

Merhabalar,

İşlemleri kullanım rahatlığı açısından **Swagger UI** üzerinden gerçekleştirebilirsiniz, tercihe bağlı **Postman** de kullanılabilir.

Ben **Swagger UI** üzerinden anlatacağım.

Login, Token oluşturma, Token'ın DB'ye kayıt edilmesi, Register

1. **UserAPI** altından **Login** metodunu açın



2. **Try it out** butonuna tıklayın



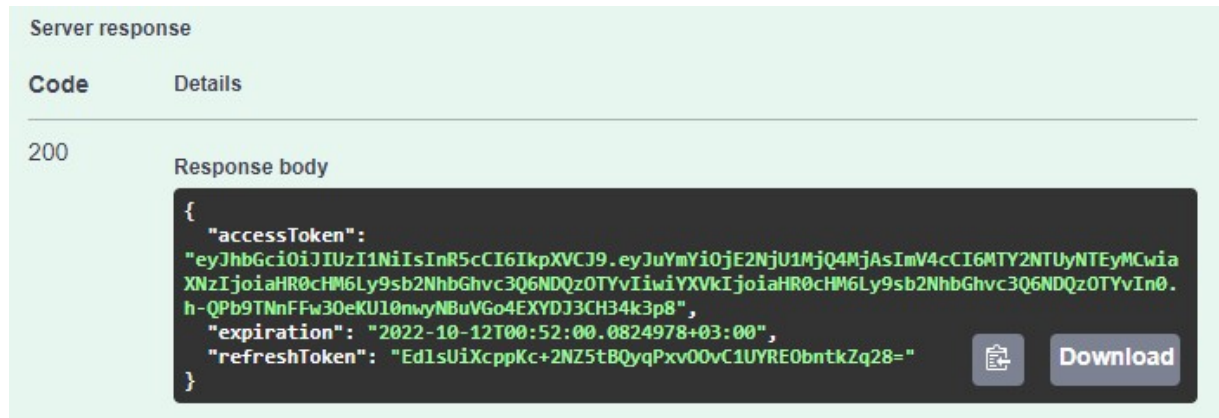
3. Aşağıdaki örnek kullanıcı bilgilerini girerek **Execute** yapın

Email * required
string(\$email)

Password * required
string(\$password)

Execute

4. Oluşan “**accessToken**” bilgisini kopyalayın (*tırnaklar hariç*)



5. Ekranın üst kısmında bulunan “**Authorize**” butonuna tıklayın

MovieSuggestion ^{v1} OAS3

/swagger/v1/swagger.json



Authorize



6. Açılan modal'daki **Value** alanına önce **"Bearer"** yazın, ardından bir boşluk bırakarak kopyaladığınız **accessToken**'i yapıştırıp **"Authorize"** butonuna tıklayın

Available authorizations

Bearer (apiKey)

JWT containing userid claim
Name: Authorization
In: header
Value:

Bearer eyJhbGciOiJIUzI1NiIs

Authorize

Close



Artık authorized oldunuz.

Yukarıdaki işlemleri yeni kullanıcı oluşturarak yapmak için **Register** yöntemini izleyebilirsiniz:

1. **UserAPI** altından **Register** metodunu açın ve **Try it out** butonuna tıklayın

UserAPI

POST

/api/UserAPI/Register

Authorize



2. **FullName**, **Email** ve **Password** alanlarını *string* tipinde doldurun ve **Execute** edin

Bu işlemin ardından **Login** işlemini tekrarlayabilirsiniz.

** Authorization süresi **10 dakika** olarak ayarlanmıştır,

10 dakikayı geçtiğiniz takdirde lütfen **Logout** olup, **Login** ve **Authorize** adımlarını tekrarlayın

1. **UserAPI** altından **GET** metodunu açın ve **Try it out** butonuna tıklayın

GET

/api/UserAPI

Authorize

Kayıtlı olan tüm kullanıcıların **Id**, **Ad Soyad** ve **Email** bilgileri getirilecektir (*Şifreler hariç tutuldu*).

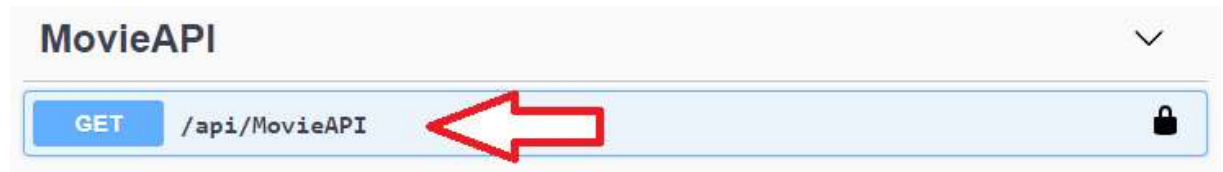
The Movie Database (TMDb) API Servisi kullanılarak Film bilgilerinin veritabanına kayıt edilmesi

Her saat başı **20** adet film (*Trending kategorisinden*) çekilerek **Film Adı** ve **Kapak Görseli** ile birlikte servisten gelen **Sayfa numarasını** da alarak **Movie** tablosuna kayıt edilmektedir.

Sayfa numarasının kullanılması amacı; *Trending* kategorisinin sürekli yenilenmiyor oluşundan ötürü kayıt edilen sayfaların geçilerek diğer sayfalardaki verilerin alınıp db'ye yazılmasıdır.

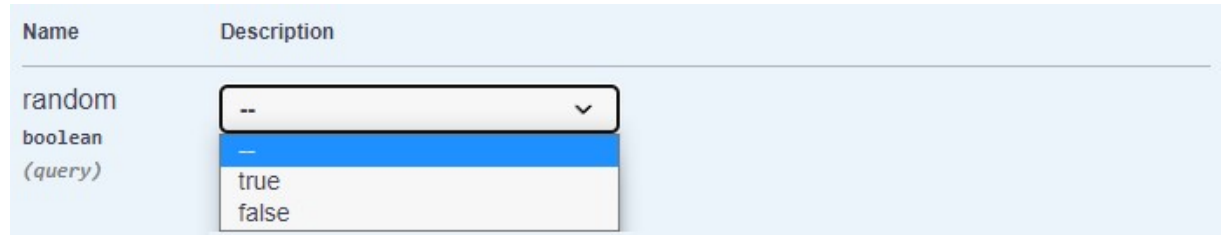
Movie tablosundan Film bilgilerinin getirilmesi, Parametrik filtreleme

1. **MovieAPI** altından **GET** metodunu açın ve **Try it out** butonuna tıklayın

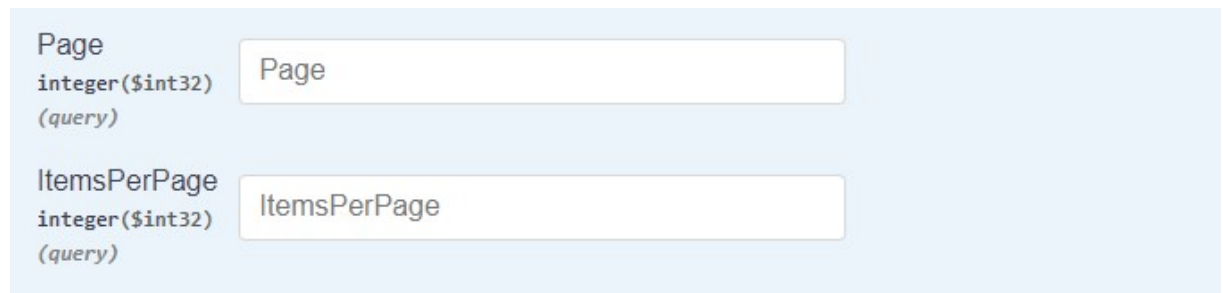


*** **Herhangi bir parametre girmez iseniz** karşınıza ilk 10 adet Film bilgisi gelecektir.

2. **Random** parametresi verilerin rastgele olarak gelmesini sağlamaktadır (*true/false*)



3. **Page** parametresi kaçıncı sayfadan veri getirileceği,
ItemsPerPage parametresi de kaç adet veri getirileceğidir.



Seçimlerinizi/işlemlerinizi tamamladıktan sonra **Execute** etmeyi unutmayın

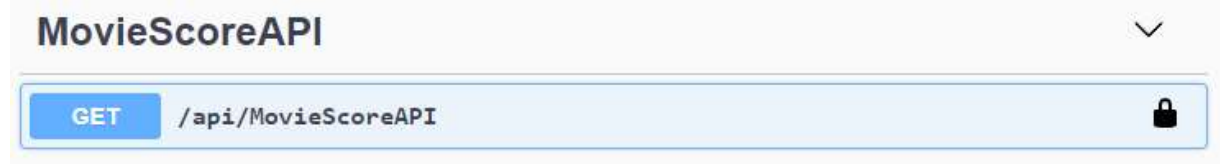
1. **MovieAPI** altından **GetMovie** metodunu açın ve **Try it out** butonuna tıklayın

İstenilen **Id** parametresi alanına **Movie** tablosundan bir Film Id'si girin,
Size **film adı**, **resim** ve **ortalama puan** bilgilerini getirecektir.



MovieScore tablosundan Film bilgileri ile kullanıcılar tarafından verilen Puan, Not ve filmin aldığı ortalama puan bilgilerinin getirilmesi, Puan ve Not kayıt işlemleri

1. **MovieScoreAPI** altından **GET** metodunu açın ve **Try it out** butonuna tıklayın



Film adı, filmi puanlayan ve filme not giren **kullanıcının Adı Soyadı**, verilen **puan**, girilen **not**, **ortalama puan** bilgileri getirilecektir.

1. **MovieScoreAPI** altından **POST** metodunu açın ve **Try it out** butonuna tıklayın



MovieId parametresine **Movie** tablosundan bir **Film Id**'si girin,

UserId parametresine **User** tablosundan bir **Kullanıcı Id**'si girin,

Rate parametresine **1-10 arasında** bir tam sayı girin (*1 ve 10 dahil*),

Note parametresine **250** karakteri aşmamak şartıyla *not* girin ve ardından **Execute** edin

1. **MovieScoreAPI** altından **GetMovieScore** metodunu açın ve **Try it out** butonuna tıklayın



movieId ve **userId** parametrelerini bir üstteki anlatımda olduğu gibi doldurup **Execute** edin,

Sadece belirtilen filme ait, belirtilen kullanıcının Puan ve Not bilgilerini Film bilgileri ile getirmektedir.

*** **TMDB** üzerinden puan bilgileri çekilmemiştir, puanlar standart olarak **0** gelmekte ve kullanıcıların girdikleri puanlar ile ortalamalar belirlenmektedir.

Filmin tavsiye edilmesi ile birlikte tavsiye maili gönderilmesi

1. **MailAPI** altından **POST** metodunu açın ve **Try it out** butonuna tıklayın

MailAPI

POST /api/MailAPI

MovieId parametresine **Movie** tablosundan bir **Film Id**'si girin,

ToAddressName parametresine mail gönderilecek kişinin Adını Soyadını girin,

ToAddress parametresine mail gönderilecek kişinin Email adresini girin ve **Execute** edin

İşlem **true** döndüğü takdirde karşı tarafa şu örnekte bir mail gidecektir:

□ ☆ ▷ Test Mail Service "Yüzüklerin Efendisi: Güç Yüzükleri" bu filmi mutlaka izlemelisin! - Film Adı : Yüzüklerin Efendisi: Güç Yüzükleri Puanı : 0 İyi seyirler...

Mail içeriği:

'Yüzüklerin Efendisi: Güç Yüzükleri' bu filmi mutlaka izlemelisin! ▷ Gelen Kutusu x

 **Test Mail Service** <testmailsent@yandex.com>
Alıcı: ben ▾



Film Adı : Yüzüklerin Efendisi: Güç Yüzükleri
Puanı : 0

İyi seyirler...

↩ Yanıtla ↪ Yönlendir

Saygılarımla,

Gökhan SAVAŞ