**KAYNAK KOD**

**Tasi.cs :** sürükle bırak işlemini yapan script dosyamız.

public class tasi : MonoBehaviour

{

Camera kamera;

Vector2 baslangic\_pos;

GameObject[] kutu\_dizisi;

Yonetim yonet;

public int toplam\_parca;

private void OnMouseDrag()

{

Vector3 pozisyon = kamera.ScreenToWorldPoint(Input.mousePosition);

pozisyon.z = 0;

transform.position = pozisyon;

}

void Start()

{

kamera = GameObject.Find("Main Camera").GetComponent<Camera>();

baslangic\_pos = transform.position;

kutu\_dizisi = GameObject.FindGameObjectsWithTag("kutu");

yonet = GameObject.Find("yonetim").GetComponent<Yonetim>();

toplam\_parca = kutu\_dizisi.Length;

yonet.hedef\_puan= toplam\_parca\*100;

yonet.hedef\_puan\_txt.text = (toplam\_parca \* 100).ToString();

Debug.Log("toplam parça=="+toplam\_parca);

}

// Update is called once per frame

void Update()

{

if (Input.GetMouseButtonUp(0))

{

foreach (GameObject kutu in kutu\_dizisi)

{

if (kutu.name==gameObject.name)

{

float mesafe = Vector3.Distance(kutu.transform.position, transform.position);

if (mesafe<=1)

{

transform.position = kutu.transform.position;

yonet.puan\_artir(100);

this.enabled = false;

}

else if(mesafe>1)

{

transform.position = baslangic\_pos;

yonet.Can\_kontrol();//düzenle

}

}

}

}

}

//private void OnCollisionEnter2D(Collision2D collision)

//{

// if (collision.gameObject.CompareTag(""))

// {

// }

//}

}

**SG.cs:** giriş sahnesi için sahneler arası geçişi sağlayan script dosyamız.

using UnityEngine.SceneManagement;

public class SG : MonoBehaviour

{

// Start is called before the first frame update

void Start()

{

}

// Update is called once per frame

void Update()

{

if (Input.GetKeyDown(KeyCode.Space ))

{

SceneManager.LoadScene(1);

}

}

public void sahnegec()

{

SceneManager.LoadScene(1);

}

public void cikis()

{

Application.Quit();

}

}

**SoundManager.cs :**  sahnedeki ses dosyalarının yönetim işlemini yapan script dosyamız.

public class SoundManager : MonoBehaviour

{

public static SoundManager instance;

[SerializeField]

private AudioSource yerlestirmeSes, kaybetmeSes, kazanmaSes, clickSes, bilgiSes, puanSes,zamankritikSes, yahooSes;

private void Awake()

{

if (instance==null)

{

instance = this;

}

}

#region ses methodları

public void yahooses()

{

zamankritikSes.Stop();

puanSes.Stop();

bilgiSes.Stop();

clickSes.Stop();

yerlestirmeSes.Stop();

kaybetmeSes.Stop();

kazanmaSes.Stop();

yahooSes.Play();

}

public void zamankritikses()

{

yahooSes.Stop();

puanSes.Stop();

bilgiSes.Stop();

clickSes.Stop();

yerlestirmeSes.Stop();

kaybetmeSes.Stop();

kazanmaSes.Stop();

zamankritikSes.Play();

}

public void puanses()

{

puanSes.Play();

}

public void bilgises()

{

yahooSes.Stop();

zamankritikSes.Stop();

puanSes.Stop();

clickSes.Stop();

yerlestirmeSes.Stop();

kaybetmeSes.Stop();

kazanmaSes.Stop();

bilgiSes.Play();

}

public void clickses() {

yahooSes.Stop();

zamankritikSes.Stop();

puanSes.Stop();

bilgiSes.Stop();

yerlestirmeSes.Stop();

kaybetmeSes.Stop();

kazanmaSes.Stop();

clickSes.Play();

}

public void yerlestirmeses()

{

yahooSes.Stop();

zamankritikSes.Stop();

puanSes.Stop();

bilgiSes.Stop();

clickSes.Stop();

kaybetmeSes.Stop();

kazanmaSes.Stop();

yerlestirmeSes.Play();

}

public void kaybetmeses()

{

yahooSes.Stop();

zamankritikSes.Stop();

puanSes.Stop();

bilgiSes.Stop();

clickSes.Stop();

yerlestirmeSes.Stop();

kazanmaSes.Stop();

kaybetmeSes.Play();

}

public void kazanmases()

{

yahooSes.Stop();

zamankritikSes.Stop();

puanSes.Stop();

bilgiSes.Stop();

clickSes.Stop();

yerlestirmeSes.Stop();

kaybetmeSes.Stop();

kazanmaSes.Play();

}

#endregion ses methodları

}

**Yonetim.cs :** sahne içerisinde genel yönetimi, ses, animasyon, panel işemleri, geri sayım, puanlama gibi işlemleri bu script dosyası üzerinden kontrol ettik.

using TMPro;

using UnityEngine;

using UnityEngine.SceneManagement;

public class Yonetim : MonoBehaviour

{

int puan = 0;

public int hedef\_puan = 500;

int saniye = 60;

public AudioSource arkaplanses;

public AudioSource puansesi;

public AudioSource kritikzamansesi;

public AudioSource kazandinses;

public AudioSource kaybettinses;

int yerlesen\_parca=0;

tasi tasima\_obj = new tasi();

public int toplam\_parca = 20;

public GameObject kaybet\_pnl;

public GameObject kazandin\_pnl;

public GameObject bilgi\_pnl;

public bool panel\_acik = false;

public TextMeshProUGUI saniye\_txt;

public TextMeshProUGUI puan\_txt;

public TextMeshProUGUI hedef\_puan\_txt;

int can;//3 yanlýþ hakkýn var

int sure = 60;// oyun süresi

// Start is called before the first frame update

void Start()

{

//hedef\_puan = tasima\_obj.toplam\_parca;

saniye\_txt.text = saniye.ToString();

puan\_txt.text = puan.ToString();

hedef\_puan\_txt.text = hedef\_puan.ToString();

InvokeRepeating("saniye\_azalt", 0.0f, 1f);

can = 3;

}

#region methotlar

public void sayi\_artir()

{

yerlesen\_parca++;//tüm kutular yerleþtirildiyse

if (yerlesen\_parca==toplam\_parca)

{

Debug.Log("sonraki sahneye geç !!!");

}

}

public void Can\_kontrol()

{

can++;//can kontrol

if (can==0)

{

Debug.Log(can+"kaybetttin pnl!!!");

}

else

{

Debug.Log(can + "kaybetttin pnl!!!");

}

}

//saniye azaltma methodu

void saniye\_azalt()

{

if (!panel\_acik)

{

saniye--;

saniye\_txt.text = saniye.ToString();

if (saniye == 10)

{

kritikzamansesi.Play();

}

if (saniye <= 0)

{

kaybettin();

}

}

}

public void puan\_artir(int deger) {

puan+=deger;

puan\_txt.text = puan.ToString();

puansesi.Play();

if (puan>=hedef\_puan)

{

arkaplanses.Stop();

Time.timeScale = 1.0f;

kazandin\_pnl.SetActive(true);

panel\_acik = true;

kazandinses.Play();

Debug.Log("kazandın");

}

}

void kaybettin() {

arkaplanses.Stop();

kaybet\_pnl.SetActive(true);

panel\_acik = true;

kaybettinses.Play();

Time.timeScale = 1.0f;

}

public void tekrar\_oyna()

{

Time.timeScale = 1.0f;//sahne zamanı durdur

SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);//aktif sahnenin indexini alır

}

public void sonraki\_sahne()

{

Time.timeScale = 1.0f;//sahne zamanı durdur

SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex+1);//aktif sahnenin indexini alır

}

public void ilk\_sahne()

{

Time.timeScale = 1.0f;//sahne zamanı durdur

SceneManager.LoadScene(0);//aktif sahnenin indexini alır

}

public void cikis()

{

Application.Quit();

}

public void devam\_et()

{

Time.timeScale = 1.0f;

kaybet\_pnl.SetActive(false);//durdur paneli gizle

panel\_acik = false;

}

public void durdur\_btn()

{

Time.timeScale = 0.0f;

bilgi\_pnl.SetActive(true);

panel\_acik = true;

}

public void bilgi\_panel\_gizle()

{

Time.timeScale = 1;//tüm hareketi aç

bilgi\_pnl.SetActive(false);

panel\_acik = false;

if (saniye <= 10)

{

kritikzamansesi.Play();

}

}

public void bilgi\_panel\_goster()

{

Time.timeScale = 1;//tüm hareketi baslat

if (saniye <= 10)

{

kritikzamansesi.Pause();

}

bilgi\_pnl.SetActive(true);

panel\_acik = true;

}

#endregion methotlar

// Update is called once per frame

void Update()

{

}

}