

3. WebApi_Handson 1.0 OAS 3.0

<http://localhost:5181/swagger/v1/swagger.json>

Employee

GET

/api/Employee/GetStandard

Parameters

Cancel

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'GET' \  
  'http://localhost:5181/api/Employee/GetStandard' \  
  -H 'accept: text/plain'
```



Request URL

http://localhost:5181/api/Employee/GetStandard

Server response

Code

Details

400

Undocumented

Error: Bad Request

Response body

Invalid request - No Auth token



Download

Response headers

```
content-type: text/plain; charset=utf-8  
date: Wed, 09 Jul 2025 14:24:30 GMT  
server: Kestrel  
transfer-encoding: chunked
```

Responses

Code

Description

Links

200

OK

No links

Media type

text/plain



Controls Accept header.

Example Value | Schema

Code

Description

Links

```
[
  {
    "id": 0,
    "name": "string",
    "salary": 0,
    "permanent": true,
    "department": {
      "id": 0,
      "name": "string"
    },
    "skills": [
      {
        "id": 0,
        "name": "string"
      }
    ],
    "dateOfBirth": "2025-07-09T14:25:05.436Z"
  }
]
```

POST

/api/Employee

Parameters

Cancel

Reset

No parameters

Request body

application/json

```
{
  "id": 0,
  "name": "string",
  "salary": 0,
  "permanent": true,
  "department": {
    "id": 0,
    "name": "string"
  },
  "skills": [
    {
      "id": 0,
      "name": "string"
    }
  ],
  "dateOfBirth": "2025-07-09T14:24:37.060Z"
}
```

Execute

Clear

Responses

Curl

```
curl -X 'POST' \
  'http://localhost:5181/api/Employee' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 0,
    "name": "string",
    "salary": 0,
    "permanent": true,
    "department": {
      "id": 0,
      "name": "string"
    },
    "skills": [
      {
        "id": 0,
```

```
"name": "string"
}
],
"dateOfBirth": "2025-07-09T14:24:37.060Z"
}'
```



Request URL

http://localhost:5181/api/Employee

Server response

Code	Details
400 <i>Undocumented</i>	<div>Error: Bad Request</div> <div>Response body</div> <div>Invalid request - No Auth token</div> <div> Download</div> <div>Response headers</div> <div>content-type: text/plain; charset=utf-8 date: Wed,09 Jul 2025 14:24:40 GMT server: Kestrel transfer-encoding: chunked</div>

Responses

Code	Description	Links
201	Created	No links

PUT

/api/Employee/{id}



Parameters

Cancel

Reset

NameDescription

id * required

integer(\$int32)
(path)

1

Request body

application/json



```
{
  "id": 0,
  "name": "string",
  "salary": 0,
  "permanent": true,
  "department": {
    "id": 0,
    "name": "string"
  },
  "skills": [
    {
      "id": 0,
      "name": "string"
    }
  ],
  "dateOfBirth": "2025-07-09T14:24:45.689Z"
}
```



Execute

Clear

Responses

Curl

```
curl -X 'PUT' \
  'http://localhost:5181/api/Employee/1' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 0,
    "name": "string",
    "salary": 0,
    "permanent": true,
    "department": {
      "id": 0,
      "name": "string"
    },
    "skills": [
      {
        "id": 0,
        "name": "string"
      }
    ],
    "dateOfBirth": "2025-07-09T14:24:45.689Z"
  }'
```



Request URL

http://localhost:5181/api/Employee/1

Server response

Code	Details
------	---------

400	Error: Bad Request
-----	--------------------

Undocumented

Response body

Invalid request - No Auth token



Download

Response headers

content-type: text/plain; charset=utf-8
date: Wed, 09 Jul 2025 14:24:49 GMT
server: Kestrel
transfer-encoding: chunked

Responses

Code	Description	Links
200	OK	No links

WeatherForecast



GET

/WeatherForecast



Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	<div>OK</div> <div>Media type</div> <div>text/plain</div> <div>Controls Accept header.</div> <div>Example Value Schema</div> <div>[{ "date": "2025-07-09", "temperatureC": 0, "temperatureF": 0, "summary": "string" }]</div>	No links

Schemas

Department { id > [...] name > [...] }

Employee { id > [...] name > [...] salary > [...] permanent > [...] department Department > {...} skills > [...] dateOfBirth > [...] }

Skill { id > [...] name > [...] }

WeatherForecast { date > [...] temperatureC > [...] temperatureF > [...] summary > [...] }