**Employee payroll details application:**

**Requirement:**

1. **POST API:**

/employee/payroll:

- API should accept the employee salary details as payload

**Request Payload:**

* + Name (Manadatory field)
  + Designation (Manadatory field)
  + project name (Manadatory field)
  + Gross salary (Manadatory field)
  + Tax percentage (Manadatory field)
  + Pension percentage (Manadatory field)
  + Addtional benifit amount (Optional field)

- Store/persist the Above data in Employee\_Details table (you can create the table with the above columns to store data)

**Http Status: 201**

**Response payload:**

Return the payload with all the details it persisted ideally the request data it persisted as above.

1. **GET API:**

/employee/payroll/employees-detail

- API should return all the employees Salary details from the **employee\_Details** table and should

calculating the tax , pension amount additional benefits amount less the tax and pension amount and return the net take home salary details as list of below Json payload.

API Response payload (Return all the employee details as list of Json data) as below

**HttpStatus**: 200

**Response payload(List of all the employees data as Json data):**

* + Name
  + Designation
  + project name
  + Gross salary amount
  + Tax percentage
  + Pension percentage
  + Additional benefit
  + Net salary amount

**Note**:

* All the amounts should be rounded upto 2 decimals if any decimal amounts.
* And all the mandatory input fields should be validated for the input and report error if those are missing, and you can have your own error messages for validation.

**Eg:**

**Given**

Employee salary is £**10000**

**When**

tax=20%, pension amount=10%, additional benefit=£100

**then**

net salary amount is 10000-(pension amount)-(tax amount)+additional benefit = **£7100.00**

**What is required:**

* Create a spring boot application with above two end points and create a executable jar.
* You can use the OpenAPI or similar for swagger interface for the API definition.
* Use Any in-memory DB to create the Employee\_Details table to persist the post API payload to store the Employee salary details.
* Use any Java/Spring DB framework for persisting data and retrieval of data.
* Health end point to check the application health, if the application is up and running.

**Note**: Github repo link will be shared with you for pushing the application code once you are ready with code

**what we expect to see:**

* Application design
* Validations and Clean code
* Crud operation for persisting the data
* Testing approach, Test coverage / TDD approach unit tests and integration tests.
* Jacoco integration for test coverage.
* jdk8 / jdk11 feature used wherever required like stream and other features.
* Design patterns / SOLID principles.