



**SAMSUN ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**  
**YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ**  
2023 – 2024 Akademik Yılı Güz Dönemi

31/01/2024

MYAZ201 – Yazılım Gereksinimi ve Modelleme Dersi Bütünleme Sınavı

Değerli öğrencilerim,

MYAZ201 Yazılım Gereksinimi ve Modelleme dersinin bütünleme sınavına hoş geldiniz. Sınav boyunca uyulması gereken kurallar aşağıda listelenmiştir. Sınavda uyulması gereken kuralları lütfen dikkatle okuyunuz.

- Sınav süresi 90 dakikadır.
- Sınav için kullanılması gereken materyaller (dokümanlar ve kaynak kodlar vb.) öğrenme yönetim sistemi üzerinden indirilmelidir.
- Sınav süresince internet ve yapay zekâ destekli araçların kullanımı yasaktır.
- Sınav süresince bilgisayar kullanabilirsiniz. Kaynak kodlar, editör kullanımı ve ders notlarına erişim sağlayabilirsiniz.
- Sınav başladıktan sonra, sınav bitimine kadar sınav salonlarına giriş/çıkış yapılmayacaktır.

**Cevap Kağıdının Hazırlanması**

- Cevap kâğıdınızın sağ üst köşesine:
  - Adınızı soyadınızı,
  - Öğrenci numaranızı
  - **Şube numaranızı**mutlaka yazınız.
- Sınavda sadece sizden istenen ifadeleri cevap kâğıdınıza geçiriniz. Tanımı istenmeyen ifadeleri yazmayınız. **Tek bir cevap kâğıdı kullanma noktasında azami özen gösteriniz.**
- Birden fazla cevap kâğıdı kullanmanız durumunda lütfen ilgili cevap kâğıtlarını numaralandırınız ve sınav görevlisinden cevap kâğıtlarını zımbalamasını isteyiniz.
- Cevap kâğıdınızda her bir soruyu net bir şekilde birbirinden ayırınız.



## SORULAR

**App.db** isimli veri tabanını indiriniz, gerekli modelleme ve kodlamaları yapınız. Aşağıda verilen isterleri dikkate alınız.

### İsterler

- Geliştirilecek olan uygulama bir **dotnet web api** uygulaması olmalıdır.
- Veri tabanı işlemleri için **Code First** yaklaşımı kullanılmalıdır.
- **ORM** için **Entity Framework Core** kullanılmalıdır.
- Veri tabanı işlemleri için kullanılacak sınıf **RepositoryContext** olarak isimlendirilmelidir.

1. Yukarıda uç noktaya gidildiğinde aşağıdaki gibi çıktı üreten bir eylem tanımlayınız.  
**CarsController.GetAllCars()** isimli metodu cevap kağıdınıza geçirin (25 p).

GET /api/cars		
Parameters		
No parameters		
Responses		
Code	Description	Links
200	Success	No links

```
1 [
2   {
3     "brand": "Tennessee",
4     "model": "Buckinghamshire generating XSS Metal",
5     "year": 2020,
6     "engine": {
7       "cylinderCount": 4,
8       "fuelType": "Benzin"
9     }
10  },
11  { ...
12  },
13  { ...
14  },
15  { ...
16  },
17  { ...
18  },
19  { ...
20  },
21  { ...
22  },
23  { ...
24  },
25  { ...
26  },
27  { ...
28  },
29  { ...
30  },
31  { ...
32  },
33  { ...
34  },
35  { ...
36  },
37  { ...
38  },
39  { ...
40  },
41  { ...
42  },
43  { ...
44  },
45  { ...
46  },
47  { ...
48  },
49  { ...
50  },
51  { ...
52  },
53  { ...
54  },
55  { ...
56  },
57  { ...
58  },
59  { ...
60  },
61  { ...
62  },
63  { ...
64  },
65  { ...
66  },
67  { ...
68  },
69  { ...
70  },
71  { ...
72  },
73  { ...
74  },
75  { ...
76  },
77  { ...
78  },
79  { ...
80  },
81  { ...
82  },
83  { ...
84  },
85  { ...
86  },
87  { ...
88  },
89  { ...
90  },
91  { ...
92  },
93  { ...
94  },
95  { ...
96  },
97  { ...
98  },
99  { ...
100 }
```



2. Aşağıda belirtilen uç noktasına gelen bir istek sonucunda sayfalama yapan bir eylem tasarımı yapınız.  
`CarsController.GetAllCarsWithPagination(int, int)` eylemini cevap kağıdınıza geçiniz.

GET

/cars/page/{pageNumber}

Parameters

Try it out

Name	Description
<b>pageNumber</b> * required integer(\$int32) (path)	pageNumber
pageSize integer(\$int32) (query)	pageSize



3. Aşağıda belirtilen uç noktaya gelen isteklere bağlı olarak ilgili kaynakları oluşturan eylemi tanımlayınız.

`CarsController.AddCars (Car[])` eylemini cevap kağıdınıza geçiriniz (25 p).

POST

/api/cars

Parameters

Try it out

No parameters

Request body

application/json

Example Value | Schema

```
[
  {
    "carId": 0,
    "brand": "string",
    "model": "string",
    "year": 0,
    "price": 0,
    "engineId": 0,
    "engine": {
      "engineId": 0,
      "horsepower": 0,
      "cylinderCount": 0,
      "fuelType": "string"
    }
  }
]
```



4. Aşağıda belirtilen uç noktaya gelen isteklere bağlı olarak ilgili kaynakları güncelleyen eylemi tanımlayınız.  
`CarsController.UpdateCar(int, Car)` eylemini cevap kağıdınıza geçiniz (25 p).

PUT

/api/cars/{id}

^

Parameters

Try it out

Name	Description
<b>id</b> * required	
integer(\$int32)	id
(path)	