

# BLM0364 Oyun Programlama

Fizik

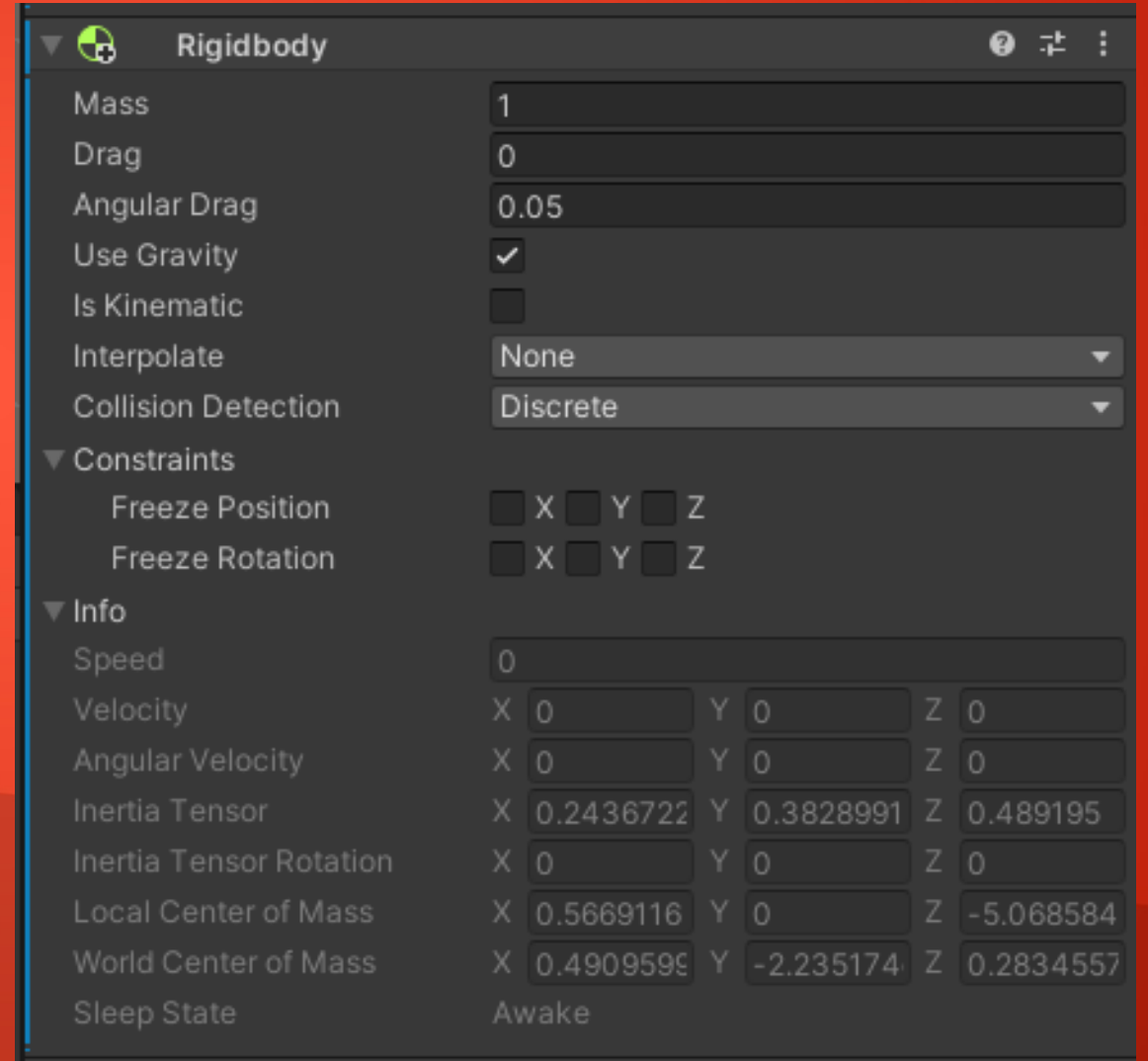
Arş. Gör. Muhammed Ömer Faruk Selvi

Bursa Teknik Üniversitesi

# Rigidbody

Rigidbody komponentine sahip nesne fizik kurallarına tabi olmaya başlar.

RB eklenmiş nesnenin kendisi veya child nesnelerinden en az biri Collider komponenti de içermelidir.

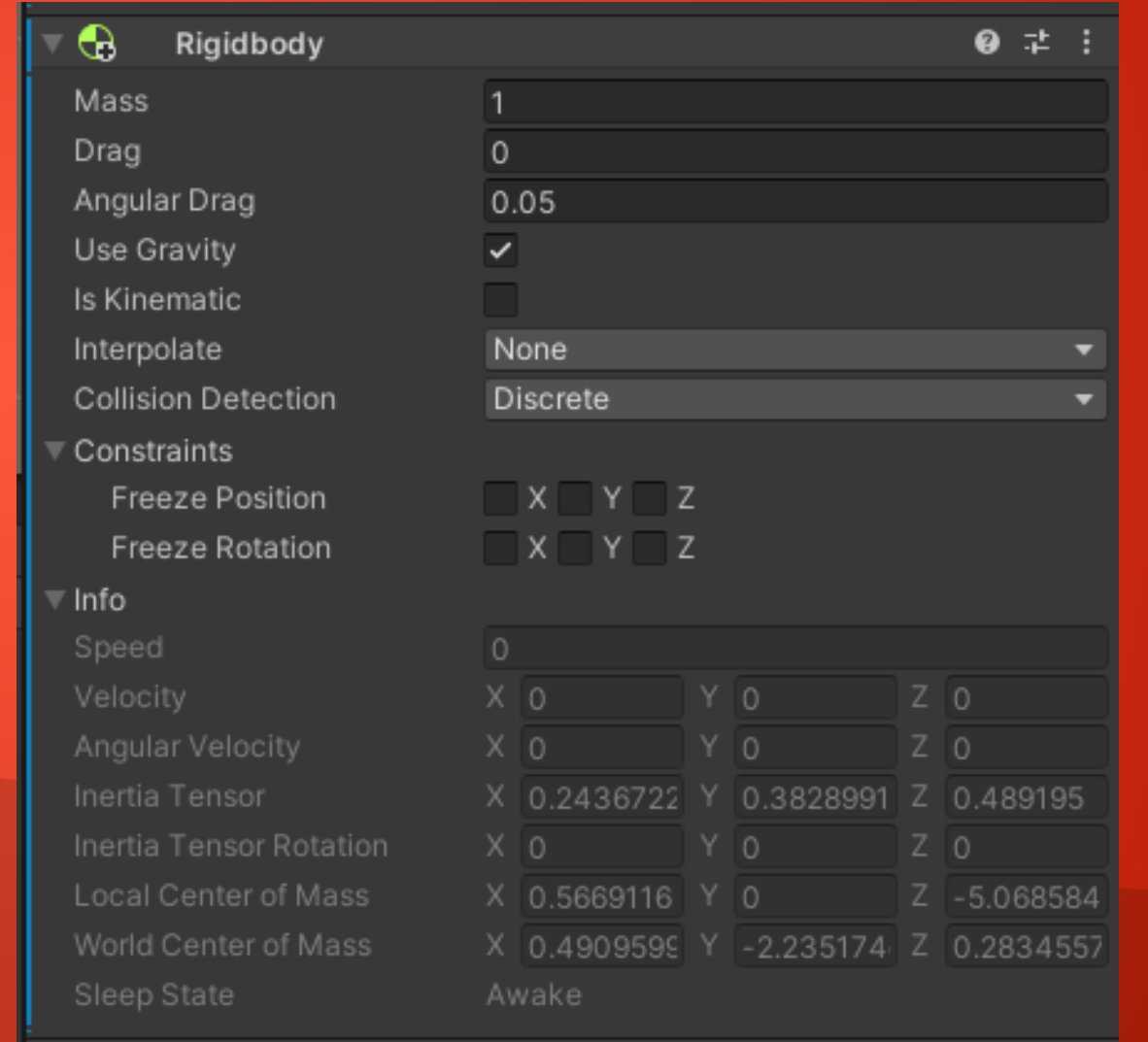


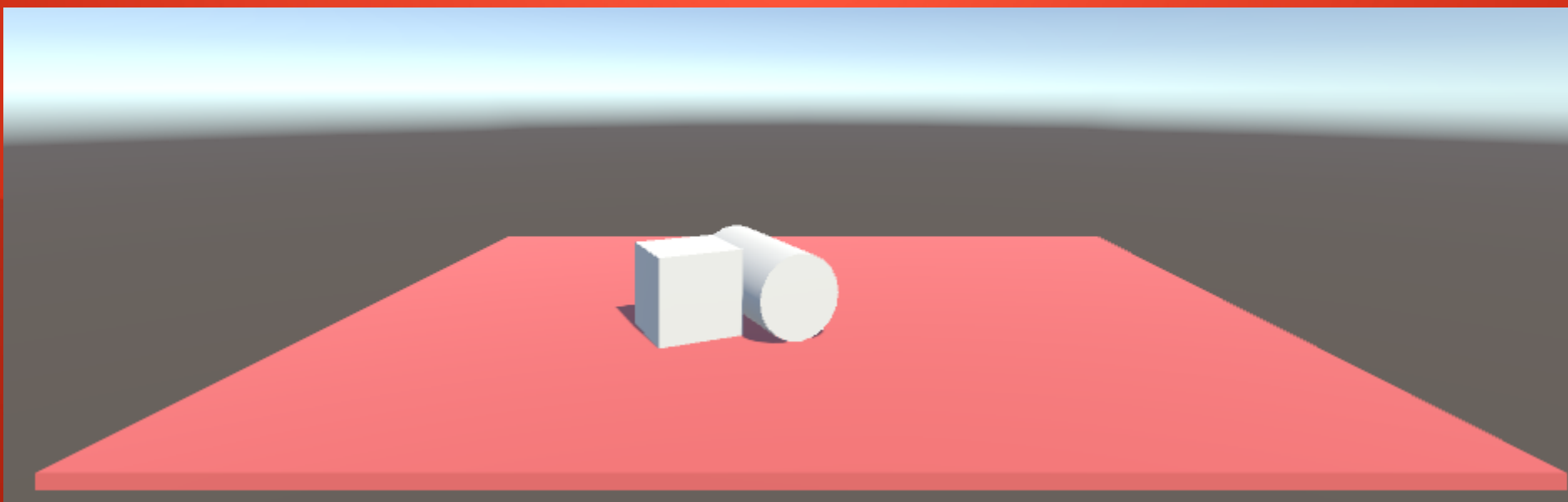
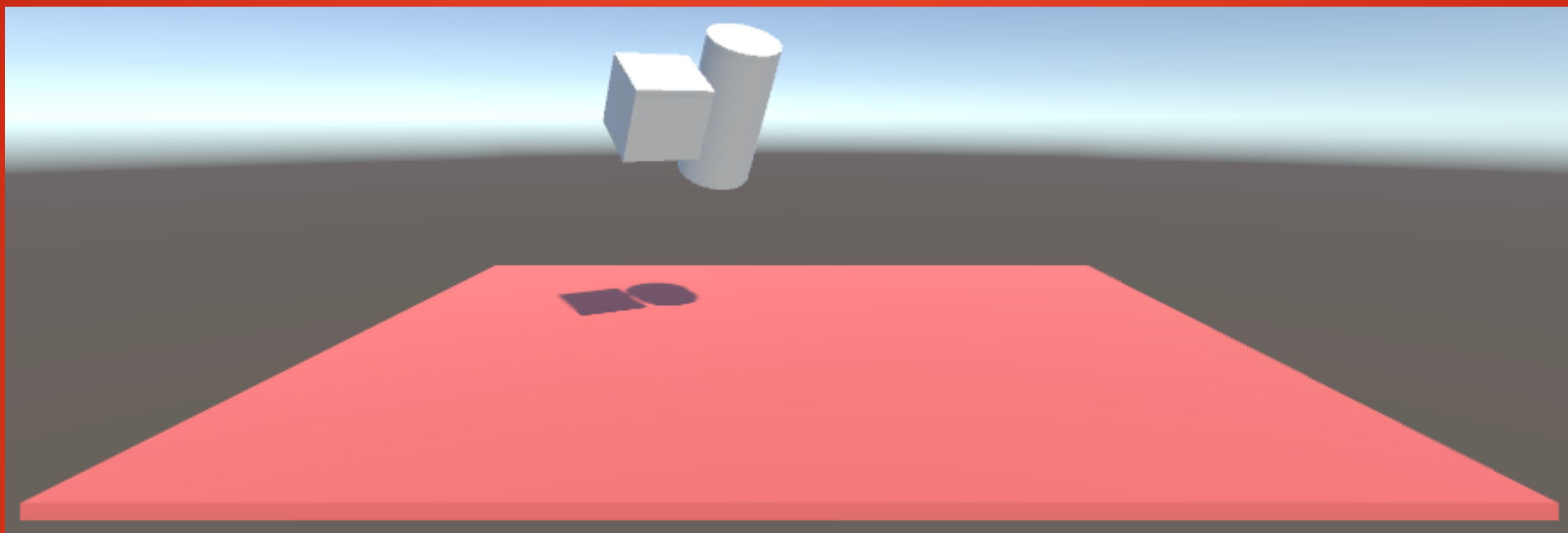
# Önemli Rigidbody Parametreleri

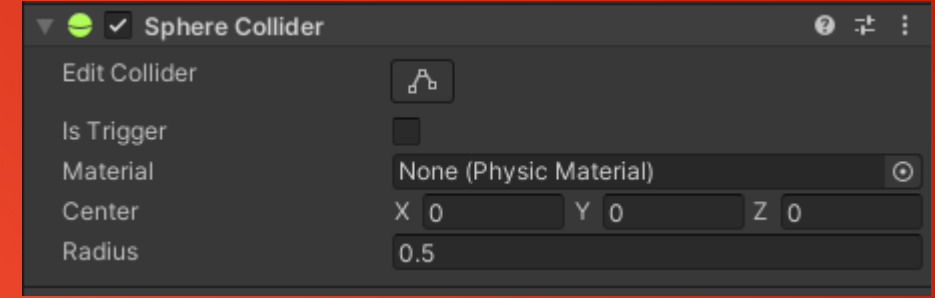
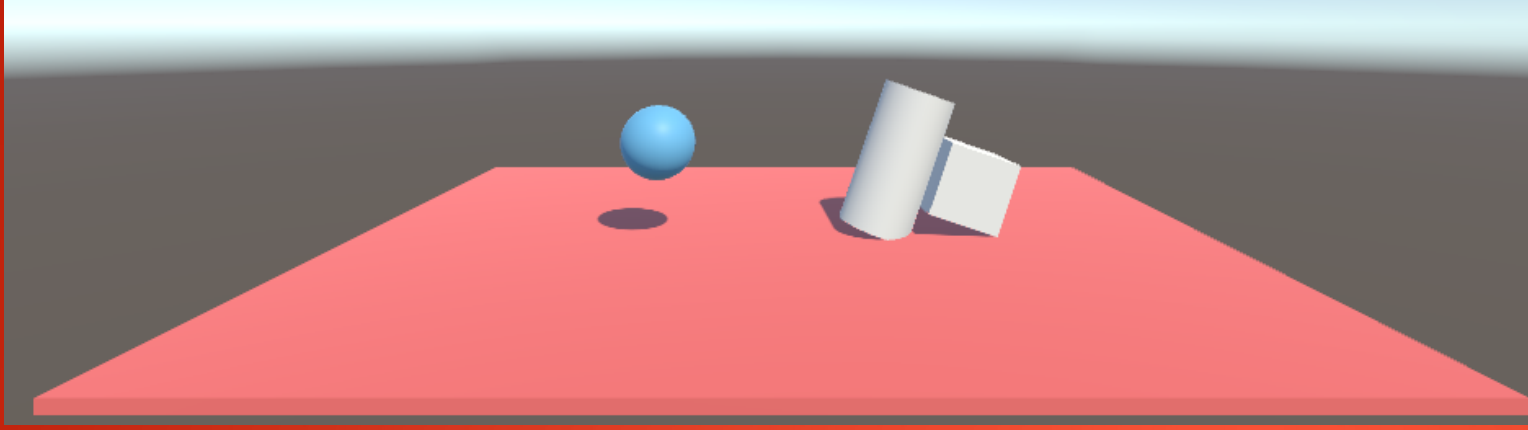
IsKinematic: Fiziğin katı cismi (igidbody) etkileyip etkilemediğini kontrol eder.

Interpolate: Sabit bir kare hızında çalışan fiziğin etkisini yumuşatmanıza olanak tanır.

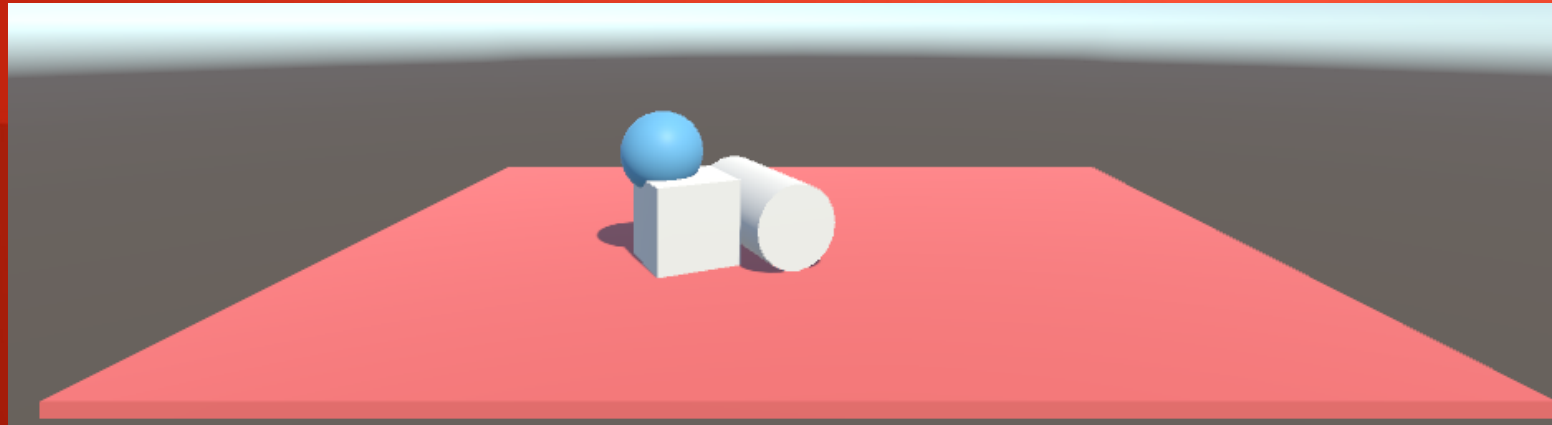
collisionDetectionMode: Rigidbody'nin çarpışma algılama modu.



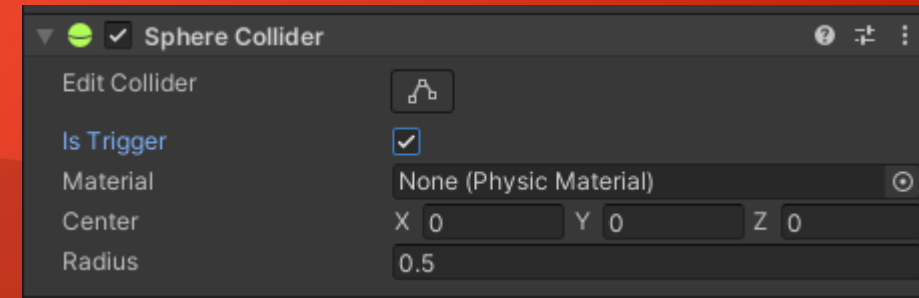




**Küre, normal collider'lı**



**Kürenin collider'ı trigger modunda**



# Collider

Collider, nesne fiziksel nesne çarpışması sağlar. Nesneler çarpışma sonucuna göre konum ve yön değiştirebilir.

Collider'ların etkili olabilmesi için çarpışan nesnelerden en az bir tanesi Rigidbody'li olmalıdır.

# Trigger

Trigger'lar çarpışma sonucu nesnelerin konum ve yönlerini değiştirmezler. Fiziksel simülasyon gerçekleşmez. Çarpışma tespit edilir ve nesneler davranışlarına aynen devam ederler.

Trigger'ların etkili olabilmesi için en az bir nesnenin aktif (statik olmayan Rigidbody komponenti olmalıdır (kinematic olabilir).

# Çarpışma Matrisi

## Collision detection occurs and messages are sent upon collision

	Static Collider	Rigidbody Collider	Kinematic Rigidbody Collider	Static Trigger Collider	Rigidbody Trigger Collider	Kinematic Rigidbody Trigger Collider
Static Collider		Y				
Rigidbody Collider	Y	Y	Y			
Kinematic Rigidbody Collider		Y				
Static Trigger Collider						
Rigidbody Trigger Collider						
Kinematic Rigidbody Trigger Collider						

## Trigger messages are sent upon collision

	Static Collider	Rigidbody Collider	Kinematic Rigidbody Collider	Static Trigger Collider	Rigidbody Trigger Collider	Kinematic Rigidbody Trigger Collider
Static Collider					Y	Y
Rigidbody Collider				Y	Y	Y
Kinematic Rigidbody Collider				Y	Y	Y
Static Trigger Collider		Y	Y		Y	Y
Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y
Kinematic Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y

# Fizik Özelliklerine Erişim

```
Rigidbody rbody;

void Start()
{
    rbody = GetComponent<Rigidbody>();
}

void FixedUpdate()
{
    if (Input.GetButton("Jump"))
    {
        rbody.AddForce(transform.up * 30.0f);
    }
}
```



# Çarpışma Kontrolünün Kodlanması

```
// void OnCollisionEnter(Collision collision)
// void OnCollisionStay(Collision collision)
void OnCollisionExit(Collision collision)
{
    Debug.Log(collision.transform.name);
}

// private void OnTriggerEnter(Collider other)
// private void OnTriggerStay(Collider other)
private void OnTriggerExit(Collider other)
{
    Debug.Log(other.transform.name);
}
```

# Uygulamalar

- İlkel bir kapsül nesnesinden oluşturduğunuz üçüncü şahıs (third-person) görünümlü karakterin yer değiştirme fonksiyonlarını fizik tabanlı hale getiriniz.
- Karakterin fiziksel hızını (`rbody.velocity`) konsola yazdırınız. Sadece ileri-geri veya sağ-sol hareket ederkenki hızını kontrol ediniz. Hem dikeyde hem yatayda hareket tuşlarını kullanarak çapraz gidildiğindeki hız farkını inceleyiniz. Karakterinizi çaprazda hız farkı olmayacak şekilde hareket ettiriniz.
- Karakteriniz haritada bulunan ve içerisinden geçilebilen bir nesneye dokunduğu zaman konsola 'Mission Completed' yazdırınız.