**Airport App Case Study**

Refer the website <https://ourairports.com/data/> & use the csv data provided there to create your own airports website to show airports and to allow people to leave comments on the site. Download the data in csv format from <https://ourairports.com/data/> and load the csv data into a RDBMS.

Week - 1

* Create tables, indexes and add proper constraints. Use JDBC for command and query.
* Use java 8 above features.
* Follow proper OOPS concepts (Encapsulation, Abstraction, Inheritance, Polymorphism).
* Classes should follow SOLID Principles
* Proper Exception Handling
* Write Java Generic classes where applicable.
* Write Junit test cases with Mockito wherever applicable.Use Maven to build the code. Use proper plugins to check the code quality at the time of build.
* **Write BDD tests to test the API without mocking and using in memory DB like H2 and run it as part of build.**

Week - 2

* Since potentially multiple users will be using this application same time classes should be thread safe. Refactor the code to make the required classes thread safe and use java concurrency package executor frameworks where you may process something concurrently.
* Refactor the code to use proper collections as per the read write requirement.

Week - 3

* Run the application, generate heap dump & thread dump and analyse the dumps to pinpoint the problem areas.

Week – 4

* Refactor the code to use applicable GOF Java design patterns in your code.

Week – 5

* Use JDBC connection pooling. All the connection pooling properties should be configurable.
* Handle transaction (ACID) properly.

Week – 6

* Refactor the project into a spring boot application. Replace JDBC with spring boot JPA with hibernate.