# Spring Boot Project

# Submission Date:28/09/2021

# Total Marks: 50 marks

# (45 for the use cases + 5 for coding standards and naming conventions)

Incorporate the following functionalities using Spring Boot in the Banking Application.

1. Create an application with Rest services using Spring Boot that has the following functionalities for the Customer.
   1. Customer Login validation
   2. Password Encryption
   3. Customer Forgot Password
   4. Customer Update Password
   5. Customer can view his/her balance.
   6. Customer can view his/her account details.
   7. Customer can view the transactions from start date to end date.
   8. Customer can upload his photo.

(Reference : <https://www.callicoder.com/spring-boot-file-upload-download-rest-api-example/>)

* 1. Customer can download his statement from a given start date and end date.

1. Create an application that has the following functionalities for the admin.
   1. Admin creates a Customer and an Account
   2. Admin can delete an account
   3. Admin can update address ,email and phone number of the customer.
   4. Admin can add all Customer details from a csv file to a database.

(Use Spring Batch)

3 Create an application that has the following functionalities for the Manager.

1. Manager can view statement of any customer. Calls the service provided in the Customer application.
2. Manager Can View Account details based on the account no.
3. Implement Spring Security using JWT for above.

Note:

1. Use the following dependencies.
   1. JPA Starter
   2. MySQL
   3. Web
   4. Dev Tools
   5. Lombok
   6. Swagger for UI.
   7. Others depending on the requirements.

Sometimes we all feel bad in lives, and I am typing this message at 2:30 AM. But how we carry our life forward, makes us who we are as a human. We often fight, get angry, don’t resolve things, cry, harm and destroy a lot. But looking to get the best stuff out of these situations is so drastically overwhelming. It may have a huge mental toll.

AND, this is the reason which led me to get attracted towards your thoughts, which are more “generous” and “humble” towards harsh/huge approaches.

In short, in this fast-moving world, I like your “Let it be”.

If ( isAnythingMisunderstoodIncorrectlyAboutKJByPK()){

correctPK(); //it’s always good to remove misconceptions}

else { return whatDoKJthinkAboutPK() ;}