

# Groceries Express! Drone Delivery

CS 4400: Introduction to Database Systems  
Mini-Exam #1 (Fall 2021 Semester) Database Description

## Database Description & Schema

**This is a description of the scenario that defines why the database has been built.**

*This database describes a system used to monitor deliveries of grocery items to customers. Customers will place orders with different stores using the system. The system will coordinate with grocery stores to identify the items being ordered and then arrange for a drone to deliver the items to the customer. On delivery, the store will be paid electronically by the customer.*

### Users

- *table **users** = (uname, first\_name, last\_name, address, birthdate)*

The system will support various types of users, including customers and employees of the grocery stores. All users must be either customers and/or employees. All users will have distinct usernames when accessing the system. The system will also maintain the first name, last name, address, and birthdate of each user.

### Stores

- *table **stores** = (storeID, sname, revenue, manager)*

Each store has a distinct identifier, and a (possibly duplicated) longer name. Each store also keeps track of the revenue it has earned from delivering orders successfully. Stores are supported by various types of employees. Employees can support the company in different roles – for example, floor workers or drone pilots. There are other employee roles such as financial data analysts or logistical coordinators, but we won't keep track of those other roles explicitly in our system.

### Employees

- *table **employees** = (uname, taxID, hired, service, salary)*

The system must track the unique tax identifier (e.g., Social Security Number for some people) for each employee for legal purposes. The system will also keep track of the date each employee was hired, along with the number of months that they have worked (i.e., time in service) at the company. The system tracks these values separately because employees sometimes take a leave of absence to train, attend conferences, or support the family (e.g., maternity leave). The system must also keep track of the salary for each employee.

### Floor Workers

- *table **floor\_workers** = (uname)*
- *table **employed\_workers** = (storeID, uname)*

Floor workers are the people you often encounter in grocery stores stocking shelves; serving in the deli, bakery, or seafood section; or acting as a cashier. Floor workers are responsible for making sure that the stores are ready to provide items as needed. The identifiers will be unique, though names might be shared. Each floor worker must be employed by at least one store, though an employee can also work at multiple stores as part of a "time-flex" plan. Each store can employ multiple floor workers.

Each store must also have one floor worker to act as the overall manager for that store. Managing a store implies that you must also be employed by that store. An employee may be the manager for at most one store. It's too much work to manage more than one store at the same time.

### Drone Pilots

- *table **drone\_pilots** = (uname, licenseID, experience)*

Some employees are drone pilots. Drone pilots control the drones as they carry groceries back and forth between the stores and the customer's homes. Being a drone pilot is unique and time-consuming role, and an employee is not allowed to be a drone pilot and floor worker at the same time. What's most important is that they pilot a drone that is contractually obligated to serve a specific store.

Each drone pilot must have a valid license to signify that they have received the proper training to operate the drone safely. Each license will have a unique ID for tracking purposes. Flight skills also tend to improve with experience, so the system must also keep track of the number of successful deliveries for each pilot.

### Drones

- *table **drones** = (storeID, droneTag, capacity, remaining\_trips, pilot)*

Stores can purchase many drones to deliver orders to customers in a timely manner. Each store will have an identifier and a name. Each drone pilot can control at most one drone at a time (i.e., controlling two or more is unsafe). It's unsafe for a human pilot to control more than one drone at a time. Each drone has been purchased/sponsored by a single store, and is used to deliver orders for that store alone. A drone must be identified relative to the specific store that it serves.

Drones need to be taken out of service for maintenance after making a certain number of trips, where a trip is equivalent to delivering one order. The system must keep track of the number of trips that a drone has remaining before it needs maintenance. Each drone has a limited capacity to carry orders up to a maximum weight. A drone must be controlled by one pilot at all times. Having multiple pilots for a single drone would eventually lead to conflicts and crashes.

### Orders & Order Lines

- *table **orders** = (orderID, sold\_on, purchased\_by, carrier\_store, carrier\_tag)*
- *table **order\_lines** = (orderID, barcode, price, quantity)*
- *table **items** = (barcode, iname, weight)*

Stores offer lots of different items, where each item has a 'universally' identifying barcode, a name, and a weight measured in an even (i.e., integer) number of pounds. Customers who wish to purchase items can place an order. Also, a specific item might be offered by many different stores, but our goal is simply to ensure that each item identified by a specific barcode has a consistent name and weight for tracking purposes.

An order must be requested by a specific customer and must also be assigned to a specific drone for eventual delivery. Each order will have an identifying receipt number, and a record of the date on which the order was placed. An order will consist of one or more lines, where each line represents a certain quantity (i.e., one or more) of an item being ordered at a given unit price.

## Customers

- *table customers = (uname, rating, credit)*

Customers are allowed to place multiple orders concurrently. We must track a customer's rating as an integer from one (1) to five (5), where a higher number indicates that the customer has been more reliable in ordering items over time. We must track the customer's credit as the number of dollars that they have to request orders.

A customer's credit must always be greater than or equal to the total cost of all of the orders for which they are currently waiting. A new order cannot be placed unless the customer requesting the order has enough credit to cover all of the customer's existing and new orders.

## Other Systems Constraints & Capabilities

The system must be able to calculate and display the cost of an order as the total cost of each line, which is the cost of the item as listed on that order multiplied by the quantity purchased. The system must also be able to calculate and display the total cost for all of the outstanding orders for each customer.

The system must be able to calculate and display the weight of an order as the total of the weight of each line, which is the weight of the individual item multiplied by the quantity purchased. The system must be able to calculate and display the total weight (i.e., payload) for all of the orders being delivered by each drone. A new order cannot be placed unless the drone assigned to deliver the order has enough lifting capacity to carry its current orders along with the new order. The system must be able to calculate and display the incoming revenue for each store as the total cost of all orders currently being delivered by drones on behalf of that store.

Each item can be listed at most once on a given order, but the quantity for that item can be one or more. The price (and the quantity) for an item can vary between different orders. An order must be delivered in its entirety by a single drone - it cannot be split across multiple drones.