**Invenco By GVR (IS360) Assignment Overview**

You will build a simple Customer Order Management System with the following core features:

* Create Customers, Products, and Orders.
* Apply business rules (e.g., stock management, premium order flagging).
* Implement RESTful APIs.
* Use Hibernate to persist data.
* Implement a simple caching mechanism for products.

**You are provided with**

* A starter Spring Boot project (customer-order-app.zip).
* Basic project structure with empty classes and interfaces.
* In-memory H2 database configuration.
* Required Maven dependencies included.
* A readme guide

****

**Task Requirements**

**1. Entity Modeling**

* Create 3 entities: Customer, Product, and Order.
* Use JPA annotations for relationships (@OneToMany, @ManyToOne, etc.).

**2. REST APIs**

Implement the following endpoints:

* POST /customers — Create a new customer.
* POST /products — Create a new product.
* POST /orders — Create a new order.
* GET /orders/{customerId} — Fetch all orders for a customer.

**3. Business Rules**

* Check product stock before placing an order.
* Deduct ordered quantity from stock when an order is placed.
* Rollback the entire order if any product is out of stock.
* If an order total exceeds $500, mark it as a premium order.

**4. Validation**

* Customer email must be unique.
* Product name must not be blank.

**5. Caching**

* Cache the list of all products for 10 minutes.
* Invalidate the cache if a product is created or updated.

**6. Error Handling**

* Provide meaningful HTTP error responses (e.g., 400, 409).

**Deliverables**

* The complete, working Spring Boot project as a .zip file or Git repository link.
* A short README.md explaining how to run your project (commands if necessary).
* A brief document explaining any assumptions or improvements