

```
using UnityEngine;
using System.Collections;

// Player class
public class CPlayer : MonoBehaviour {

    // Gun prefab
    protected Object gunPrefab;
    // Gun game object
    protected GameObject gun;

    protected Animator animator;

    // Shoot relative position
    protected Vector3 shootRelativePosition;

    // Shooting flag
    protected bool isShooting = false;

    // Shoot delay time
    public float shootDelayTime;

    // Player energy
    public float hp;

    // Game level manager reference
    public GameObject gameLevelManager;

    // Energy bar game reference
    public GameObject energyBar;

    void Awake()
    {
        animator = GetComponentInChildren<Animator>();
    }

    void Start () {

        InitPlayer();
    }

    // Player initialize
    void InitPlayer()
    {
        gunPrefab = (Object)Resources.Load("Prefabs/Guns/Gun");

        gun = (GameObject)Instantiate(gunPrefab, transform.position ,
            Quaternion.identity);

        gun.transform.parent = gameObject.transform;
    }

    // Attack area touch event notification
    public void NotiAttackTouchEvent(Vector3 touch_position)
    {

```

```

    if (isShooting) return;

    StartCoroutine("ShootStartCroutine", touch_position);
}

// 발포 코루틴
IEnumerator ShootStartCroutine(Vector3 touch_position)
{
    // 터치 영역과 플레이어를 기준으로한 발포 위치를 구함
    shootRelativePosition = touch_position - transform.position;

    // 플레이어의 발포 애니메이션을 수행함
    PlayShootAnimation();

    // 발포 상태를 발포 중으로 변경함
    isShooting = true;

    // 발포 대기 간격 만큼 대기함
    yield return new WaitForSeconds(shootDelayTime);

    // 발포 상태를 발포 가능 상태로 변경함
    isShooting = false;
}

// 플레이어의 애니메이션을 수행함
public void PlayShootAnimation()
{
    // 발포 위치와 플레이어간의 발사 각도를 측정함
    float angle = Mathf.Atan2(shootRelativePosition.y, shootRelativePosition.x) *
        * Mathf.Rad2Deg;
    Debug.Log("발사 각도 : " + angle);

    // 발사 각도에 맞는 애니메이션을 수행함
    if (angle <= -5f && angle > -28f)
    {
        animator.SetTrigger("Fire1");
    }
    else if (angle <= -28f && angle > -40f)
    {
        animator.SetTrigger("Fire2");
    }
    else if (angle <= -40f && angle > -65f)
    {
        animator.SetTrigger("Fire3");
    }
    else if (angle <= -65f)
    {
        animator.SetTrigger("Fire4");
    }
}

// 발포 애니메이션 수행 중 발포 애니메이션 이벤트를 받음
public void ShootAnimationEvent()
{
    // 총에게 발포를 요청함
    ShootGun();
}

```

```
}

// 총에게 발포를 요청함
public void ShootGun()
{
    Vector2 shootVec = new Vector2(shootRelativePosition.x,
        shootRelativePosition.y);

    // 총에게 발포를 지시함
    gun.SendMessage("Fire", shootVec);
}

public void HitPlayer()
{
    // 게임 시작을 한게 아니면 무시함
    if (!CGameInfo.IS_GAME_START) return;

    // 플레이어의 에너지가 완전히 소모되면
    if (hp <= 0)
    {
        hp = 0f;

        // 게임을 종료함
        CGameInfo.IS_GAME_START = false;

        // 게임 종료 팝업을 띄움
        gameLevelManager.SendMessage("GameEnd");

        return;
    }

    hp -= 5f;

    energyBar.SendMessage("DecreaseEnergyBar", hp * 0.01f);
}
}
```