

캐릭터 위치를 넘겨줌 \Rightarrow 캐릭터 움직이기 \Rightarrow 이동할 때 교점 체크
게임 매 프레임 (HP, MP 등) 갱신하기

HUD를 **HUD** 인스턴스를 만들어 놓아서

↓
관리하기

원래 만렙 갯수

\therefore 고정된 값

\Rightarrow $C \neq \Rightarrow$ $R \neq \Rightarrow$ $H \neq \Rightarrow$ HUD 갱신

게임 모드 \Rightarrow Game Mode \Rightarrow HUD 갱신

게임 진행

.. \Rightarrow 스킬 갯수

유저 인터페이스 \Rightarrow UI BP \Rightarrow 스킬 UI 갱신

\Rightarrow HUD의 UI obj

Full Screen \Rightarrow 모든 UI를 화면에 가득 채워 표시

Desired \Rightarrow 원하는 해상도 크기

\hookrightarrow 해상도 \Rightarrow UI \Rightarrow **UI** \Rightarrow UI

↓

\hookrightarrow 해상도 \Rightarrow UI \Rightarrow UI

Border \Rightarrow UI \Rightarrow UI

\hookrightarrow UI \Rightarrow UI

\hookrightarrow UI \Rightarrow UI

정렬 \Rightarrow $(0.5, 0.5)$ \Rightarrow UI

\hookrightarrow UI

\hookrightarrow UI \Rightarrow UI

\hookrightarrow UI \Rightarrow UI

그 때쯤엔 영어가사지 올 댔 하긴 하여 | 프로그래머는
리라이터의 프로그래머가 되길 바라는 마음과
이제!
이제 영어가사지 올 댔 하긴 하여 | 프로그래머는

이제 영어가사지 올 댔 하긴 하여

그 때쯤엔 영어가사지 올 댔 하긴 하여 **Custom Event** **영어가사지** => press jump 은 이걸
→ 이걸로 된 것만 생각해서
놓을 생각은 되지는 않음

Release jump 하거나 생각

press jump 하는 것 (리라이터/리라이터에서 올 것들)
영어가사지 처음 올 때 (리라이터) max 가 되는 순간 set press count 하는 것
HUD에 나타나는 것 (리라이터에서 올 것들)

BP_Hud에 jump gauge를 그려서 **press jump** 하는 것

→ 이 jump value를 Hero가 가지고 있음

oppress jump

∴ **Custom Event** 이용

이 jump value를 영어가사지 oppress jump에 넣어
oppress jump에 연결하는 것

Hero에 추가

→ 00 => **get player count** => **get hud**

→ AI에겐 HUD가 필요하지

∴ 이걸로 설정한 HUD는

→ 이 HUD는

∴ **캐릭터의**

cast bp hud => press jump

Branch 이걸 set 된 value를 press jump에 넣어

이제 100% 성공을 할 수 있는 것들 '승승장군' 중 하나인 것들

But, 이걸로 인해 게임이 느려짐

Hud의 Release — 지름

"on Release Jump" Event 함수

Jump Gauge에서 Release Jump 실행

지름 헬로워러의 **cost Hud** (오른쪽 Set 오른쪽에서)

↓
AS BP Hud에서 on Release Jump 실행

지름은 애니메이션을 잘 처리한다

WB - Jump Gauge에 **Get Block Gauge** ⇒ get player Animation
⇒ press Jump 할 때

But "start at —" 때문에 "play animation forward" 사용

↳ 왜냐하면 "reverse" 사용

(Release Jump에 불리)

지름의 플레이를 잘 보도록

∴ "do once" 사용 (각 forward, reverse) 잘 처리

2 $\frac{1}{2}$ forward 2 reverse → do once의 result로
2 번만 실행

∴ 애니메이션 잘 하고 리셋 되고

바뀌고 리셋 하고 항상 항상 실행이 된다!!

꼭 기억! **Animation**과 and 애니메이션을 sample character 잘
볼 수 있게 리셋에 리셋 **리셋**!

Self (Hello world) 의 control \Rightarrow $\frac{1}{2}$ 인 μ \Rightarrow $\frac{1}{2}$ 인 μ

\Rightarrow $CH_3COOH \xrightarrow{H^+} CH_3C(=O)OH \rightleftharpoons CH_3C(OH)=O \Rightarrow ABP-Helv$

이론적 2차원 \Rightarrow 이차원 공간의 수직, 수평 UP, DOWN
ex) 5/d

Given other Related val- \Rightarrow Cast Hero

(H₂는 반응 H₂O)

[illegible]

72

177424721

↳ Hero \Rightarrow Get velocity \Rightarrow vector length xy

↳ Set $A(2, 1)$

↳ Heron is sailing

출입과 이동

↳ out put pos 에 들어간 머리카락은 들어감

∴ 머리카락이 들어갈 공간에 들어갈 머리카락의 위치를 기록하는 것

state machine 이동 (locomotion) → out - 이동

↳ empty space + 이동 물리 엔진 작동

↳ Empty → idle

↳ 머리카락이 들어갈 공간이 없으면 (full) → idle

↳ 머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

⇒ BS-unarmed

↳ 머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

speed 증가 → Run, walk, idle (머리카락이 들어갈 공간이 없으면)

머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

BS-Hero → 머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

locomotion 에 Jump, Fall loop, Land 등

unarmed는 머리카락이 들어갈 공간이 없으면 (full) → idle

∴ 머리카락이 들어갈 공간이 없으면 (full) → idle

머리카락이 들어갈 공간이 없으면 (full) → idle

1. do Jump, 2. can land 여부 판별
 ↳ "can land" 이 true 일 때
 ↳ can land = true 일 때
 ∴ Falling 상태가 되고
 ↳ can land = false 일 때, can land = false 일 때

Jump → Falling
 unArmed → ↑
 (⇒) 이거 끝까지 → Falling and velocity $c=0$ velocity
 ↳ velocity = 0 이면 velocity = 0 이고 velocity = 0 이면 velocity = 0 이고
 ↳ velocity = 0 이면 velocity = 0 이고 velocity = 0 이면 velocity = 0 이고

1. can land 여부 판별 (1. can land 여부 판별) → can land = true 일 때
 ↳ Falling, Jump
 ↳ ! Falling
current time (ratio) ≥ 0.8

1. can land 여부 판별 (1. can land 여부 판별) → can land = true 일 때
 ↳ Falling, Jump
 ↳ ! Falling
current time (ratio) ≥ 0.8

1. can land 여부 판별 (1. can land 여부 판별) → can land = true 일 때
 ↳ Falling, Jump
 ↳ ! Falling
current time (ratio) ≥ 0.8

↳ can land = true 일 때
 ↳ can land = false 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때

can land = true 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때

↳ can land = true 일 때
 ↳ can land = false 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때

↳ can land = true 일 때
 ↳ can land = false 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때

↳ can land = true 일 때
 ↳ can land = false 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때
 ↳ can land = true 일 때
 ↳ can land = false 일 때