

READ_ME

METHODLAR:

GET → Menüye göz atmak için restorana giden bir müşteri düşünün. Müşteri, menüyü görüntüleyerek sunulan yemekleri görebilir ancak menüyü değiştiremez veya yeni bir yemek ekleyemez.

POST → Menüsü değişmez, ancak yeni bir sipariş eklenir.

PUT → önceki siparişini günceller.

Response Codes:

405: Method Not Allowed - Bu durum kodu, bir kaynağa gönderilen HTTP methodunun desteklenmediğini belirtir. Örneğin, bir GET isteği yapılması gereken bir kaynağa POST isteği yapıldığında karşımıza çıkabilir. Günlük hayatta, bir web sayfasında bir formu doldururken "Gönder" butonuna basmadan önce "bu formu gönderebilir miyim?" sorusunu hatırlayabiliriz.

200: OK - Bu durum kodu, bir HTTP isteğinin başarıyla tamamlandığını ve geri dönen içeriğin isteğe uygun olduğunu belirtir. Örneğin, bir web sayfasının başarıyla yüklendiği durumda alınabilir. Günlük hayatta, bir internet tarayıcısına bir web sayfası URL'sini girdiğimizde sayfanın başarıyla yüklenmesi bu durumu temsil eder.

400: Bad Request - Bu durum kodu, bir HTTP isteğinin hatalı olduğunu ve sunucu tarafından anlaşılamadığını belirtir. Örneğin, bir API'ye geçersiz bir parametre gönderildiğinde alınabilir. Günlük hayatta, bir web sayfasındaki bir formu eksik ya da hatalı doldurduğumuzda bu hatayı alabiliriz.

404: Not Found - Bu durum kodu, istenen kaynağın bulunamadığını belirtir. Örneğin, bir web sayfasının yanlış bir URL ile istendiğinde alınabilir. Günlük hayatta, bir internet tarayıcısına var olmayan bir web sayfasının URL'sini girdiğimizde bu hatayı alabiliriz.

Payload nedir?

Payload, bir HTTP isteği veya yanıtının gövde kısmında taşıdığı veri setidir. Örneğin, bir POST isteği yaparken gönderilen veriler veya bir GET isteğine verilen parametreler bir

payload oluşturabilir. Günlük hayatta, bir e-posta gönderirken e-posta metni ve ekleri içeren bölüm payload olarak düşünülebilir.

API 1: Get All Products List

```
{ "responseCode": 200, "products": [{ "id": 1, "name": "Blue Top", "price": "Rs. 500", "brand": "Polo", "category": { "usertype": { "usertype": "Women", "category": "Tops" } }, { "id":  
.....  
Ran 1 test in 0.892s  
OK
```

API 2: POST To All Products List

```
{ "responseCode": 405, "message": "This request method is not supported." }  
200  
.  
-----  
Ran 1 test in 0.857s  
OK
```

API 3: Get All Brands List

```
200  
{ "responseCode": 200, "brands": [{ "id": 1, "brand": "Polo"}, { "id": 2, "brand": "H&M"}, { "id": 3, "brand": "Madame"}, { "id": 4, "brand": "Madame"}, { "id": 5  
.....  
Ran 1 test in 0.880s  
OK
```

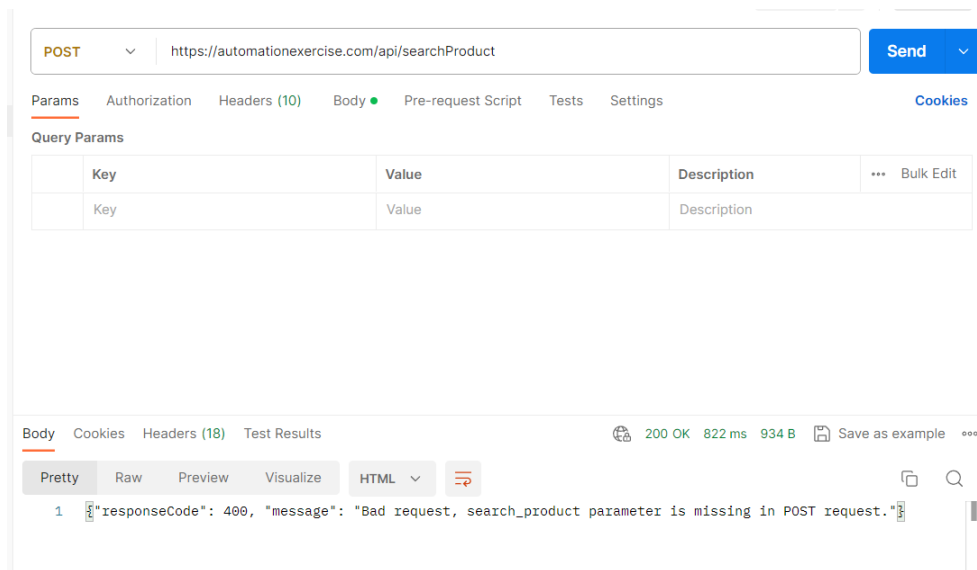
API 4: PUT To All Brands List

```
{"responseCode": 405, "message": "This request method is not supported."}
.
-----
Ran 1 test in 0.820s

OK
```

API 5: POST To Search Product

search_product parameter: API'ye gönderilen POST isteğinde bir arama sorgusu belirtmek için kullanılır.



POST <https://automationexercise.com/api/searchProduct> Send

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

Query Params

Key	Value	Description	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (18) Test Results 200 OK 822 ms 934 B Save as example

Pretty Raw Preview Visualize HTML

```
1 {"responseCode": 400, "message": "Bad request, search_product parameter is missing in POST request."}
```

```
200
.
-----
Ran 1 test in 0.870s

OK
```

API 6: POST To Search Product without search_product parameter

```
FAIL: test_api_6 (__main__.Numbers)
-----
Traceback (most recent call last):
  File "C:/Users/Dell/PycharmProjects/pythonProject1/main.py", line 41, in test_api_6
    self.assertEqual(400, response.status_code)
AssertionError: 400 != 200
-----

Ran 1 test in 0.860s

FAILED (failures=1)
```

API 7: POST To Verify Login with valid details & API 8: POST To Verify Login without email parameter

```

-----

Ran 1 test in 0.849s

OK
```

API 9: DELETE To Verify Login

```
-----
Traceback (most recent call last):
  File "C:/Users/Dell/PycharmProjects/pythonProject1/main.py", line 66, in test_api_9
    self.assertEqual(cevap.status_code, 405, "not supported!")
AssertionError: 200 != 405 : not supported!
-----

Ran 1 test in 0.850s
```

API 10: POST To Verify Login with invalid details

```
AssertionError: 404 != 200
```

```
-----  
Ran 1 test in 0.887s
```

```
FAILED (failures=1)
```

API 11: POST To Create/Register User Account

```
AssertionError: 201 != 200 : User created!
```

```
-----  
Ran 1 test in 0.849s
```

API 12: DELETE METHOD To Delete User Account

```
-----  
Traceback (most recent call last):
```

```
  File "C:/Users/Dell/PycharmProjects/pythonProject1/main.py", line 112, in test_api_12  
    self.assertEqual(200, cevap.status_code, "User update!")
```

```
AssertionError: 200 != 405 : User update!
```

```
-----  
Ran 1 test in 0.881s
```

```
FAILED (failures=1)
```

API 13: PUT METHOD To Update User Account

```
AssertionError: 'User updated!' != 'Account not found!'
- User updated!
+ Account not found!
```

```
-----
Ran 1 test in 0.992s
```

API 14: GET user account detail by email

```
AssertionError: 'User_id' not found in {'responseCode': 404, 'message': 'Account not found with this email, try another email!'}
-----
Ran 1 test in 0.916s

FAILED (failures=1)
```