Krushna joshi

Walmart Advanced Software Engineering Job Simulation

Introduction

This report provides an overview of the **Walmart Advanced Software Engineering Job Simulation**, which involved solving complex technical challenges for various teams at Walmart. The simulation required applying **software engineering principles, data structures, algorithms, and system design** to develop efficient and scalable solutions. Key tasks included implementing an **optimized heap data structure in Java** for the shipping department, designing a **UML class diagram** for a data processor with multiple operating modes, and creating an **Entity-Relationship (ER) diagram** to design a structured database for Walmart's pet department. This experience enhanced my **problem-solving abilities, software development expertise, and analytical thinking**, preparing me for real-world challenges in the IT industry.

Implementation

Task 1: Advanced Data Structures

The task is to implement a novel data structure - the project lead is calling it a power of two max heap. The rest of my team is doing their best to come up with a better name. The requirements of the data structure are as follows:

- The heap must satisfy the heap property.
- Every parent node in the heap must have 2^x children.
- The value of x must be a parameter of the heap's constructor.
- The heap must implement an insert method.
- The heap must implement a pop max method.
- The heap must be implemented in Java.
- The heap must be performant.
- You must use a more descriptive variable name than x in your implementation.

Task 2: Software Architecture

The task is to draft a UML class diagram describing the data processors for a pipeline. The component responsible for reading in input data is being designed by another engineer, so I only need to worry about what happens to the data when it reaches your processor. i have assume three classes already exist:

Datapoint: this class represents both raw and processed data points. Any time data moves between methods I have used this class as an abstraction.
ModeIdentifier: an enum used to identify a processor mode.
DatabaseIdentifier: an enum used to identify a database connection.

Here are the requirements for design:

- The processor must implement a configure method that accepts a ModeIdentifier and a DatabaseIdentifier as parameters.
  - This method is called to change the operating mode of the processor, and/or select the current database.
- The processor must be able to change between the following modes:
  - Dump mode: simply drops the data.
  - Passthrough mode: inserts the data into the currently configured database.
  - Validate mode: validates the data, then inserts it (both operations are carried out on the currently configured database).
- The processor must be able to swap between the following databases. Each database will require a different implementation to insert and validate data
  - Postgres.
  - Redis.
  - Elastic.
- The processor must implement a process method that accepts a DataPoint as a parameter.
  - This method will have different behavior depending on the currently configured mode and database.

Task 3:Relational Database Design

The task is to draft a UML class diagram describing the data processors for a pipeline. The component responsible for reading in input data is being designed by another engineer, so I only need to worry about what happens to the data when it reaches your processor.

I have assumed that three classes already exist:

1. Datapoint: this class represents both raw and processed data points. Any time data moves between methods you may use this class as an abstraction.
2. ModeIdentifier: an enum used to identify a processor mode.
3. DatabaseIdentifier: an enum used to identify a database connection.

Your task is to draft an ERD for an appropriately normalized relational database that satisfies these requirements:

- The database should store information related to the following products.
  - Pet food, which has a name, manufacturer, weight, flavor, and target health condition.
  - Pet toys, which have an associated material, name, manufacturer, and durability.
  - Pet apparel, which has a color, manufacturer, size, name, and specific care instructions.

- Each product should be associated with one or more animals.
- Each product should be associated with a manufacturer.
- The database should track customers and their transactions.
  - It should store customer names and email addresses.
  - Customers can make transactions to purchase one or more products.
  - Each transaction should store the date and the products involved.

- The database should track shipments to various Walmart locations.
  - Each location should be represented by a name and a zip code.
  - A shipment is recorded as an origin, a destination, and a collection of products, each with an associated quantity.

Task 4: Data Munging

Part 1: Get the data

First, I need to get your hands on the relevant data. The shipping department has been kind enough to provide you with a repository containing all of their spreadsheets, as well as a copy of the sqlite database. First, fork and clone the repository at: https://github.com/theforage/forage-walmart-task-4

Part 2: Populate the database

My task was to insert all of the data contained in the provided spreadsheets into the SQLite database. I had written a Python script which:

- Read each row from the spreadsheets.
- Extracts the relevant data.
- Munges it into a format that fits the database schema.
- Inserts the data into the database.

Spreadsheet 0 is self contained and can simply be inserted into the database, but spreadsheets 1 and 2 are dependent on one another. Spreadsheet 1 contains a single product per row. I needed to combine each row based on its shipping identifier, determine the quantity of goods in the shipment, and add a new row to the database for each product in the shipment. The origin and destination for each shipment in spreadsheet 1 are contained in spreadsheet 2.i have assumed that all the given data is valid - product names are always spelled the same way, quantities are positive, etc

solutions :

Task 1 : Advanced Data Structures solution

Task 2 : Software Architecture

Task 3: :Relational Database Design

Task 4 :: Data Munging

Conclusion

After completion of these program i gain experience in these skills:

- Java
- Data Structures Optimization
- Software Architecture
- UML
- SQLRelational Database Design
- Python
- Data Munging

Certificate of Walmart Advanced Software Engineering Job Simulation

Certificate