```
using UnityEngine;
using System.Collections;

// Monster class
public class CMonster : MonoBehaviour {

    // Monster State
    protected enum STATE { WALK, ATTACK, DIE }
    protected STATE monsterState = STATE.WALK;

    // Hit sound audio clip
    public AudioClip hitAudioClip;

    // Monster speed
    public float speed;

    protected Animator animator;
    protected SpriteRenderer spriteRenderer;

    // Player reference
    protected Transform player;

    // Monster hit count
    protected int hitCount = 0;
    // Monster hit effect position
    public Transform hitPoint;
    // Monster attack effect position
    public Transform attackHitPoint;
    // Monster hit effect prefab
    protected Object hitPrefab;
    // Monster attack hit effect prefab
    protected Object attackHitPrefab;

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

        hitPrefab = (Object)Resources.Load("Prefabs/Effects/Hit");
        attackHitPrefab = (Object)Resources.Load("Prefabs/Effects/AttackHit");
    }

    void Start () {
        InitMonster();
    }

    void Update () {
        OrderZIndex();
    }

    // Monster initialize
    public virtual void InitMonster()
    {
        player = GameObject.Find("Player").transform;

        StartMove();
    }

    // Monster move start
    public void StartMove()
    {
        Walk();
    }

    // Monster attack
    public void Attack()
    {
        monsterState = STATE.ATTACK;

        GetComponent<Rigidbody2D>().velocity = Vector2.zero;

        animator.SetTrigger("Attack");
    }

    // Monster walk
    public void Walk()
    {
        monsterState = STATE.WALK;

        AnimatorStateInfo ani_state_info = animator.GetCurrentAnimatorStateInfo(0);

        if (!ani_state_info.IsName("Walk"))
            animator.SetTrigger("Walk");

        GetComponent<Rigidbody2D>().AddForce(Vector3.left * speed);



    }

    // Monster hit
    public void Hit()
    {
        GameObject hit_effect = (GameObject)Instantiate(hitPrefab,                ↩
          hitPoint.position, Quaternion.identity);
        Destroy(hit_effect, 0.3f);

        if (hitCount > 2)
        {
            DoDestroy(1f);
            return;
        }

        hitCount++;

        CGameSound.PlayGameSound(hitAudioClip, transform.position);

        GetComponent<Rigidbody2D>().velocity = Vector2.zero;

        animator.SetTrigger("Hit");
    }
    // Monster "OnTrigger" event
    void OnTriggerEnter2D(Collider2D collider)
    {
        if (collider.tag == "DefenceArea")
        {
            Attack();
        }
        else if (collider.tag == "Bullet")
        {
            collider.SendMessage("DoDestroy", 0f);

            Hit();
        }
    }

    // Hit effect animation complete
    public void HitAnimationComplete()
    {
        if (monsterState == STATE.WALK)
            Walk();
        else
            Attack();
    }

    // Attack animation start complete
    public void AttackAnimationStartComplete()
    {
        GameObject hit_effect = (GameObject)Instantiate(attackHitPrefab,
          attackHitPoint.position, Quaternion.identity);
        Destroy(hit_effect, 0.3f);

        player.SendMessage("HitPlayer");
    }

    // Attack animation complete
    public void AttackAnimationComplete()
    {
        player.SendMessage("HitPlayer");
    }

    // Monster Death
    public void DoDestroy(float delay_time = 0f)
    {
        GetComponent<BoxCollider2D>().enabled = false;

        CGameInfo.GAME_SCORE += 1;

        GetComponent<Rigidbody2D>().velocity = Vector2.zero;

        animator.SetTrigger("Die");

        Destroy(gameObject, delay_time);
    }

    // Monster order z index
    public void OrderZIndex()
    {

        float y = transform.position.y * 100f;

        if (y <= 0)
        {
            y = Mathf.Abs(y);
        }
        else
        {
            y = -y;
        }

        spriteRenderer.sortingOrder = (int)y;

    }

}
```