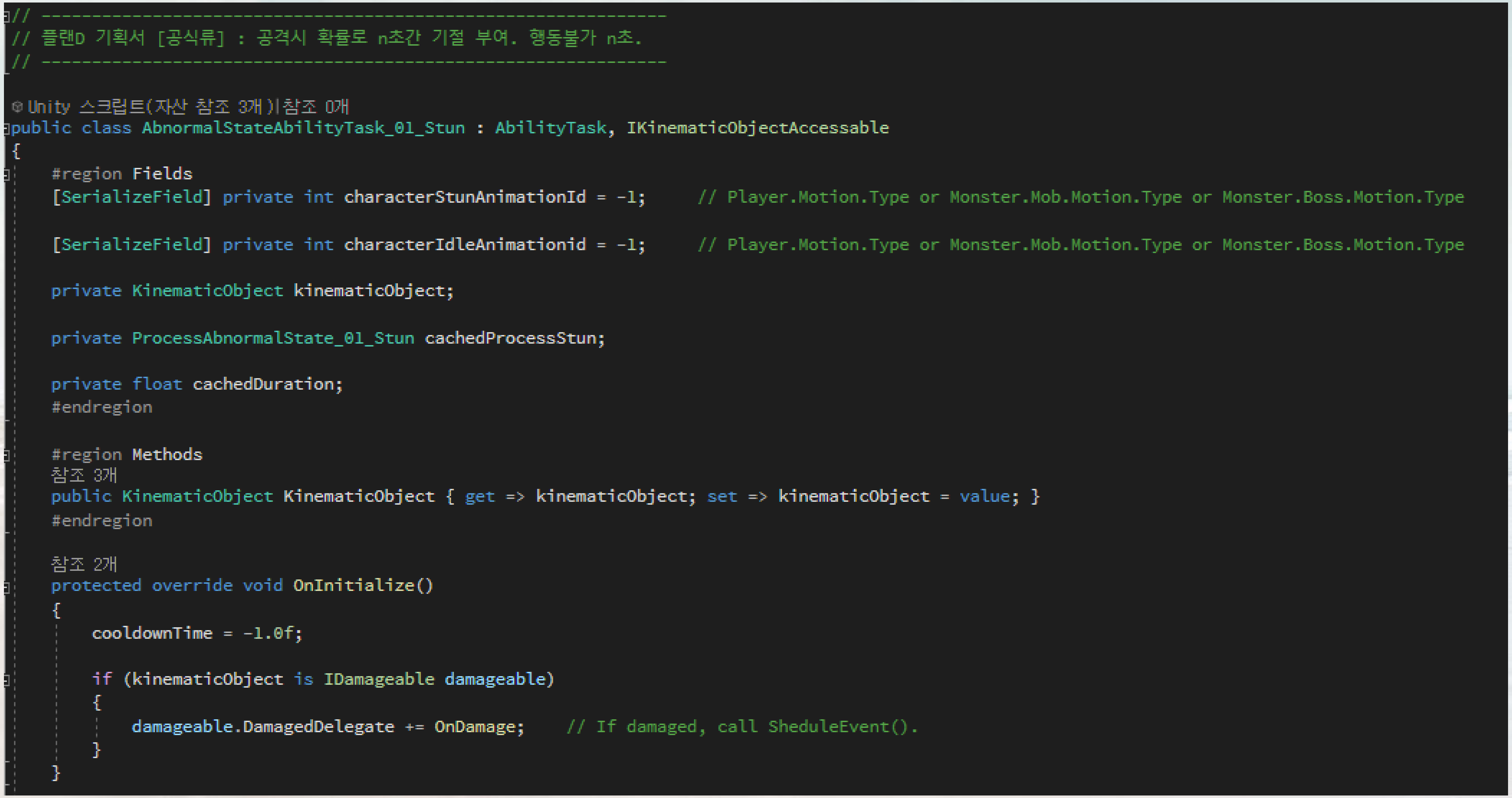
상태 이상 – Stun 구현 중심으로

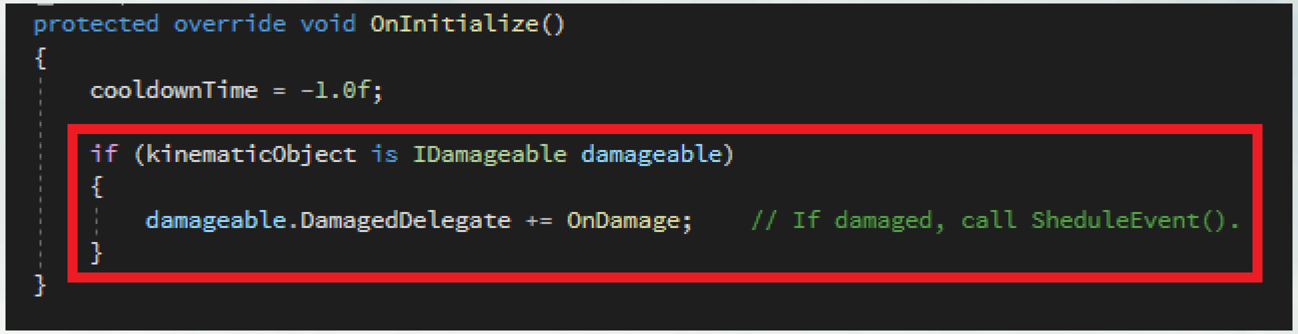
어빌리티 태스크 구현



```
// ---------------------------------------------------------------
// 플랜D 기획서 [공식류] : 공격시 확률로 n초간 기절 부여. 행동불가 n초.
// ---------------------------------------------------------------

Unity 스크립트(자산 참조 3개)|참조 0개
public class AbnormalStateAbilityTask_01_Stun : AbilityTask, IKinematicObjectAccessable
{
    #region Fields
    [SerializeField] private int characterStunAnimationId = -1;     // Player.Motion.Type or Monster.Mob.Motion.Type or Monster.Boss.Motion.Type

    [SerializeField] private int characterIdleAnimationid = -1;     // Player.Motion.Type or Monster.Mob.Motion.Type or Monster.Boss.Motion.Type

    private KinematicObject kinematicObject;

    private ProcessAbnormalState_01_Stun cachedProcessStun;

    private float cachedDuration;
    #endregion

    #region Methods
    참조 3개
    public KinematicObject KinematicObject { get => kinematicObject; set => kinematicObject = value; }
    #endregion

    참조 2개
    protected override void OnInitialize()
    {
        cooldownTime = -1.0f;

        if (kinematicObject is IDamageable damageable)
        {
            damageable.DamagedDelegate += OnDamage;    // If damaged, call SheduleEvent().
        }
    }
```

상태 이상 – Stun 구현 중심으로

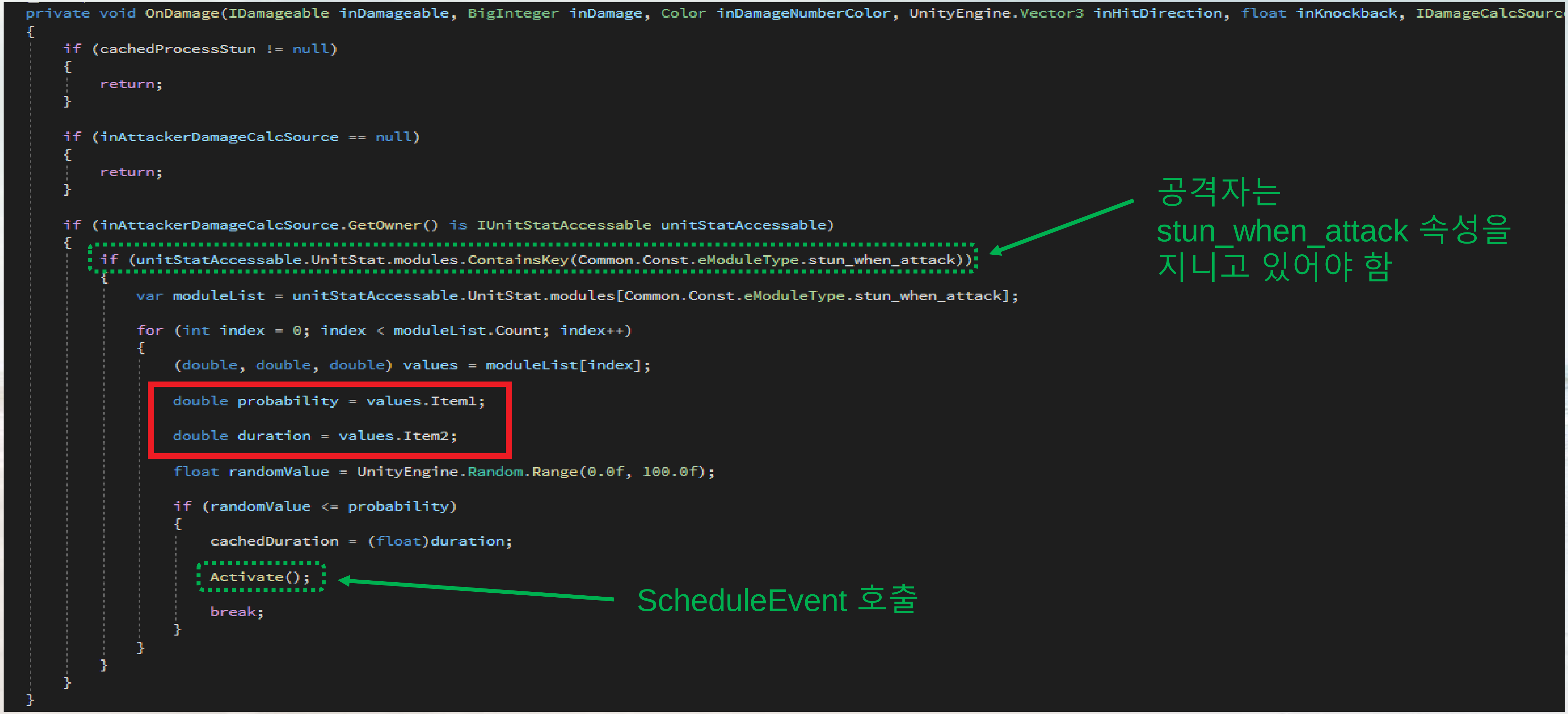
공격 받을 시 확률로 n초간 기절 부여. 행동 불가 n초.



```
protected override void OnInitialize()
{
    cooldownTime = -1.0f;

    if (kinematicObject is IDamageable damageable)
    {
        damageable.DamagedDelegate += OnDamage;    // If damaged, call SheduleEvent().
    }
}
```

상태 이상 – Stun 구현 중심으로

공격 받을 시 확률로 n초간 기절 부여. 행동 불가 n초.



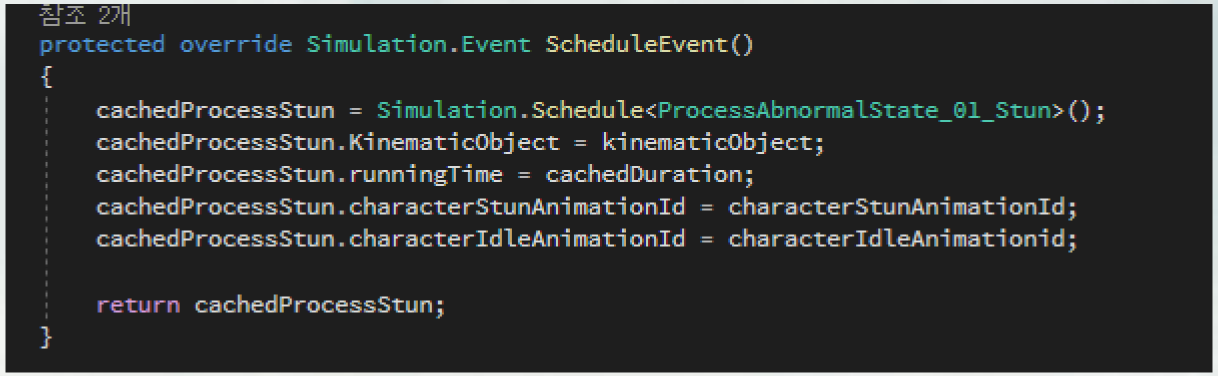
```csharp
private void OnDamage(IDamageable inDamageable, BigInteger inDamage, Color inDamageNumberColor, UnityEngine.Vector3 inHitDirection, float inKnockback, IDamageCalcSourc
{
    if (cachedProcessStun != null)
    {
        return;
    }

    if (inAttackerDamageCalcSource == null)
    {
        return;
    }

    if (inAttackerDamageCalcSource.GetOwner() is IUnitStatAccessable unitStatAccessable)
    {
        if (unitStatAccessable.UnitStat.modules.ContainsKey(Common.Const.eModuleType.stun_when_attack))
        {
            var moduleList = unitStatAccessable.UnitStat.modules[Common.Const.eModuleType.stun_when_attack];

            for (int index = 0; index < moduleList.Count; index++)
            {
                (double, double, double) values = moduleList[index];

                double probability = values.Item1;

                double duration = values.Item2;

                float randomValue = UnityEngine.Random.Range(0.0f, 100.0f);

                if (randomValue <= probability)
                {
                    cachedDuration = (float)duration;

                    Activate();

                    break;
                }
            }
        }
    }
}
```

공격자는
stun_when_attack 속성을
지니고 있어야 함

ScheduleEvent 호출

상태 이상 – Stun 구현 중심으로

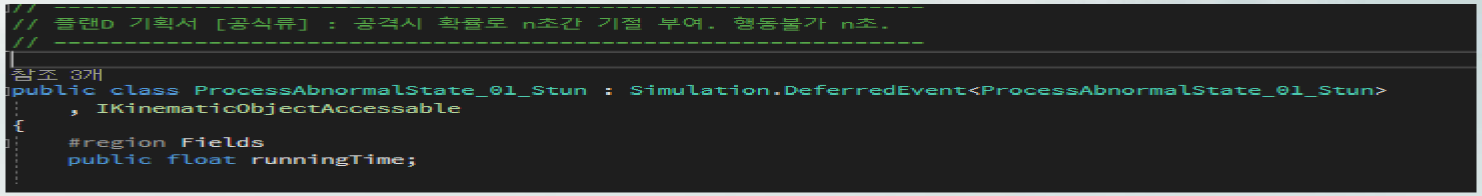
공격 받을 시 확률로 n초간 기절 부여. 행동 불가 n초



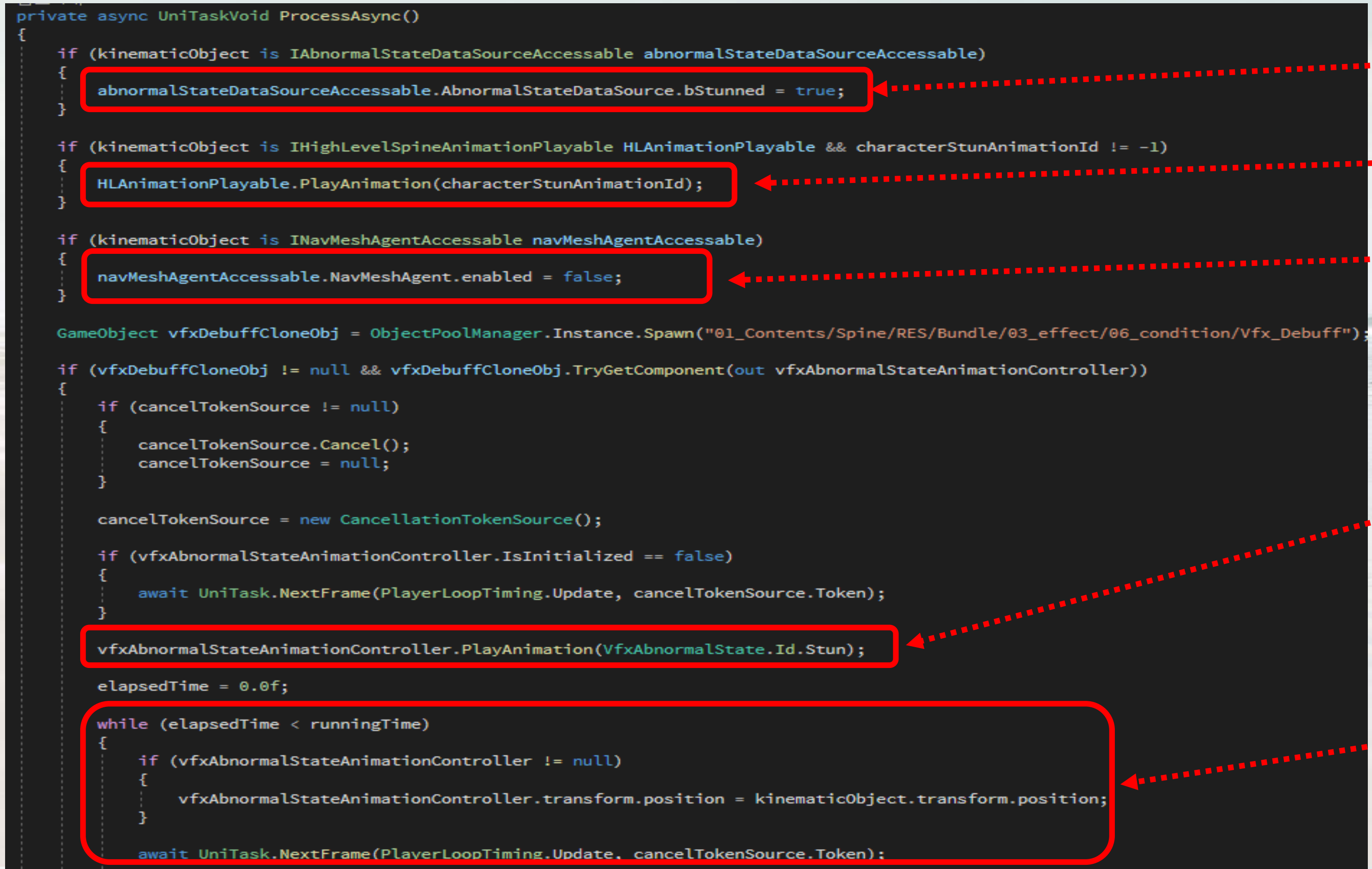
```
참조 2개
protected override Simulation.Event ScheduleEvent()
{
    cachedProcessStun = Simulation.Schedule<ProcessAbnormalState_01_Stun>();
    cachedProcessStun.KinematicObject = kinematicObject;
    cachedProcessStun.runningTime = cachedDuration;
    cachedProcessStun.characterStunAnimationId = characterStunAnimationId;
    cachedProcessStun.characterIdleAnimationId = characterIdleAnimationid;

    return cachedProcessStun;
}
```

상태 이상 – Stun 구현 중심으로

공격 받을 시 확률로 n초간 기절 부여. 행동 불가 n초.



```
// ------------------------------------------------------------------
// 플랜D 기획서 [공식류] : 공격시 확률로 n초간 기절 부여. 행동불가 n초.
// ------------------------------------------------------------------

참조 3개
public class ProcessAbnormalState_01_Stun : Simulation.DeferredEvent<ProcessAbnormalState_01_Stun>
    , IKinematicObjectAccessable
{
    #region Fields
    public float runningTime;
```



```
private async UniTaskVoid ProcessAsync()
{
    if (kinematicObject is IAbnormalStateDataSourceAccessable abnormalStateDataSourceAccessable)
    {
        abnormalStateDataSourceAccessable.AbnormalStateDataSource.bStunned = true;
    }

    if (kinematicObject is IHighLevelSpineAnimationPlayable HLAnimationPlayable && characterStunAnimationId != -1)
    {
        HLAnimationPlayable.PlayAnimation(characterStunAnimationId);
    }

    if (kinematicObject is INavMeshAgentAccessable navMeshAgentAccessable)
    {
        navMeshAgentAccessable.NavMeshAgent.enabled = false;
    }

    GameObject vfxDebuffCloneObj = ObjectPoolManager.Instance.Spawn("01_Contents/Spine/RES/Bundle/03_effect/06_condition/Vfx_Debuff");

    if (vfxDebuffCloneObj != null && vfxDebuffCloneObj.TryGetComponent(out vfxAbnormalStateAnimationController))
    {
        if (cancelTokenSource != null)
        {
            cancelTokenSource.Cancel();
            cancelTokenSource = null;
        }

        cancelTokenSource = new CancellationTokenSource();

        if (vfxAbnormalStateAnimationController.IsInitialized == false)
        {
            await UniTask.NextFrame(PlayerLoopTiming.Update, cancelTokenSource.Token);
        }

        vfxAbnormalStateAnimationController.PlayAnimation(VfxAbnormalState.Id.Stun);

        elapsedTime = 0.0f;

        while (elapsedTime < runningTime)
        {
            if (vfxAbnormalStateAnimationController != null)
            {
                vfxAbnormalStateAnimationController.transform.position = kinematicObject.transform.position;
            }

            await UniTask.NextFrame(PlayerLoopTiming.Update, cancelTokenSource.Token);
```

Stun 플래그 설정. Behavior Tree 조건 노드(attack, move …) 에서 사용

캐릭터 Stun 애니메이션 호출

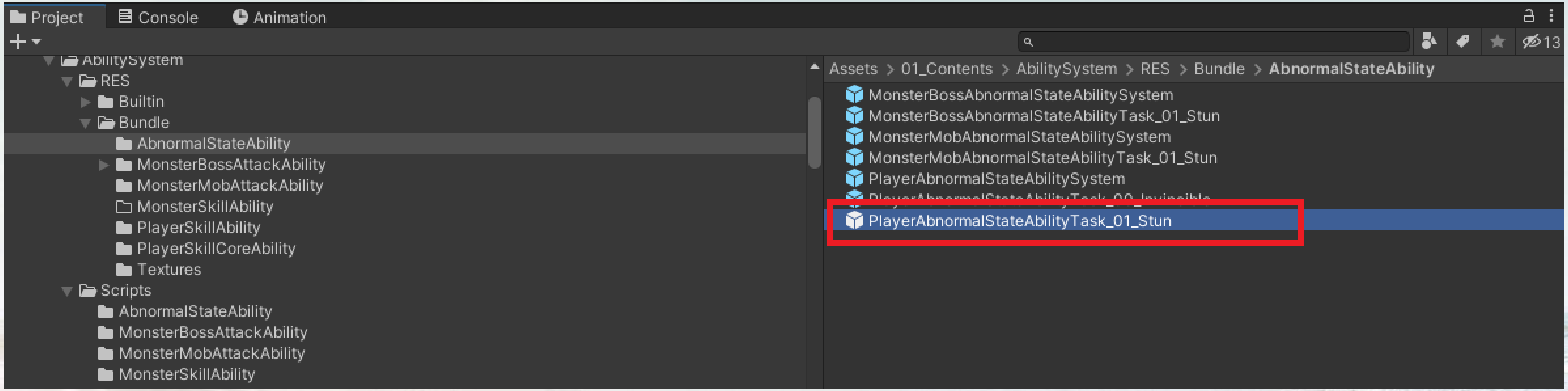
혹시 모르니 NavMeshAgent 도 끔

머리 위에 빙글빙글 이펙트 출력

빙글빙글 이펙트가 머리 위에 계속 있도록

상태 이상 – Stun 구현 중심으로

Stun 어빌리티 태스크 프리팹 생성



Project | Console | Animation

13

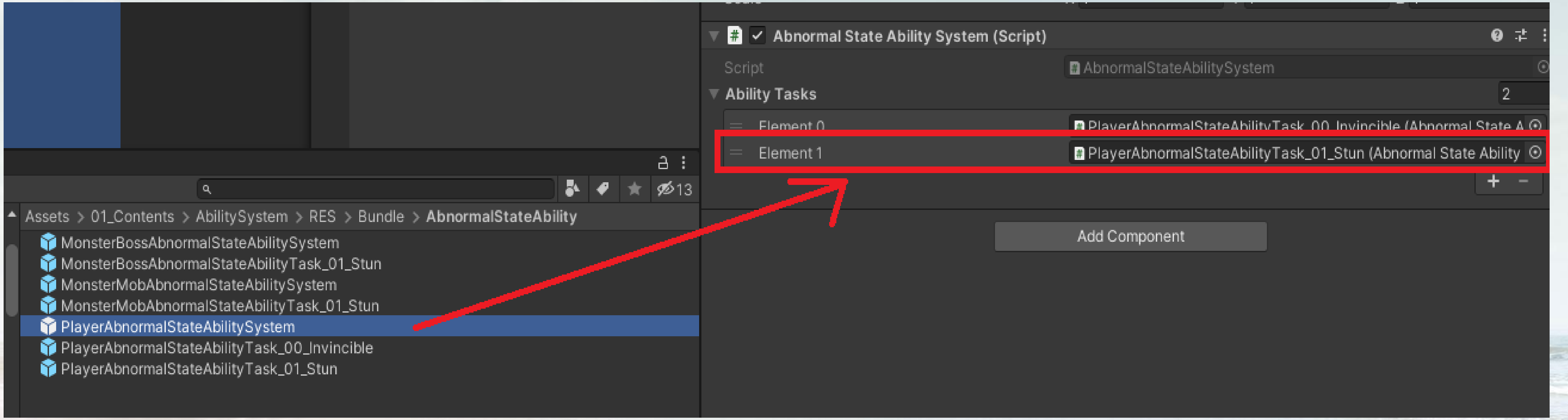
- AbilitySystem
  - RES
    - Builtin
    - Bundle
      - AbnormalStateAbility
      - MonsterBossAttackAbility
      - MonsterMobAttackAbility
      - MonsterSkillAbility
      - PlayerSkillAbility
      - PlayerSkillCoreAbility
      - Textures
  - Scripts
    - AbnormalStateAbility
    - MonsterBossAttackAbility
    - MonsterMobAttackAbility
    - MonsterSkillAbility

Assets > 01_Contents > AbilitySystem > RES > Bundle > **AbnormalStateAbility**

- MonsterBossAbnormalStateAbilitySystem
- MonsterBossAbnormalStateAbilityTask_01_Stun
- MonsterMobAbnormalStateAbilitySystem
- MonsterMobAbnormalStateAbilityTask_01_Stun
- PlayerAbnormalStateAbilitySystem
- PlayerAbnormalStateAbilityTask_00_Invincible
- PlayerAbnormalStateAbilityTask_01_Stun

상태 이상 – Stun 구현 중심으로

PlayerAbnormalStateAbilitySystem 에 등록



Assets > 01_Contents > AbilitySystem > RES > Bundle > AbnormalStateAbility

- MonsterBossAbnormalStateAbilitySystem
- MonsterBossAbnormalStateAbilityTask_01_Stun
- MonsterMobAbnormalStateAbilitySystem
- MonsterMobAbnormalStateAbilityTask_01_Stun
- PlayerAbnormalStateAbilitySystem
- PlayerAbnormalStateAbilityTask_00_Invincible
- PlayerAbnormalStateAbilityTask_01_Stun

Abnormal State Ability System (Script)

Script	AbnormalStateAbilitySystem
Ability Tasks	2
Element 0	PlayerAbnormalStateAbilityTask_00_Invincible (Abnormal State A
Element 1	PlayerAbnormalStateAbilityTask_01_Stun (Abnormal State Ability