# Test task

# For junior Java developer

You must implement an HTTP API server with basic CRUD operations for the Customer entity.

Expected result: link to git repository.

Backend - Spring (better with Spring Boot).

Frontend - not required.

Database - any relational persistent SQL DB (MySQL/Postgress/...).

Authentication - not required.

## Endpoints

### Create customer

POST /api/customers

Content-Type application/json

Body:

fullName: String (2..50 chars including whitespaces)

email: String (2..100 chars, unique, should include exactly one @)

phone: String (6..14 chars, only digits, should start from +, optional field)

Response body:

id: Long

fullName: String

email: String

phone: String

### Read all customers

GET /api/customers

Response body:

List of:

id: Long

fullName: String

email: String

phone: String

### Read customer

GET /api/customers/{id}

Response body:

id: Long

fullName: String

email: String

phone: String

### Update customer

Put or Patch /api/customers/{id}

Content-Type application/json

Body:

id: Long

fullName: String (2..50 chars including whitespaces)

phone: String (6..14 chars, only digits, should start from +)

Response body:

id: Long

fullName: String

email: String

phone: String

### Delete customer

Delete /api/customers/{id}

Just mark a customer as deleted, but leave his data in DB. Related DB column: is\_active.

## Database structure

### customer table

id: bigint

created: bigint

updated: bigint

full\_name: varchar

email: varchar

phone: varchar

is\_active: bool