|  |
| --- |
| **Task preview - Java | Spring Boot, JPA, REST API | Blog app** |
|  |
| **Blog Engine powered by Spring Boot** |
| **Introduction** |
| This project is a simple REST application for serving a blog. It uses Spring Boot, Spring Data JPA and some other helper libraries. |
| **Problem Statement** |
| Currently, application can only serve blog post details by executing GET request at /posts/{id}, where {id} is a post identifier. |
| Your task is to add comments feature. The application should be able to serve two new kinds of requests: |
| POST at /posts/{id}/comments which should: |
|  |
| Save a new comment with current date and time for post with passed {id} |
| Return 201 Created if comment is created successfully for post with passed {id} |
| Return 404 Not Found if post with passed {id} does not exist |
| GET at /posts/{id}/comments which should: |
|  |
| Return all comments sorted by creation date in descending order for a post with passed {id} |
| Return empty list if a post with passed {id} does not exists or when it does not contain any comments |
| There is a service class CommentService that you should implement. See contents of that class to see detailed requirements. For each method, there is a javadoc comment that describes expected behavior. |
| There are some tests that are currently failing and your solution should satisfy these tests requirements. |
| Remember that provided tests verify the functional correctness, however, you should remember to fulfill all best practices as well. |
| **Hints** |
| Feel free to create new and modify existing files. You can add some new dependencies if you will, to do so just modify appropriate pom.xml. **Do not change unit tests**, however. |
| Please follow conventions used in this project. |