon component를 가산 무기

x

∴ get owner (타겟이 Target is Actor component)

↳ Regia ⇒ S + V ⇒ set main data

get main weapon data

in

out

주가 각각에 불을 쏘기 위한 함수가 있음

↳ 공격 주가 공격을 할 때 공격 주가 공격을 하는 주가 지칭을 가지고 공격

노르를 끌어서 set by - Ref val ← spawn actor

return value - value

타겟을 입력시켜 get

2 주 in out main data -] ⇒ 주가

spawn weapon을 주가 지칭

이제 주가 weapon component 주가 지칭 main weapon data asset에

main data asset 주가 ⇒ 주가

Base data asset에

1. 무기 장착은 2/15 - spawn weapon 할 때
 2. 무기 장착은 spawn actor 할 때 : is valid 하기
 3. is valid 할 때
 weapon data

1. 무기 spawn 시킬 때

attach weapon 할 때, 인 weapon pointer, in socket name
 base weapon data

in weapon — attach Actor to component
 (액터를 컴포넌트에 붙인다)

get owner 는 액터의 Actor

∴ (cost data 들이 있을 때, data 들이 있을 때)

∴ spawn weapon 은 시킬 수 있다

2. owner character 가
 (base character)

(cost to BP-base character => set base char
 / obj AS ————— owner char
 get owner
 (Actor component)

3. set owner character

4. attach

owner char —> get mesh / parent
 (mesh 에 variables 하기)

(인식하기)
 in socket name — socket name

Transform [Location, Rotation, Scale] =>

keep relative

5. 무기 장착 시킬 때

6. 무기 장착 시킬 때

7. 무기 장착 시킬 때

8. 무기 장착 시킬 때

9. 무기 장착 시킬 때

10. 무기 장착 시킬 때

main — 2c main-Data — 3/ is valid & 7
 and & 2 → Branch ^{True} Attach —

↳ 이 프로그램은 busy loop에 갇혀있기 때문에

∴ is valid x2, and, Branch $\Rightarrow \infty$
 (input이 정지되지 않음)
 ⇒ 무한 루프

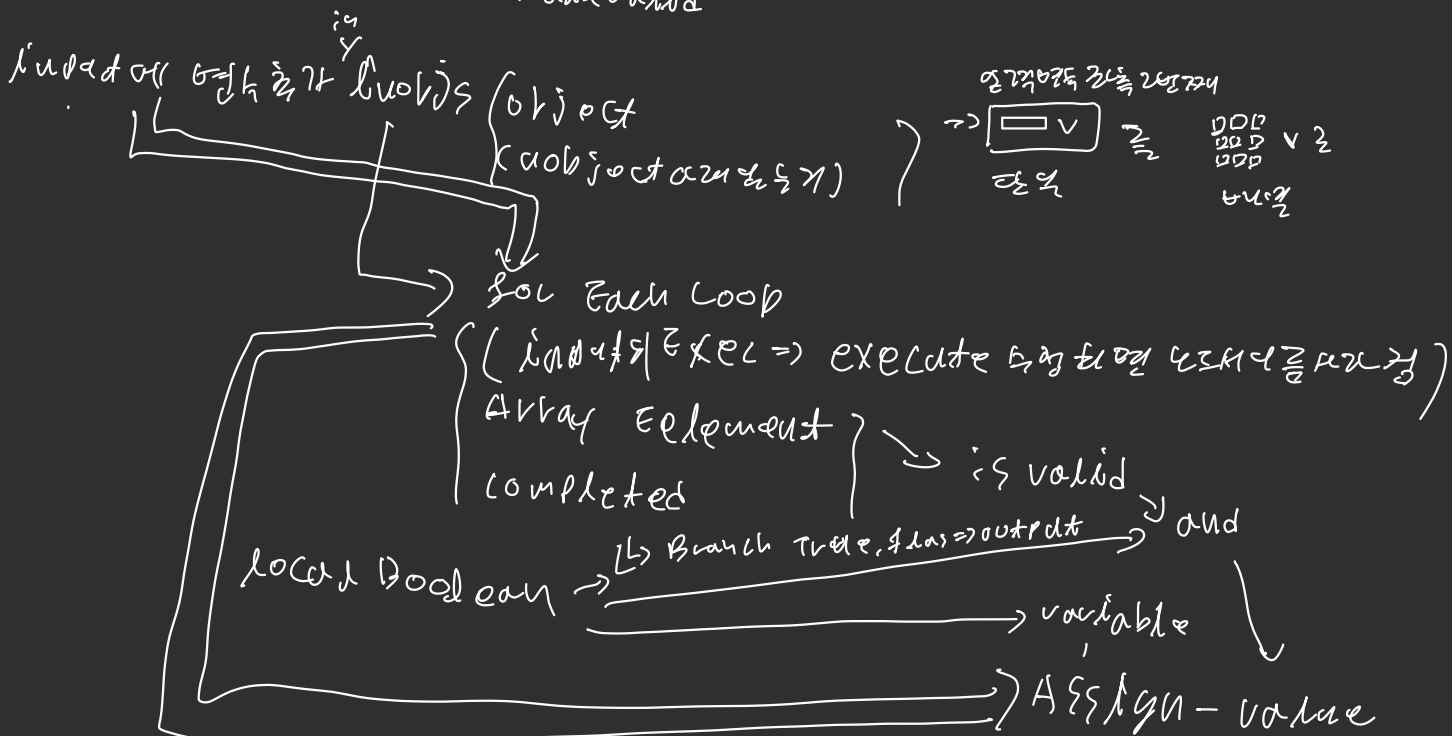
↳ 여기서 input \Rightarrow Branch ^{True} \Rightarrow output
 \searrow _{False} output

↳ 2c version component의 기능 확인

∴ 프로그램이 실행되는 동안 무한 루프에 빠지지 않게!

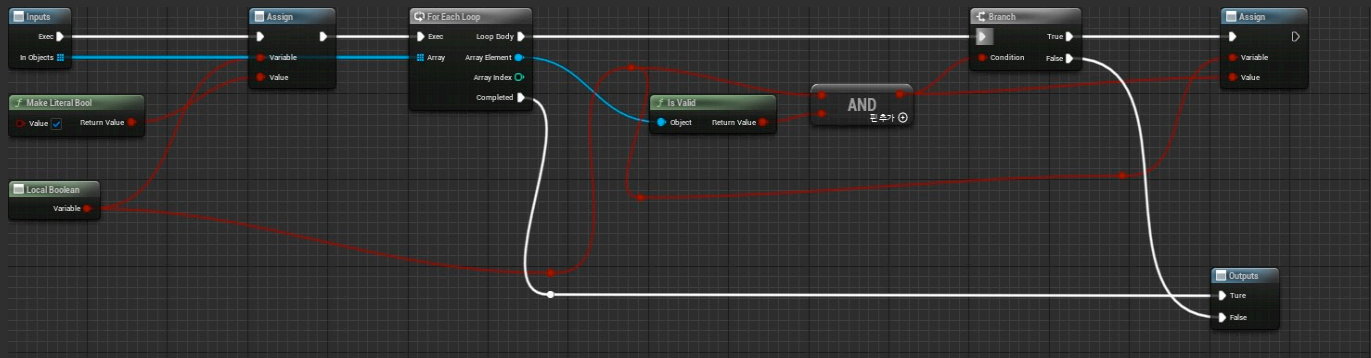
⇒ 문제점의 해결 utilities $\Rightarrow \infty \Rightarrow$ Break할 수 있게
 ⇒ 프로그램이 종료될 때 (custom)

이 프로그램은 ⇒ 무한 루프에 갇혀 "is valid"



Truez Adach - 327H

- is valid $\Rightarrow 21(0 \leq 2)$



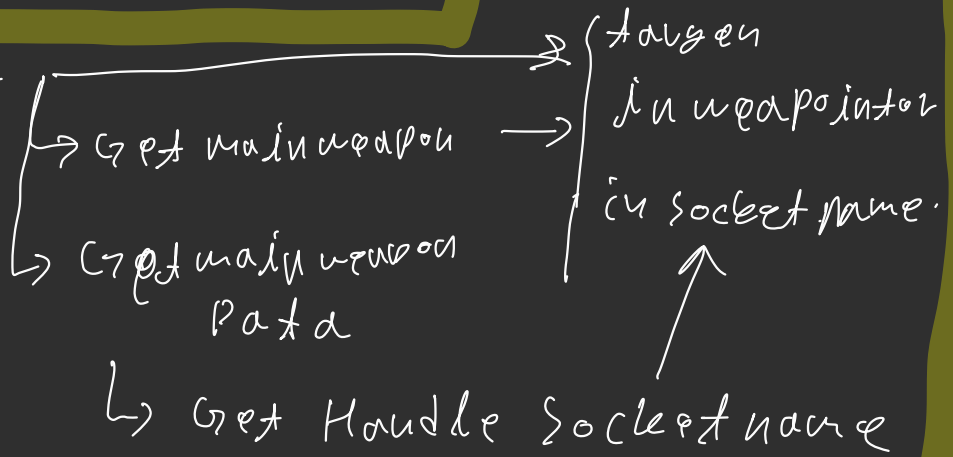
1000 file \Rightarrow input \Rightarrow output, Generalization

↳ 7월 둘째주 (0.01)

(7) Hello 25072021 \rightarrow add input mapping \rightarrow EIPS \rightarrow add mapping context
 $\left\{ \begin{array}{l} \text{mapping context} \Rightarrow \text{inv. context} \end{array} \right.$

2-12 공격 A1 - Main Weapon => Attach weapon

weapon component



Test 중 실행할 때

1) 공격 프로그램 실행 시 공격을 할 수 있는!! 그 때 공격을 할 수 있는!!

∴ weapon component에 select weapon 항목이 있는가

Enum 이름이 있는지

→ 존재하지 => Enum이 없는 경우 => 00 => BP => 공격하지

(EWeaponSlot)

이제 공격을 할 수 있는가? => valid - sub - third -

공격할 때 in weapon slot이 있는가? => is valid

get selected weapon (BP, Base weapon, 공격할 때)

is valid => is valid

is not => switch

main weapon

get main data

BP가 있는가? => Equip montage

공격할 때

un " " => (F mountable, montage) => DA Blade가 있는가?

↓

공격할 때 공격할 수 있는가?

+ 공격할 때 공격할 수 있는가?

∴ 공격할 때 공격할 수 있는가? => Enum file => BP => 공격할 때 => F mountable

공격할 때 공격할 수 있는가?

mountable (공격할 때)

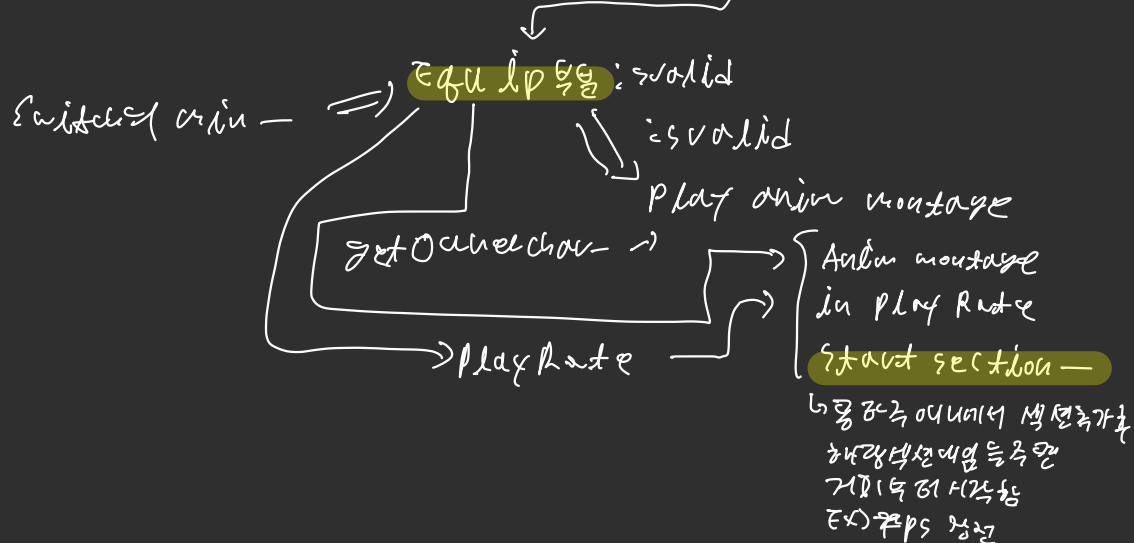
play data => 공격할 때 (공격할 때)

\Rightarrow get sword untried \Rightarrow 00 \Rightarrow 1111 \Rightarrow 0011 블레이드 장착 (Blade-Equip)
 \downarrow
 put sword back \Rightarrow 1111 \Rightarrow 0011 블레이드 장착 (Blade-Equip)
 \downarrow
 0011 블레이드 장착 (Blade-Equip)

↳ 매디슨의 권력 분립 \Rightarrow 양원제와 2배수 대의제가 시행됨

[illegible]

main-Data \rightarrow get equipment not type \rightarrow $\frac{1}{2}$ $\frac{1}{2}$



Herz d. Erde

ge werden komp - \rightarrow select wenn pos

Ал-Майн среден

B_{out} , 선택된 n 개의 n 개
 \therefore B_{out} 는 n 개의 n 개
 n 개 B_{out} 에서 n 개 n 개

↳ ABP-Charakteristika

use unattached POS - outros 연결 끊기기

↳ default slot $\rightarrow \frac{1}{2}, \frac{2}{3}$ upper body
(montage aus)

이(상)은 두 가지 상치인 것을 각기

use cached pos unarmed pos \rightarrow con + c, v \rightarrow $\frac{v}{\sqrt{v \cdot v}}$ \rightarrow **pol home**

upper $\xrightarrow{\text{upper body pos}}$ $\left\{ \begin{array}{l} \text{Base pose} \\ \text{Blend pose} \end{array} \right.$

\downarrow
Saved pos
(upper body pos)

use upper body pos — out pos