H-E-B DIGITAL

FOUNDATIONAL TECHNOLOGY – PLATFORM ENGINEERING

Technical Problem Review

The first part of your Engineering interview includes producing some sample code for you to review with the interviewer during the Technical Problem Review interview. Please choose from any of the problems outlined below and code your solution in the language of your choice.

Prior to starting your project, your recruiter will schedule a 15-minute Q&A session via Zoom with the interviewer so that you have the opportunity to get clarification on the problems outlined and ask any questions you may have. After your 15-minute Q&A, you will begin working on the project, of your choice, over the next few days. The 1-hour Problem Review and code review will be scheduled via Zoom a few days after your Q&A session. Be sure that you have provided your GitHub username to your recruiter. You will receive an invite to a code repository on the HEB-Recruiting GitHub account at https://github.com/heb-recruiting where you will submit your code for review. Please note you are not expected to spend more than 2-hours of your time coding your solution.

How to prepare:

- Ask questions during both the Q&A and the Problem Review! The interviewer is not here to trick you, they are here to help you and are trying to get a sense of what it would be like working with you on a problem collaboratively.
- Make sure to review your code and are able to walk someone through your decisions.
- If you use any 3rd party libraries, make sure you know general details of their underlying implementation and complexities.
- The reviewer may ask you about test cases you haven't considered, or introduce new complexities not included originally, think out loud as you work through how your code may be adjusted to handle them.
- Have your code open in your IDE so you can share your screen.

- Come to the Problem Review prepared to discuss the algorithms, data structures, design patterns, and the scalability of the solution presented.

Problem Statements

Select one (1) of the problems outlined below to work a solution for in the coding language of your choice.

SHOPPING CART PROBLEM

PRICE COMPARISON TOOL PROBLEM

Shopping Cart Problem

We sell a lot of groceries, so we fill and empty tons of virtual shopping carts. To sell groceries online we need a few features the business has asked for. Below are four shopping cart features the business has requested.

This will be a multi-part question and the expectation is you likely won't complete it in the 2-hour time allotment. Be sure that you have provided your GitHub username to your recruiter. You will receive an invite to a code repository on the HEB-Recruiting GitHub account at https://github.com/heb-recruiting where you will submit your code for review.

If you find any issues with the files we have sent or have other questions don't hesitate to reach out to your recruiter or the H-E-B Partner that you met with during the 15-min Q&A session.

Assumptions

- All items are single quantities
- The shopping list is static for each feature section
- The coupon list is a separate JSON file
- In feature 4, the final price of an item cannot be negative
- The tax rate is .0825(8.25%)
- All prices are in USD

Feature 1

Calculate the total of the given shopping cart. Print out the Grand Total of the shopping cart.

Feature 2

Calculate the total of a given shopping cart and include the sales tax amount from the subtotal of the cart. Print out the following:

- Subtotal
- Tax Total
- Grand Total

Feature 3

Not all are items are taxable. Re-calculate the tax total based on the boolean field in the JSON file "isTaxable". If "isTaxable" is true then calculate sales tax for that item, if it is false skip the sales tax calculation. Print out the following:

- Subtotal
- Tax Total
- Grand Total

Feature 4

The business would also like to support coupons for certain items in a cart. Extend the solution by applying coupons contained in the coupon.json file.

Print out the following:

- Subtotal
- Tax Total
- Grand Total

Price Comparison Tool Problem

This will be a multi-part question and the expectation is you likely won't complete it in the 2-hour time allotment. Be sure that you have provided your GitHub username to your recruiter. You will receive an invite to a code repository on the HEB-Recruiting GitHub account at https://github.com/heb-recruiting where you will submit your code for review.

If you find any issues with the files we have sent or have other questions don't hesitate to reach out to your recruiter or the H-E-B Partner that you met with during the 15-min Q&A session.

Build a price comparison tool that will get pricing and availability data from these sources:

URL	https://appedia.heb-platform-interview.hebdigital-prd.com/api/v1/itemdata?upc=101

Response	{
	"price": "$4.77",
	"stock": 7
	}

Description	Price is returned as a string Availability('stock') returned as an integer value, 0 meaning out of stock.

URL	https://micromazon.heb-platform-interview.hebdigital-prd.com/101/productinfo

Response	{
	"available": true,
	"price": 5.67
	}

Description	Price is returned as a double Availability('available') returned as a boolean value

URL	https://googdit.heb-platform-interview.hebdigital-prd.com/101

Response	{
	"a": [
	{
	"l": 8839,
	"q": 4
	},
	{
	"l": 1292,
	"q": 0
	}
	],
	"p": 478000000
	}

Description	Price is returned as microcents (ie 234000000 == $2.34) Availability('a') is returned as an array of objects with quantity('q') available at location('l')

Requirements

- Return the URL that has the lowest price and has the item in stock at any location.
- Make it easy to swap item UPC.
- Make it easy to add additional merchant URLs to query for price comparison.
- Write unit/integration tests.