using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Enemy : MonoBehaviour {

 public Animator m\_Animator;

 public SpriteRenderer SR;

 public BoxCollider2D box;

 public GameObject HTurnSFX;

 public GameObject CTurn1SFX;

 public GameObject CTurn2SFX;

 public GameObject CTurn3SFX;

 public GameObject CTurn4SFX;

 public GameObject CTurn5SFX;

 public float colliderX;

 //Next six lines from EnemyPool

 public bool isPlayerClose;

 public bool canFlip;

 public int number;

 public int who;

 public int odds = 0;

 public int oddsWait = 0;

 public int CTOdds = 0;

 public bool attacking;

 public bool idle;

 public bool withinRange;

 public bool canHands;

 public float offsetTimeCount;

 public float timeCount;

 public float anticipationSpeed;

 void Start()

 {

 m\_Animator = gameObject.GetComponent<Animator> ();

 SR = gameObject.GetComponent<SpriteRenderer> ();

 box = gameObject.GetComponent<BoxCollider2D> ();

 //Next three lines from EnemyPool

 idle = false;

 attacking = false;

 withinRange = false;

 anticipationSpeed = -0f;

 colliderX = box.offset.x;

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanHTurn");

 m\_Animator.ResetTrigger ("Attack");

 m\_Animator.ResetTrigger ("Idle");

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 }

 void Update ()

 {

 timeCount += Time.deltaTime;

 if (GameControl.instance.gameOver == true)

 {

 gameObject.SetActive (false);

 }

 if (EnemyPool.instance.slowDown == true)

 {

 gameObject.GetComponent<ScrollingObject> ().useAnticipationSpeed = true;

 }

 else if(EnemyPool.instance.slowDown == false)

 {

 gameObject.GetComponent<ScrollingObject> ().useAnticipationSpeed = false;

 }

 if (timeCount >= 3)

 {

 oddsWait = Random.Range (1, 9);

 offsetTimeCount += Time.deltaTime;

 }

 else if (timeCount < 3)

 {

 oddsWait = 0;

 }

 if (oddsWait >= 1 && oddsWait <= 3 && offsetTimeCount >= 0 && attacking == false)

 {

 attackOdds ();

 timeCount = 0;

 offsetTimeCount = 0;

 }

 else if (oddsWait > 3 && oddsWait <= 6 && offsetTimeCount >= 0.5f && attacking == false)

 {

 attackOdds ();

 timeCount = 0;

 offsetTimeCount = 0;

 }

 else if (oddsWait > 6 && oddsWait <= 9 && offsetTimeCount >= 1 && attacking == false)

 {

 attackOdds ();

 timeCount = 0;

 offsetTimeCount = 0;

 }

 if (idle == true)

 {

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.ResetTrigger ("CanHTurn");

 m\_Animator.ResetTrigger ("Attack");

 m\_Animator.SetTrigger("Idle");

 idle = false;

 }

 if (transform.position.x > 0)

 {

 SR.flipX = true;

 gameObject.GetComponentInChildren<DetectPlayer> ().ASR.flipX = true;

 gameObject.GetComponentInChildren<Attack> ().SR2.flipX = true;

 gameObject.GetComponent<BoxCollider2D> ().offset = new Vector2(-colliderX, 0);

 }

 if (transform.position.x < 0 && SR.flipX == true)

 {

 SR.flipX = false;

 gameObject.GetComponentInChildren<DetectPlayer> ().ASR.flipX = false;

 gameObject.GetComponentInChildren<Attack> ().SR2.flipX = false;

 gameObject.GetComponent<BoxCollider2D> ().offset = new Vector2(colliderX, 0);

 }

 }

 private void OnTriggerEnter2D (Collider2D other)

 {

 if (other.GetComponent<PlayerMovement> () != null && other.gameObject.tag == "Player")

 {

 isPlayerClose = true;

 }

 if (other.gameObject.tag == "Valid")

 {

 withinRange = true;

 }

 }

 private void OnTriggerExit2D(Collider2D other)

 {

 if (other.gameObject.tag == "Player")

 {

 isPlayerClose = false;

 }

 if (other.gameObject.tag == "Valid")

 {

 withinRange = false;

 }

 }

 //Next two functions from EnemyPool

 void attackOdds ()

 {

 odds = Random.Range (1,18);

 if (odds >= 1 && odds <= 3)

 {

 //Play harmless animation & set animation rate / timevariable

 m\_Animator.ResetTrigger ("Attack");

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.ResetTrigger ("Idle");

 m\_Animator.SetTrigger("CanHTurn");

 if (GlobalControl.Instance.SFXOn == true)

 {

 HTurnSFX.SetActive (true);

 }

 }

 else if (odds > 3 && odds <= 6)

 {

 //Play creepy animation & set animation rate / timevariable

 m\_Animator.ResetTrigger ("Attack");

 m\_Animator.ResetTrigger ("CanHTurn");

 m\_Animator.ResetTrigger ("Idle");

 //m\_Animator.SetTrigger("CanCTurn");

 CTurn();

 HTurnSFX.SetActive (false);

 }

 else if (odds > 6 && odds <= 15 && isPlayerClose == true)

 {

 GameControl.instance.valid = true;

 GameControl.instance.player.GetComponent<SpriteRenderer>().color = Color.red;

 GameControl.instance.playerSpeed = GameControl.instance.whichFeathers;

 gameObject.GetComponent<ScrollingObject> ().useAnticipationSpeed = true;

 EnemyPool.instance.slowDown = true;

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.ResetTrigger ("CanHTurn");

 m\_Animator.ResetTrigger ("Idle");

 m\_Animator.SetTrigger("Attack");

 who = number;

 attack ();

 HTurnSFX.SetActive (false);

 }

 else if (odds > 15)

 {

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.ResetTrigger ("CanHTurn");

 m\_Animator.ResetTrigger ("Idle");

 m\_Animator.ResetTrigger ("Attack");

 HTurnSFX.SetActive (false);

 }

 }

 void CTurn ()

 {

 CTOdds = Random.Range (1,5);

 if (CTOdds == 1)

 {

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.SetTrigger ("CanCTurn");

 if (isPlayerClose == true)

 {

 if (GlobalControl.Instance.SFXOn == true)

 {

 CTurn1SFX.SetActive (true);

 }

 CTurn2SFX.SetActive (false);

 CTurn3SFX.SetActive (false);

 CTurn4SFX.SetActive (false);

 CTurn5SFX.SetActive (false);

 }

 }

 if (CTOdds == 2)

 {

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.SetTrigger ("CanCTurn2");

 if (isPlayerClose == true)

 {

 if (GlobalControl.Instance.SFXOn == true)

 {

 CTurn2SFX.SetActive (true);

 }

 CTurn1SFX.SetActive (false);

 CTurn3SFX.SetActive (false);

 CTurn4SFX.SetActive (false);

 CTurn5SFX.SetActive (false);

 }

 }

 if (CTOdds == 3)

 {

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.SetTrigger ("CanCTurn3");

 if (isPlayerClose == true)

 {

 if (GlobalControl.Instance.SFXOn == true)

 {

 CTurn3SFX.SetActive (true);

 }

 CTurn2SFX.SetActive (false);

 CTurn1SFX.SetActive (false);

 CTurn4SFX.SetActive (false);

 CTurn5SFX.SetActive (false);

 }

 }

 if (CTOdds == 4)

 {

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.ResetTrigger ("CanCTurn5");

 m\_Animator.SetTrigger ("CanCTurn4");

 if (isPlayerClose == true)

 {

 if (GlobalControl.Instance.SFXOn == true)

 {

 CTurn4SFX.SetActive (true);

 }

 CTurn2SFX.SetActive (false);

 CTurn3SFX.SetActive (false);

 CTurn1SFX.SetActive (false);

 CTurn5SFX.SetActive (false);

 }

 }

 if (CTOdds == 5)

 {

 m\_Animator.ResetTrigger ("CanCTurn2");

 m\_Animator.ResetTrigger ("CanCTurn3");

 m\_Animator.ResetTrigger ("CanCTurn4");

 m\_Animator.ResetTrigger ("CanCTurn");

 m\_Animator.SetTrigger ("CanCTurn5");

 if (isPlayerClose == true)

 {

 if (GlobalControl.Instance.SFXOn == true)

 {

 CTurn5SFX.SetActive (true);

 }

 CTurn2SFX.SetActive (false);

 CTurn3SFX.SetActive (false);

 CTurn4SFX.SetActive (false);

 CTurn1SFX.SetActive (false);

 }

 }

 }

 void attack ()

 {

 attacking = true;

 PlayerMovement.instance.bigSplash.SetActive (false);

 Time.timeScale = 1f;

 }

 void anticipation ()

 {

 Time.timeScale = 0.5f;

 }

}