Taco Loco Java Challenge:

Taco Loco, a quick service fleet of taco trucks, is building a new mobile app to enable their customers to place orders for their delicious tacos. They've hired Detroit Labs to build the backend services to power this app.

As a backend developer, you've been tasked with creating a web service to calculate order totals. The service will take as input the items and quantities ordered, and respond with the order total.

Taco Loco's menu consists of the the following items:

Veggie Taco, $2.50 ea.

Chicken or Beef Taco, $3.00 ea.

Chorizo Taco, $3.50 ea.

If a customer orders 4 or more tacos, then a 20% discount should be applied to the entire order.

Taco Loco wants you to define the data format for the request and response, and prefers using JSON.

You should build the service using Java, and we encourage the use of existing libraries and frameworks, like Spring Boot/Spring MVC (or similar), to ease development. Security is not required.

What artifacts do I need to submit?

• Quick start guide that details how to build and publish the service, e.g. as a WAR, executable JAR, etc.

FAQ

How long should I/may I spend on this exercise?

You may spend the amount of time necessary to submit a solution that is a true representation of your abilities. Please add in an estimation of how much time you spent when you submit your solution back to your contact.

If you normally use a project template when creating new applications, feel free to use it for this exercise.

How will my submission be evaluated?

At Detroit Labs, we care about:

• **Code quality, style, and readability**. Code should be of excellent quality and should follow standard style guides for the language/platform of choice. We value clarity, simplicity, and maintainability. You should be proud of the code you send us for this exercise.

• **Architecture and design**. The implementation mechanics, libraries, databases, and architecture(s) used should be suitable for this exercise. Avoid using unnecessary tools that add more complexity than value.

• **Error and edge case handling**. Identify and handle any errors and edge cases as you would for a production application.

• **Appropriate documentation**. The application should contain the level of docu- mentation and commentary you would use in a production application. Your commit history should be clean, with each commit representing a solid piece of work in your application.

• **Completeness and correctness of features**. The features implemented should be complete, and accurate/correct in line with the asks of the exercise.

• **Platform familiarity**. Your solution should use components and patterns in line with platform and/or framework guidelines.

• **UX & Design**. We're assessing your development skills, not your design skills, so don't spend time making fancy graphics. We only ask that the user interface be clean and clear and follow relevant platform guidelines.

Submitting your completed solution:

Share your Github link and an estimation of the time your spent on your solution to your contact at Detroit Labs when you have completed the assignment.

Once we have reviewed your submission we will be in contact regarding the outcome and next steps. Thank you, and Good Luck! We can't wait to see what you come up with!