**Benchmark Assessment:**

1. Create a spring webservice application on start.spring.io(10 min)
   1. Rest Services
   2. Validations and Auditing
   3. Mysql Database
   4. Any other program structures you may need
2. For the below table structure create JPA entities and repository ( 20 min )
3. /\*
4. CoderPad provides a basic SQL sandbox with the following schema.
5. You can also use commands like `show tables` and `desc employees`
6. employees                             projects
7. +---------------+---------+           +---------------+---------+
8. | id            | int     |<----+  +->| id            | int     |
9. | first\_name    | varchar |     |  |  | title         | varchar |
10. | last\_name     | varchar |     |  |  | start\_date    | date    |
11. | salary        | int     |     |  |  | end\_date      | date    |
12. | department\_id | int     |--+  |  |  | budget        | int     |
13. +---------------+---------+  |  |  |  +---------------+---------+
14. |  |  |
15. departments                  |  |  |  employees\_projects
16. +---------------+---------+  |  |  |  +---------------+---------+
17. | id            | int     |<-+  |  +--| project\_id    | int     |
18. | name          | varchar |     +-----| employee\_id   | int     |
19. +---------------+---------+           +---------------+---------+
20. \*/

3. Create a department CRUD service (20 min )

4. Create an Employee Service CRUD service. Only the department should be assigned, with no projects assigned. Ensure proper validations and integrity (20 min)

4. Design and create REST services for assigning projects to Employees (30 min)

5. Add Validation to the first\_name and salary  fields in the Employee POJO. first\_name should not be empty and salary should be a non-negative number (2 decimals allowed).

Do later.

6. Add a Global Exception handler to the REST application. It should handle the above exceptions as well as any generic exception thrown. (20 mins)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*