

# OYUN PROGRAMLAMA DERS RAPORU 1

AD-SOYAD: Kubilay İnanç

Öğrenci No: 22360859047

Github: <https://github.com/MandallF/Oyun-Programlama>

Üçüncü ders raporu olarak Unity uygulamasında aşağıdaki temel konuları inceledik.

## 1.Rigidbody Component:

Unity bize hazır olarak fizik motorları sağlıyor. 2D ve 3D için ayrı ayrı “Rigidbody” ve “Rigidbody 2D” fizik motorları mevcut. Bu bize fiziksel etkileşimleri gerçekçi bir şekilde oyunumuza eklememizi sağlıyor.

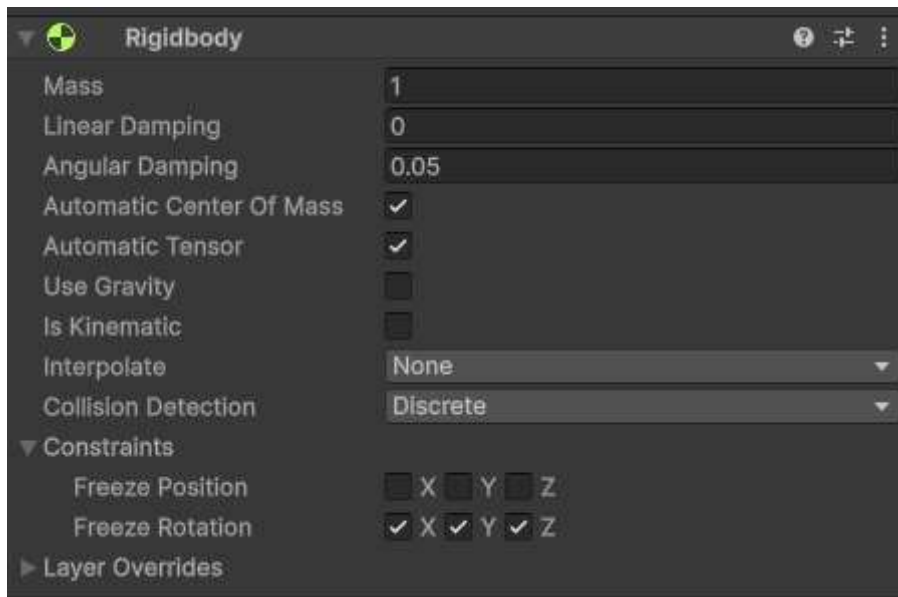
Kısaca: Rigidbody, bir objeye fizik kurallarını kazandırır diyebiliriz.

Eklendiği nesnede yerçekiminden etkilenip etkilenmeyeceğine, kinematik olup olmamasına (fizik kurallarından etkilenmesi), nesnenin ağırlığının olup olmamasına ve sürtünmesinin olup olmamasına bu eklenti sayesinde karar verebiliriz.

Ayrıca artık nesneyi hareket ettirirken fizik kuralları ile çakışmaması için Rigidbody’nin Transform.position hareket fonksiyonunu kullanmalıyız.

Rigidbody’nin tek başına çalıştığını düşünmeyin! Bu eklenti Collider ile birlikte çalışır.

Collider ise bir nesnenin şeklinin sınırlarını belirler diyebiliriz. Her nesne için farklıdır. Bu iki eklenti birlikte çalışarak çarpışmalar gerçekleşir.



Şekil1

## 2.Is Trigger :

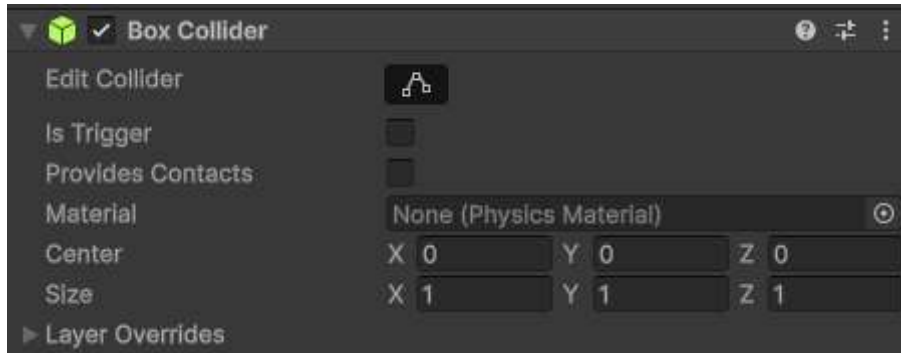
Unity’de bir Collider bileşeni, bu objenin alanını belirler demiştik.

Yani oyun motoru o alanı görünmez bir kutu gibi düşünür, başka objeler oraya çarptığında etkileşim olur.

Normalde bir collider, başka bir collider’la fiziksel çarpışma yaratır ama bazen biz çarpışmayı **fiziksel olarak değil, sadece algısal olarak** kullanmak isteriz. GTA 5’teki checkpoint gibi içinden geçeriz, bir etkileşim olur ancak nesnede fiziksel bir değişiklik olmaz.

İşte o durumu “Is Trigger” sağlıyor. Toparlayacak olursam “Is Trigger” kısaca: Collider’da fiziksel çarpışma yapmaz, sadece temas algılar. Yani obje görünmez hale gelir ama hâlâ “biri bana girdi mi?” sorusunu cevaplayabilir. Bunun fonksiyonları ise:

<b>OnTriggerEnter()</b>	Başka bir collider bu trigger alanına <i>girdiğinde</i>
<b>OnTriggerStay()</b>	Başka bir collider bu alan <i>içindeyken</i> her frame’de
<b>OnTriggerExit()</b>	Başka bir collider bu alandan <i>çıktığında</i>



Şekil2

## 3.Sonuç:

Unity uygulamasında bir nesneye nasıl fizik kurallarını ekleyeceğimizi öğrendik. Ayrıca nesnelerin bu etkileşimleri sırasında tepkinin illaki fiziksel olmaması gerektiğini anlamış olduk.