Typically takes 30 minutes/1 hour total:

A quick coding question:

Write a method that returns the "pivot" index of a list of integers. We define the pivot index as the index where the sum of the numbers on the left is equal to the sum of the numbers on the right. Given [1, 4, 6, 3, 2], the method should return 2, since the sum of the numbers to the left of index 2 is equal to the sum of numbers to the right of index 2 (1 + 4 = 3 + 2). If no such index exists, it should return -1. If there are multiple pivots, you can return the left-most pivot.

You can write the method in any language. Make sure that the method:

   • runs successfully

   • handles all edge cases

   • is as efficient as you can make it!

A successful answer will fulfill the above criteria.

And a longer, design question:

Let's design a small part of Gumroad. Gumroad helps people sell their products online. As part of that we charge the buyers' credit cards and deposit money to the sellers' accounts. Here's a simplified version of our data model:

Seller <1----\*> Product <1----\*> Purchase

Each seller can have multiple products, and each product can have multiple purchases. Every purchase should increase the seller's balance. The seller can at any time refund a purchase, which should decrease his/her balance.

We want to pay the sellers every other week for all their sales up to a week before that. For instance we would pay the sellers on Friday the 10th for all their sales up to Friday the 3rd. We want to have the flexibility to change this schedule; we might want to pay the sellers once a week instead of every other week.

How would you design the data model that would support these requirements? Feel free to use a drawing tool to sketch out your design but please explain the rationale behind your design and how it would support:

- purchases and refunds increasing and decreasing the balances

- rolling payouts as described above

Imagine the number of sellers and products will be in the millions, and the number of purchases will be in the hundreds of millions. Describe the indexes that you will add to the different tables/collections.

We use ActiveRecord and MySQL, but your answer doesn't have to.