

Database Structure for Pizza Billing System

Table: Items

- **Description:**
This table stores the details of items available in the pizza shop. Each item has attributes such as name, type, and price.
- **Fields:**
 - `id` (Primary Key, Integer, Auto Increment): Unique identifier for each item.
 - `name` (String): Name of the item (e.g., Margherita Pizza, Coke).
 - `type` (String): Type of the item (e.g., Food, Beverage).
 - `price` (Float): Price of the item in currency.
- **Relationships:**
 - Each item can be included in multiple invoices but is uniquely identified in this table.

Field Name	Data Type
ID	<i>INTEGER (Primary Key, Auto Increment)</i>
Name	<i>TEXT</i>
Type	<i>TEXT</i>
Price	<i>REAL</i>

Table: Invoices

- **Description:**
This table stores information about invoices created for customers. Each invoice is linked to multiple items through the `InvoiceItems` relationship.
- **Fields:**
 - `id` (Primary Key, Integer, Auto Increment): Unique identifier for each invoice.
 - `customer_name` (String): Name of the customer for whom the invoice is generated.
 - `total` (Float): Total amount of all items in the invoice before tax.
 - `tax` (Float): Tax amount calculated for the invoice.
 - `grand_total` (Float): Final amount after adding tax to the total.
- **Relationships:**
 - An invoice can have multiple items. This is managed through an intermediary table, typically called `InvoiceItems` (not explicitly coded in the provided structure).

Field Name	Data Type
ID	<i>INTEGER (Primary Key, Auto Increment)</i>
Customer_Name	<i>TEXT</i>
Total	<i>REAL</i>
Tax	<i>REAL</i>
Grand_Total	<i>REAL</i>

Intermediary Table: InvoiceItems (Proposed)

- **Description:**
This is an implicit relationship table that links invoices and items. Although not explicitly defined in the `invoice.go` and `item.go`, it is essential for managing many-to-many relationships between the `Invoices` and `Items` tables.
- **Fields:**
 - `invoice_id` (Foreign Key): References the `id` field in the `Invoices` table.
 - `item_id` (Foreign Key): References the `id` field in the `Items` table.
 - `quantity` (Integer): Quantity of the specific item in the invoice.
- **Relationships:**
 - Establishes a many-to-many relationship between invoices and items.

Database Relationships

1. **Invoices to Items:**
 - **Type:** Many-to-Many
 - **Managed Through:** An intermediary table `InvoiceItems`.
 - **Explanation:** An invoice can have multiple items, and each item can appear on multiple invoices.