

## **Week 6 Lab Activity**

David Abeiku Saah

Computer Science Department

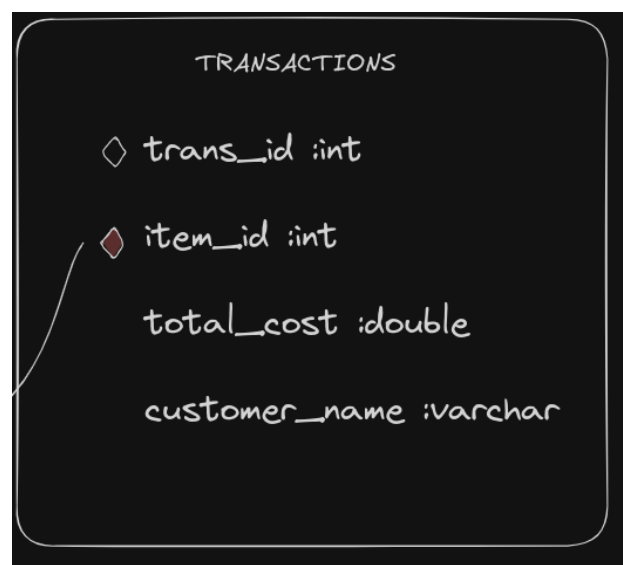
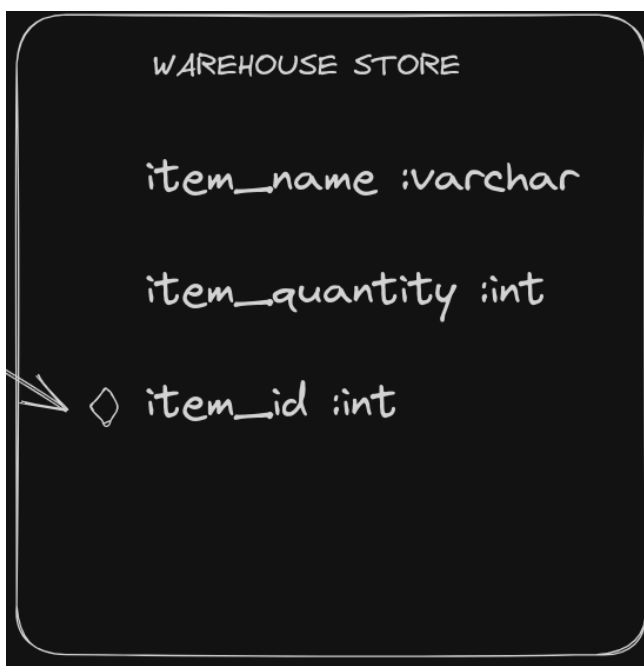
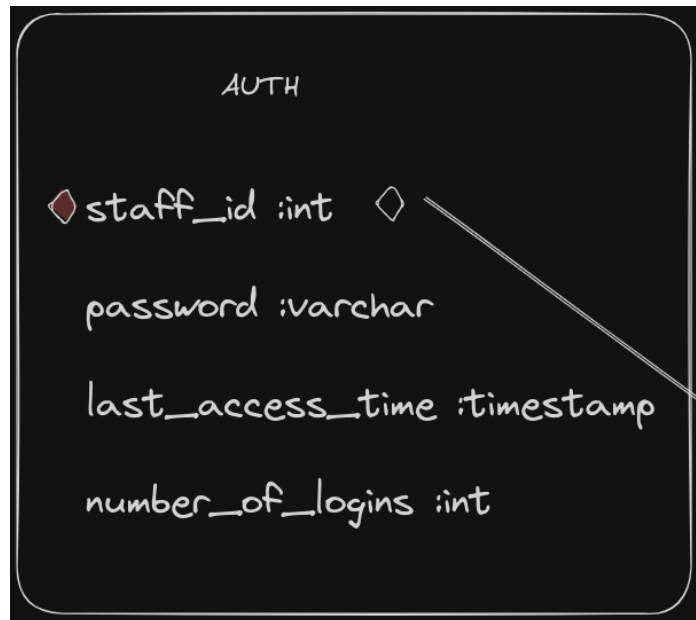
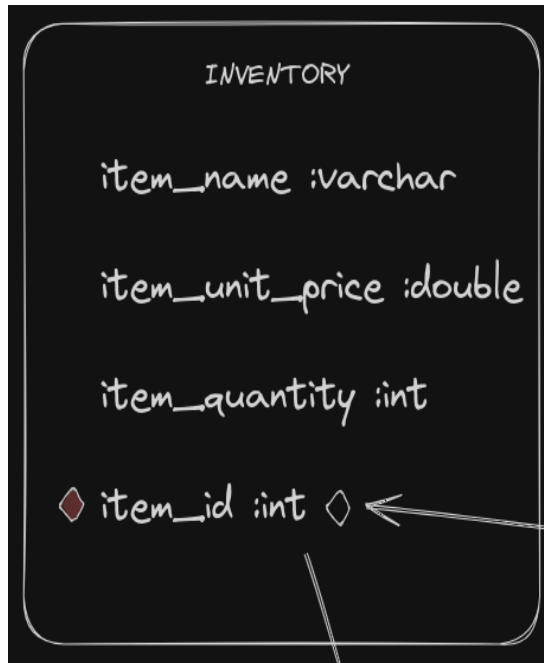
CS323: Database Management / Database Systems

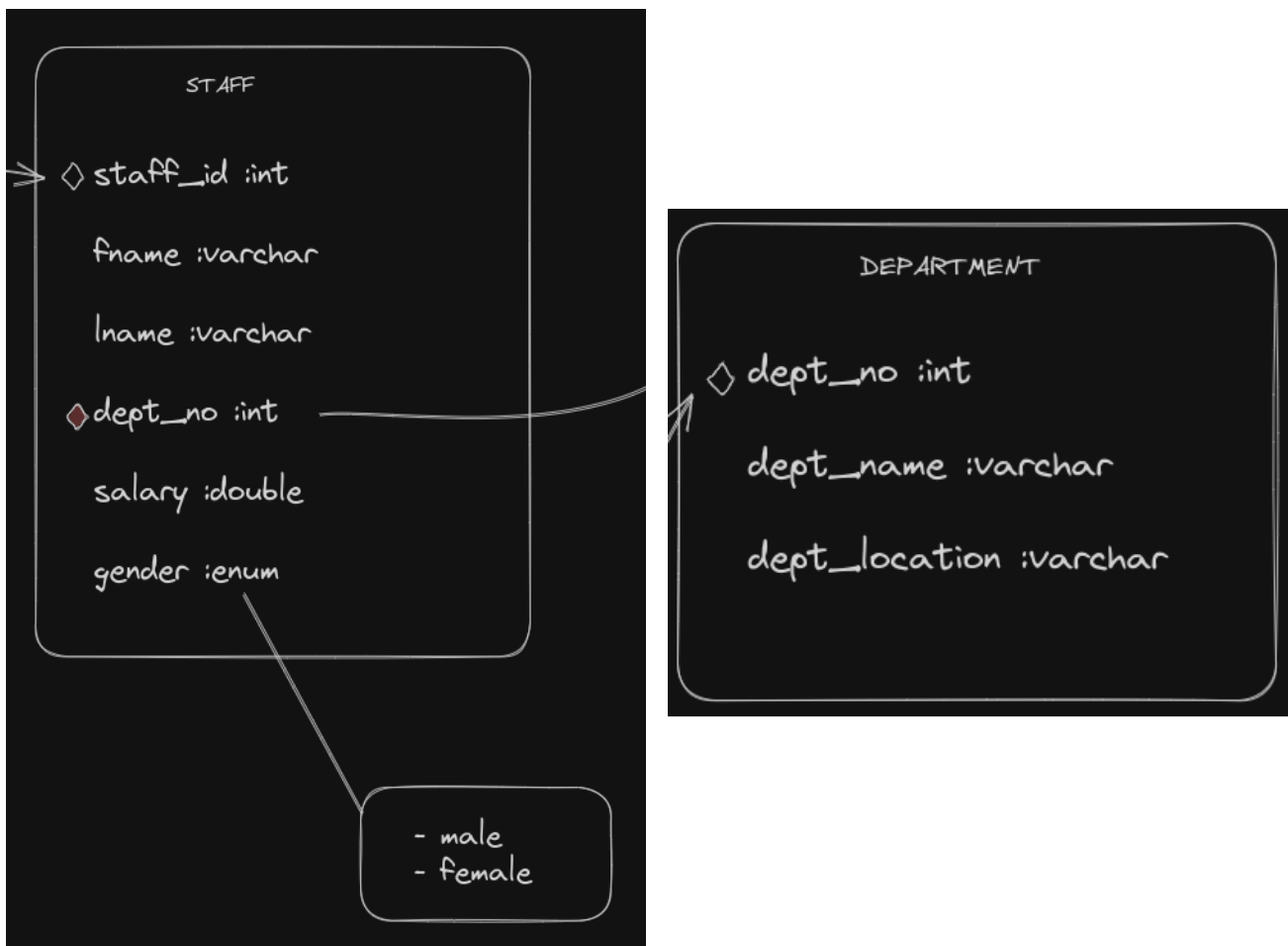
Mr Stephane Nwolley

September 29, 2023

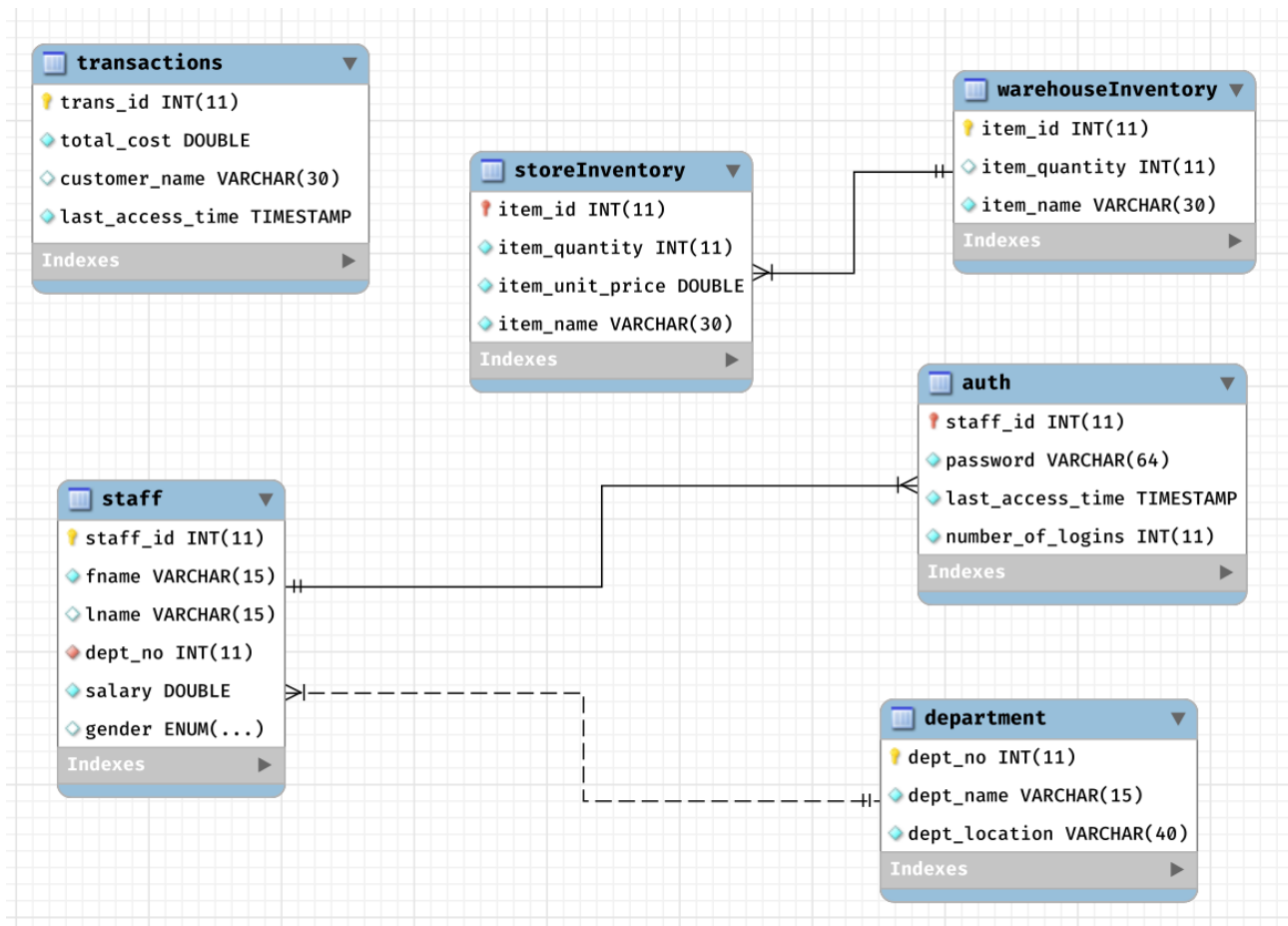
Assigned database: **Store Inventory Management System.**

**List of tables (from sketches)**





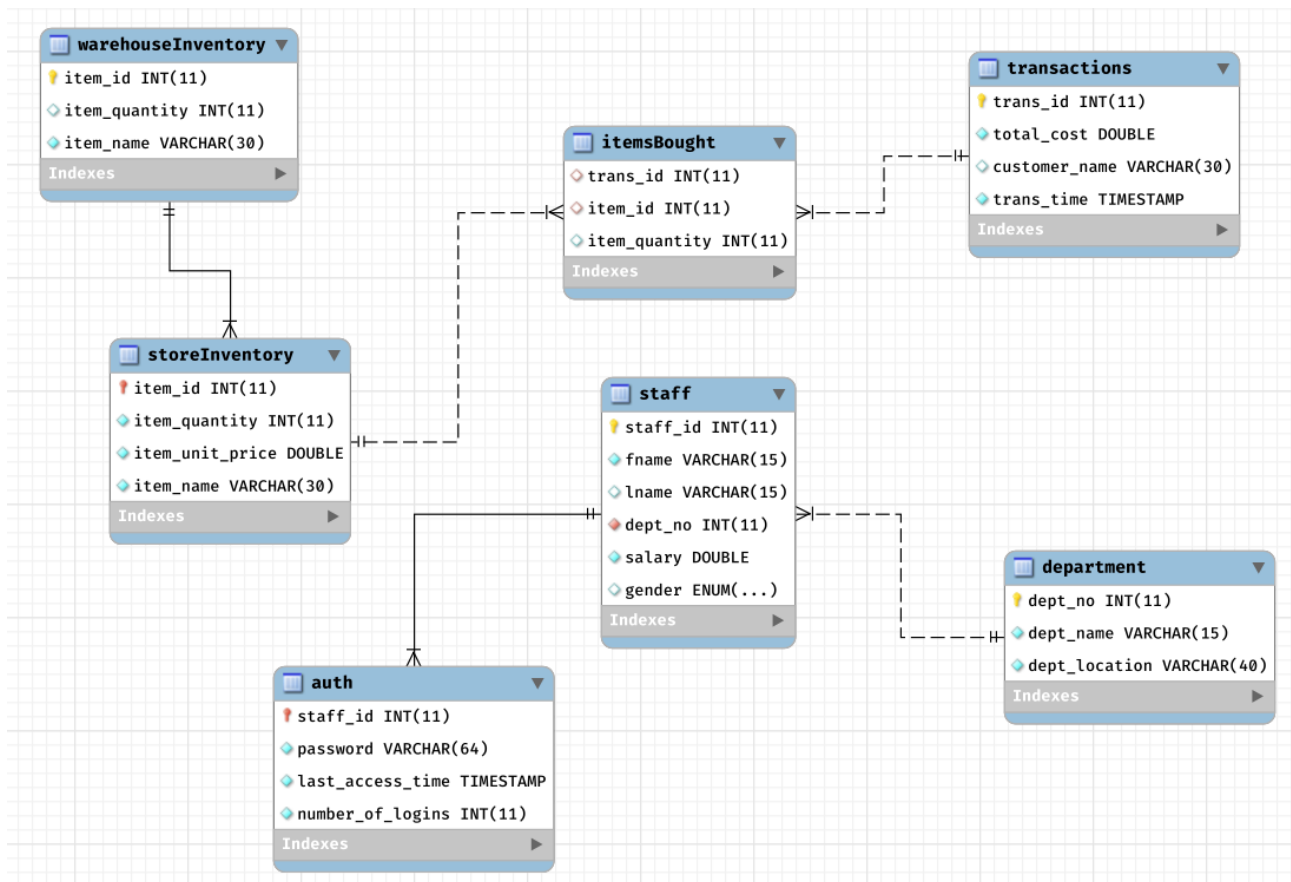
### Conceptual Design (Generated by workbench)



### Updates

- There was the need to keep track of a list of items bought by a customer. However, mysql does not support array types and also custom data types to leverage that functionality.
  - An idea for a fix was to use a table to hold the item ids of the sold products and use *trans\_id* to access them. To simulate the functionality of a list of items, I allowed duplicated *trans\_id*. That way, a SELECT command with WHERE = *trans\_id* will get the list of *item\_ids* that is connected to the transaction.
- Added *trans\_time* to *transactions* table to keep track of the time a transaction is made.

## Conceptual Design (after updates)



## Keys in the Database

Attribute	Key type → Relation
staff_id	<ul style="list-style-type: none"> <li>Primary key → auth</li> <li>Primary key → staff</li> </ul>
dept_no	<ul style="list-style-type: none"> <li>Foreign key → auth</li> <li>Primary key → department</li> <li>Foreign key → staff</li> </ul>
trans_id	<ul style="list-style-type: none"> <li>Primary key → transactions</li> </ul>
item_id	<ul style="list-style-type: none"> <li>Foreign key → itemsBought</li> <li>Primary key → warehouseInventory</li> <li>Primary key → storeInventory</li> <li>Foreign key → storeInventory</li> <li>Foreign key → itemsBought</li> </ul>
item_name	<ul style="list-style-type: none"> <li>Candidate key → warehouseInventory</li> <li>Candidate key → storeInventory</li> </ul>

## Functionalities for Store Inventory Database

*Prompt for chatgpt:* “list 10 functionalities for a storeinventory database system that has the following relations: auth, department, itemsBought, staff, storeInventory, warehouseInventory, and transactions.”

*Response (screenshots):*



A store inventory database system with the mentioned relations (auth, department, itemsBought, staff, storeInventory, warehouseInventory, and transactions) can support a wide range of functionalities to help manage and optimize store operations. Here are 10 functionalities for such a system:

**1. User Authentication and Authorization:**

- Manage user accounts and access permissions.
- Authenticate users before granting access to the system.

**2. Inventory Management:**

- Track and update real-time inventory levels for both the store and warehouse.
- Receive, store, and retrieve information about items in stock.

**3. Item Information:**

- Maintain a comprehensive database of item details, including SKU, name, description, price, and manufacturer.

**4. Transaction Recording:**

- Record sales, purchases, returns, and other inventory-related transactions.
- Link transactions to specific staff members and items sold.

**5. Department Management:**

- Categorize items by department (e.g., electronics, clothing, groceries).
- Assign items to specific departments for organization and reporting.

**5. Department Management:**

- Categorize items by department (e.g., electronics, clothing, groceries).
- Assign items to specific departments for organization and reporting.

**6. Staff Management:**

- Manage staff information, including names, roles, contact details, and schedules.
- Track staff activities, such as item restocking, sales, and returns.

**7. Warehouse Inventory:**

- Monitor inventory levels in the warehouse.
- Support transfer of items between the store and the warehouse.

**8. Reporting and Analytics:**

- Generate reports on sales, inventory levels, and staff performance.
- Analyze sales trends, identify popular items, and make data-driven decisions.

**9. Supplier and Vendor Information:**

- Store details of suppliers and vendors.
- Track supplier contacts, product catalogs, and purchase history.

**10. Stock Alerts and Reordering:**

- Set up automatic alerts for low-stock items.
- Initiate purchase orders for restocking low or out-of-stock items.

These functionalities will enable the store inventory database system to efficiently manage inventory, sales, staff, and other aspects of store operations while providing valuable insights for decision-making and optimization.

*Chosen functionalities:*

1. Staff management.
2. Transaction recording.
3. Stock alerts and reordering.
4. Reporting and analytics.
5. Inventory management.
6. Item information.
7. Warehouse inventory.