1) **We prepare our Spring Boot App for Production Usage:**
   a) Since Heroku supports PostGreSQL, we need to modify our code to use PostGreSQL for Production.

      Implementation: To handle the DB2 database for Local and PostGreSQL for production we can make use of the concept of profiles in Spring Boot, using which we can provide a different set of configuration properties for each profile.

   Back-End URL Deployed to Heroku:

      https://limitless-brook-45203.herokuapp.com/ |
      https://git.heroku.com/limitless-brook-45203.git

      For production grade environment we are not going to store the database credentials inside the source code and take the risk of exposing it to the public for that reason we need to remove the URL, username and password for spring.datasource properties and we will inject dynamically at Runtime using Heroku.

   b) Replace the Hard Coded URLs with properties (localhost with Env Variables)
2) **We will Deploy our App to Heroku Cloud Platform:**
   a) Install the Heroku CLI
   b) Learn some Heroku Concepts when deploying our backend Application
3) **We prepare our Angular App for Production:**
   a) Manage Hard Coded URLs through Env Variables
   b) Change the starter scripts in the package.json file
4) **Deploy the Angular App to Heroku Cloud:**

   Front-End URL Deployed to Heroku:

      https://obscure-dawn-87005.herokuapp.com/ |
      https://git.heroku.com/obscure-dawn-87005.git

   For more details refer: https://itnext.io/how-to-deploy-angular-application-to-heroku-1d56e09c5147