Machine generated alternative text:
res 

Programming Test

Please write two Java processes:

1. Application Process (Server)
   * RestService

Rest service will expose RestAPI using Spring's embedded tomcat(spring boot)

API:

* Add triangle entity without hypotenuse (json):

Should include the length of 2 edges of the triangle

* StorageService

Storage service will support two configurable persistency options: In-Memory \ File.

Please implement in-memory storage with a preparation for file persistency.

API:

* Add entity (id, data)
* Update entity (id, data, state(pre-calc/post-calc)
* Get entities (by id, all, state(pre-calc/post-calc)
* Get entities count (by id , all, state(pre-calc/post-calc)

* AlgoService (async bean)
  + Algo service will execute algorithm cycle every configurable interval.
  + Each cycle will retrieve triangle entities without hypotenuse from the storage service, calculate the hypotenuse for each one and update the triangle data in the storage service.

* SystemLoggerService
  + The system logger will register to StorageService entity creation and changes and write logs accordingly.
  + The system logger will write the total entities count in the system every configurable interval.

1. Data Injector Process (client)
   * Data Injector will randomly generate two legs of right triangles and send the data to the application using Rest service invocation.

* Please use the following frameworks and tools: Maven, Spring.
* You're free to use any other familiar frameworks & tools.
* Modular, Generic & Clean code will be appreciated.
* No GUI is needed.
* Make sure to write logs to see the process action.