UNITY Örnek Sorular

1. Texture2D tipinde özel (private) bir değişken tanımlayın ve adını \_icon yapın. Bu değişkene, Unity Inspector üzerinden bir ikon texture atanabilsin. OnGUI() metodunu kullanarak ekranın sol üst köşesine 50X50 piksel boşluk bırakarak bir ikon çizdirin. Çizilen ikonun boyutu 64x64 piksel olacak şekilde ayarlanmalıdır.
2. Bir MonoBehaviour sınıfı oluşturun ve adını ScoreDisplay olarak belirleyin. OnGUI() metodunu kullanarak aşağıdaki metinleri ekrana çizdirin: "Score: 1500" metnini ekranın sol üst köşesinden 30 piksel sağa ve 20 piksel aşağıya yerleştirin. "Level: 3" metnini ekranın sağ üst köşesinden 30 piksel sola ve 20 piksel aşağıya yerleştirin. "Time: 45" metnini ekranın tam ortasına yerleştirin. Metnin genişliği 60 piksel, yüksekliği 20 piksel olacak şekilde ayarlayın.
3. Bir MonoBehaviour sınıfı oluşturun ve adını GameOverlay olarak belirleyin.

İki adet Texture2D tipinde özel (private) değişken tanımlayın: \_healthIcon ve \_coinIcon. OnGUI() metodunu kullanarak \_healthIcon ikonunu ekranın sol üst köşesinden 20 piksel sağa ve 20 piksel aşağıya yerleştirin. İkonun boyutu 48x48 piksel olmalıdır.

"Health: 100" metnini \_healthIcon ikonunun hemen sağında (10 piksel boşluk bırakarak) yerleştirin.

\_coinIcon ikonunu ekranın sağ üst köşesinden 20 piksel sola ve 20 piksel aşağıya yerleştirin. İkonun boyutu 40x40 piksel olmalıdır.

"Coins: 50" metnini \_coinIcon ikonunun hemen solunda (10 piksel boşluk bırakarak) yerleştirin.

1. Bir MonoBehaviour sınıfı oluşturun ve adını CubeScaler olarak belirleyin. Bu script, fare y eksenindeki hareketiyle bir küp nesnesinin boyutunu değiştirecektir *(fare yukarı hareket ettiğinde küp büyümeli, aşağı hareket ettiğinde küçülmeli)*. Farenin hareketine bağlı olarak boyut değişim hızını belirleyen bir değişken oluşturun, varsayılan değeri 2f olsun. Fare hareketini zaman bağımsız hale getirin.

Not: (X, Y, Z eksenlerinde aynı ölçek uygulanmalı).

1. Bir MonoBehaviour sınıfı oluşturun ve adını CubeMover olarak belirleyin. Bu script, fare X eksenindeki hareketiyle bir küp nesnesinin konumunu değiştirecektir *(fare sağa hareket ettiğinde küp sağa, sola hareket ettiğinde sola gidecek)*. Farenin hareketine bağlı olarak konum değişim hızını belirleyen bir değişken oluşturun, varsayılan değeri 2f olsun. Fare hareketini zaman bağımsız hale getirin.
2. Bir MonoBehaviour sınıfı oluşturun ve adını CubeRotator olarak belirleyin. Bu script, bir küp nesnesinin "Player" tag'ine sahip bir oyuncunun tag’ından konum bilgilerini alarak ona doğru dönmesini sağlayacaktır.
3. Bir MonoBehaviour sınıfı oluşturun ve adını EnemyHider olarak belirleyin. Bu script, bir düşman nesnesinin "Player" tag'ine sahip bir oyuncuya olan mesafesine bağlı olarak aktif veya pasif olmasını sağlayacaktır. Bu değişken, oyuncunun pozisyonunu takip edecek player\_pos adında bir değişken tanımlayın. Eğer düşman konumu Player konumundan 10f birimden uzaksa (gameObject.SetActive() metodu kullanarak düşman pasif hale gelsin.
4. Bir player nesnesine verilece sağlık scripti için public bir TakeDamage metodu oluşturun. Metot, bir int parametresi (damageAmount) alsın. \_currentHealth değerini damageAmount kadar azaltın.
   * Eğer \_currentHealth sıfır veya sıfırdan küçükse, player nesnesini yok edin.
   * Eğer \_currentHealth 1 ile 30 arasındaysa, konsola "Ağır yaralı”
   * Eğer \_currentHealth 31 ile 60 arasındaysa “Hafif yaralı”
   * Eğer \_currentHealth 61’den büyükse, konsola "Sağlık durumu iyi"

Mesajlarını konsola yazdırın (Debug.Log kullanın).

1. Bir MonoBehaviour sınıfı oluşturun ve adını DistanceCalculator olarak belirleyin. sahnede bulunan kapsül ile küp. Script kapsüle verilecek. Küp nesnesi cube tag’ına sahiptir. Bu script küp nesnesine ait tag’ı bulacak ve kapsül ile arasındaki mesafeyi hesaplayıp konsola yazdıracaktır. Debug.Log kullanın.

Mesaj formatı şu şekilde olmalı: "Distance between cubes: [mesafe]".

1. Bir MonoBehaviour sınıfı oluşturun ve adını EnemyScanner olarak belirleyin. Bu script, ekranın tam ortasında bulunan bir silindirin raycast kullanarak etrafı taramasını ve "Enemy" tag'ine sahip bir nesneyi bulduğunda durmasını sağlayacaktır. Eğer raycast bir nesneye çarparsa, çarpılan nesnenin tag'ini kontrol edin. Eğer tag "Enemy" ise: tarama dursun enemy pozisyonunu konsola yazdırın. Eğer tag "Enemy" değilse, tarama devam etsin.