

# Backend Exam API reference

---

## Customer

`/api/customer` endpoint for managing customers

---

### GET Customer by ID

`localhost:8080/api/customer/1`

---

### GET All customers

`localhost:8080/api/customer`

Gets all customers, can specify pagesize and page number. Default pagesize is 5, default page is 1. Page size is not required to specify.

---

### POST Create new customer

`localhost:8080/api/customer`

**Body** raw (json)

---

json

```
{
  "customerName": "new name",
  "customerEmail": "New email"
}
```

### POST Add address to customer

`localhost:8080/api/customer/1/address/1`

localhost:8080/api/customer/1/address/1

---

## DELETE Delete customer

localhost:8080/api/customer

### Body raw (json)

```
json

{
  "customerId": 1
}
```

---

## PUT Update customer

localhost:8080/api/customer

### Body raw (json)

```
json

{
  "customerId": 2,
  "customerName": "new name",
  "customerEmail": "New email"
}
```

---

## Address

`/api/address` endpoint for managing customers

---

## GET Get address by id

localhost:8080/api/address/1

---

## GET Get all addresses

localhost:8080/api/address

Get all addressess. Has pagination and page size.

---

## POST Add address

localhost:8080/api/address

**Body** raw (json)

---

json

```
{
  "country": "Norway",
  "area": "Oslo",
  "streetAndNumber": "Cool Street 43"
}
```

---

## DELETE Delete address

localhost:8080/api/address

**Body** raw (json)

---

json

```
{
  "addressId": 1
}
```

---

## PUT Update address

localhost:8080/api/address

**Body** raw (json)

---

json

```
json

{
  "addressId": 3,
  "country": "Cool New Country",
  "area": "OlsoStreet",
  "streetAndNumber": "Not So Cool Street 43"
}
```

---

## Order

`/api/order` endpoint for managing orders

---

### GET Get order by ID

localhost:8080/api/order/3

---

### GET Get all orders

localhost:8080/api/order

Get list of all orders. Default pagesize is 10 and default page is 1. Can be changed.

---

### POST Create new order

localhost:8080/api/order

**Body** raw (json)

---

```
json

{
  "orderDescription": "New customer order"
}
```

---

### POST Add machine to order

localhost:8080/api/order/2/machine/1

---

## POST Add customer to order

localhost:8080/api/order/21/customer/12

## DELETE Delete order

localhost:8080/api/order

**Body** raw (json)

```
json

{
  "orderId": 4
}
```

## PUT Update order

localhost:8080/api/order

**Body** raw (json)

```
json

{
  "orderId": 2,
  "orderDescription": "New order Description"
}
```

## Machine

/api/machine endpoint for managing machines

## GET Get machine by ID

localhost:8080/api/machine/1

Get machine by ID.

---

## GET Get all machines

localhost:8080/api/machine

Get all machines. Pagesize and number does not need to be set, default values is pagesize = 10 and pagenumber = 1.

---

## POST Create new machine

localhost:8080/api/machine

**Body** raw (json)

---

json

```
{
  "machineName": "New Machine Name",
  "machineDescription": "New machine description"
}
```

## POST Add subassembly to machine

localhost:8080/api/machine/1/subassembly/1

---

## DELETE Delete machine

localhost:8080/api/machine

**Body** raw (json)

---

json

```
{
  "machineId": 2
}
```

---

## PUT Update machine

localhost:8080/api/machine

### Body raw (json)

---

json

```
{
  "machineId": 3,
  "machineName": "Updated machine name",
  "machineDescription": "Updated machine description"
}
```

---

## Subassembly

`/api/subassembly` endpoint for managing subassemblies

---

## GET Get subassembly by ID

localhost:8080/api/subassembly/1

---

## GET Get all subassemblies

localhost:8080/api/subassembly

Get all subassemblies. Default page size is 10. Uses query params to set page and pagesize.

---

## POST Create new subassembly

localhost:8080/api/subassembly

### Body raw (json)

---

json

```
{
  "subassemblyName": "Subassmebly for a Speaker Cone manufacturing machine",
  "subassemblyDescription": "This subassmebly creates the speaker coil"
}
```

---

## POST Add part to subassembly

localhost:8080/api/subassembly/2/part/2

---

## DELETE Delete subassembly

localhost:8080/api/subassembly

### Body raw (json)

json

```
{
  "subassemblyId": 13
}
```

---

## PUT Update subassembly

localhost:8080/api/subassembly

### Body raw (json)

json

```
{
  "subassemblyId": 2,
  "subassemblyName": "M5 Bolt",
  "subassemblyDescription": "Used for bolting stuff together"
}
```

---

## Part

/api/part endpoint for managing parts



---

## GET Get part by ID

localhost:8080/api/part/1

---

## GET Get all parts

localhost:8080/api/part

Get list of all parts.

---

## POST Create new part

localhost:8080/api/part

**Body** raw (json)

json

```
{  
  "partName": "M5 Bolt",  
  "partDescription": "Used for bolting stuff together"  
}
```

---

## DELETE Delete part

localhost:8080/api/part

**Body** raw (json)

json

```
{  
  "partId": 1  
}
```

---

## DELETE Delete part with path variable

localhost:8080/api/part/14

### Body raw (json)

---

json

```
{  
  "partId": 1  
}
```

## PUT Update part

localhost:8080/api/part

### Body raw (json)

---

json

```
{  
  "partId": 3,  
  "partName": "M5 Bolt",  
  "partDescription": "Used for bolting stuff together"  
}
```