# Swagger spec for REST API

# Table of Contents

# Overview

This is a sample spec for Department REST API

# Version information

Version: v1

# Contact information

Contact: Igor Contact Email: @mrigor

# License information

License: test task for EPAM License URL: http://www.testcompany.com/licenses/LICENSE-2.0.html

Terms of service: http://www.testcompany.com/terms-and-conditions

# Tags

- departments: Endpoint for Department specific operations
- employees: Endpoint for Employee specific operations

# Paths

## create new department

```
POST /rest/departments
```

### Description

create new department

### Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| BodyParameter | body | new department | false | Department | |

## Responses

| HTTP Code | Description | Schema |
|---|---|---|
| 200 | successful operation | Department |

## Consumes

- application/json

## Produces

- application/json

## Tags

- departments

# update department

```
PUT /rest/departments
```

## Description

update department

## Parameters

| Type | Name | Description | Required | Schema | Default |
|---|---|---|---|---|---|
| BodyParameter | body | new department | false | Department | |

## Responses

| HTTP Code | Description | Schema |
|---|---|---|
| default | successful operation | No Content |

## Consumes

- application/json

# get all departments

```
GET /rest/departments
```

## Description

get all departments

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | Department |

## Produces

• application/json

## Tags

• departments

# get all employees with information about average salary

```
GET /rest/departments/withAvgSalary
```

## Description

get department's list with information about average salary by each department. If department don't have any employees, then AVG salary will by = 0

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | DepartmentWithAverageSalary array |

## Produces

- application/json

## Tags

- departments

# delete department by id

```
DELETE /rest/departments/{id}
```

## Description

delete department by id

## Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| PathParameter | id | identifier of department | true | integer (int32) | |

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 404 | department with such identifier doesn't exists | No Content |

## Tags

- departments

# get department by id

```
GET /rest/departments/{id}
```

## Description

get department by id

## Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| PathParameter | id | identifier of department | true | integer (int32) | |

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | Employee |
| 404 | department with such identifier doesn't exists | No Content |

## Produces

- application/json

## Tags

- departments

# get employee's list from department

```
GET /rest/departments/{id}/employees
```

## Description

get employee from department

## Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| PathParameter | id | identifier of department | true | integer (int32) | |

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | Employee array |

| HTTP Code | Description | Schema |
|---|---|---|
| 404 | department with such identifier doesn't exists | No Content |

## Produces

- application/json

## Tags

- departments

# create employee

```
POST /rest/employees
```

## Description

create employee

## Parameters

| Type | Name | Description | Required | Schema | Default |
|---|---|---|---|---|---|
| BodyParameter | body | | false | Employee | |

## Responses

| HTTP Code | Description | Schema |
|---|---|---|
| 200 | successful operation | Employee |

## Consumes

- application/json

## Produces

- application/json

## Tags

- employees

# update employee

```
PUT /rest/employees
```

## Description

update employee

## Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| BodyParameter | body | new employee | false | Employee | |

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| default | successful operation | No Content |

## Consumes

- application/json

## Tags

- employees

# get all employees

```
GET /rest/employees
```

## Description

get all employees

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | Employee array |

## Produces

- application/json

## Tags

- employees

# get filtered list of employees

```
GET /rest/employees/filtered
```

## Description

get filtered list of employees by params

## Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| QueryParameter | from | like '2016-01-01' | false | string (date) | |
| QueryParameter | to | like '2016-01-01' | false | string (date) | |
| QueryParameter | departmentid | | false | integer (int32) | |

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | Employee array |

## Produces

- application/json

## Tags

- employees

# delete employee by id

```
DELETE /rest/employees/{id}
```

## Description

delete employee by id

## Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| PathParameter | id | identifier of employee | true | integer (int32) | |

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | Employee |
| 404 | employee with such identifier doesn't exists | No Content |

## Tags

- employees

# get employee by id

```
GET /rest/employees/{id}
```

## Description

get employee by id

## Parameters

| Type | Name | Description | Required | Schema | Default |
|------|------|-------------|----------|--------|---------|
| PathParameter | id | identifier of employee | true | integer (int32) | |

## Responses

| HTTP Code | Description | Schema |
|-----------|-------------|--------|
| 200 | successful operation | Employee |
| 404 | employee with such identifier doesn't exists | No Content |

## Produces

- application/json

## Tags

- employees

# Definitions

## Department

| Name | Description | Required | Schema | Default |
|------|-------------|----------|--------|---------|
| id | | false | integer (int32) | |
| name | | false | string | |

## DepartmentWithAverageSalary

| Name | Description | Required | Schema | Default |
|------|-------------|----------|--------|---------|
| id | | false | integer (int32) | |
| name | | false | string | |
| averageSalary | | false | integer (int32) | |

## Employee

| Name | Description | Required | Schema | Default |
|------|-------------|----------|--------|---------|
| id | | false | integer (int32) | |
| fullName | | false | string | |
| birthDay | | false | string (date) | |

| Name | Description | Required | Schema | Default |
| --- | --- | --- | --- | --- |
| salary | | false | integer (int32) | |
| departmentId | | false | integer (int32) | |