Flask Hiring Test

Why this test?

We build applications for the web, so understanding how web applications work, is a pre-requisite for any new engineer. Flask is one of the simplest and well written Python based web application frameworks and can easily be learned in a few hours.

If you have never worked on web applications before, this test will also help you evaluate whether this is something you are skilled at or can easily learn doing.

Purchase/Sales Management Web Application

The goal is to create a web application using Flask framework to manage a small petty shop where there are items, purchases and sales.

The application should cover the following sample scenario:

This is just a sample scenario and your submission need not follow the same terminologies.

There is a company called “Namma Kadai”. Its initial cash balance is Rs. 1000.

There are five items.

- Pen - Rs.5/each
- Pencil - Rs.2/each
- Eraser - Rs.1/each
- Sharpener - Rs.2/each
- Geometry box - Rs.10/each

Namma Kadai purchases 10 Geometry boxes (10 * 10 = Rs.100). With this its cash balance becomes (1000 - 100 = Rs.900).

Namma Kadai sells 10 boxes for 15 each and earns (15 * 10 = Rs.150). With this, its cash balance becomes (900 + 150 = Rs.1050)

At any point in time, Namma Kadai should be able to add new items, make new purchases and make new sales and see its running cash balance.

The application should cover the following functionalities:

Database Tables:

- Company (`company_name, cash_balance`)
- Item (`item_id, item_name`)
- Purchase (`purchase_id`, `timestamp`, `item_id`, `qty`, `rate`, `amount`)
- Sales (`sales_id, timestamp, item_id,qty,rate,amount)`

Note:

1. Primary keys can be text / varchar
2. “Rate” can be filled as any value while making a purchase or sales. Corresponding amount (rate * qty) should be updated in the cash balance of the company.

Views:

- Add/Edit/View Item
- Add/View Purchase
- Add/View Sales

Report:

- Show current cash balance.

Bonus

- **Add the “qty” column to the “Item” table. Make it 0 initially. On every “Purchase” add the corresponding qty. On every “Sales” reduce the corresponding “qty”. Show a report of Item and its current qty.**

Use Cases:

- Create 3/4 Items
- Create 3/4 Purchase
- Create 3/4 Sales
- Show the updated Cash Balance
- If the Bonus task is attempted, Show the updated Item quantities.