

○ ○ ○

```
/// <summary>
/// Updates stamina UI based on attack cooldown.
/// </summary>
private void UpdateUI()
{
    mainUI.currentStamina = cooldown_doAttack * 100;
}

/// <summary>
/// Handles cooldown timer for being able to attack again.
/// </summary>
private void HandleGetAttackCooldown()
{
    if (getAttack)
        cooldownGetAttack -= Time.deltaTime;

    if (cooldownGetAttack < 0)
        getAttack = false;
}

/// <summary>
/// Checks for player input to perform an attack.
/// </summary>
private void HandleAttackInput()
{
    if (cooldown_doAttack < 0 && Input.GetKeyDown(KeyCode.Space))
    {
        animator.SetTrigger("attack"); // Play attack animation
        attack = true;
        getAttack = true;

        // Reset cooldown timers
        cooldown = saveCooldown;
        cooldown_doAttack = cooldown_saveDoAttack;
        cooldownGetAttack = save_cooldownGetAttack;

        // Play sound effect
        GameObject soundEffect = Instantiate(soundEffect_dash);
        Destroy(soundEffect, 1f);
    }
    else
    {
        cooldown_doAttack -= Time.deltaTime;
    }
}

/// <summary>
/// Manages how long the attack state stays active.
/// </summary>
private void HandleAttackState()
{
    if (cooldown > 0)
        cooldown -= Time.deltaTime;
    else
        attack = false;
}

/// <summary>
/// Adjusts player movement speed during attack.
/// </summary>
private void AdjustMovementSpeed()
{
    if (attack)
        playerController.speed = saveSpeed * 1.125f; // 12.5% speed boost
    else
        playerController.speed = saveSpeed;
}
```