

Interview Assignment

Please complete this small development exercise **on your own** – we would like to go over your solution together during your onsite/skype interview. You are welcome to bring your laptop with you to the interview or publish the code on github/bitbucket/gitlab. If anything is unclear there is no need to contact us for more information – please use your best judgement. You will not be judged on requirements that are unclear. If you are unable to complete the entire exercise, please complete what you can, we can discuss the deltas. The more you complete, the better.

Please create a simple employee skills tracking application that:

- Provides a RESTful (or at least JSON over HTTP) interface
- Is written using any language, but JAVA (with Spring Boot) is preferred
- Contains a simple web UI written using any technology, but preferable a SPA framework such as React or Angular
- Uses one or more databases for persistence; Any database(s) can be used; for simplicity an in-memory database such as H2 or a modern NoSQL engine is acceptable. Be prepared to explain your persistence choice(s) and describe how you architected the data tier to scale

Things to consider:

- Cleaning up the OpenApi Spec / Suggesting any new elements.
- Testing (Unit, Functional, Integration, Automated and Performance)
- Code Quality and structure
- Scalability (Can you solution easily scale to millions of transactions per day?)
- Extensibility (how would you add additional endpoints)
- Modern practices
- Cloud/Container friendly (deployment and DevOps pipeline are some examples)

Implement using the Following API

For all APIs, please include error handling for scenarios with incorrect or incomplete request data.

Copy and paste the attached YAML content to <http://editor.swagger.io/>

prft-interview-assign
ment.yaml

Example:

← → C Not secure editor.swagger.io

Google Chrome isn't your default browser Set as default

Swagger Editor

File Edit Insert Generate Server Generate Client

```
1 openapi: 3.0.1
2 info:
3   description: Perficient Interview Assignment - Employee Skills Application.
4   version: 0.0.1
5   title: Employee Skills Application
6 servers:
7   - url: http://localhost:8080
8     description: Development server
9   - url: https://localhost:8080
10    description: Development server with TLS Profile
11 paths:
12   /employees:
13     get:
14       tags:
15         - employee
16       summary: Get all Perficient employees.
17       operationId: findallEmployees
18       responses:
19         '200':
20           description: Retrieved all Perficient employees.
21           content:
22             '*/*':
23               schema:
24                 type: array
25                 items:
26                   $ref: '#/components/schemas/Employee'
27       headers:
28         X-Total-Count:
29           description: The total count of Perficient employees returned.
30           schema:
31             type: integer
32     post:
33       tags:
34         - employee
35       summary: Create a Perficient employee.
36       operationId: createEmployee
37       requestBody:
38         description: Perficient employee object
39         required: true
40         content:
41           '*/*':
```

Employee Skills Application

0.0.1 OAS3

Perficient Interview Assignment - Employee Skills Application.

Servers

http://localhost:8080 - Development server

employee

GET /employees Get all Perficient employees.

POST /employees Create a Perficient employee.

GET /employees/{employeeId} Find a Perficient employee by ID.

PUT /employees/{employeeId} Update a Perficient employee by ID.

DELETE /employees/{employeeId} Delete a Perficient employee by ID.

skill