Simple Sales Application Documentation

Introduction

The Simple Sales Application is designed to handle sales invoices, customers, and articles in a sales environment. It utilizes a Microsoft SQL Server as the Relational Database Management System (RDBMS) to store and manage the data. This documentation provides an overview of the application's objects and their relationships, helping developers understand the structure and functionality of the system.

Database Schema

The database schema consists of three main tables: `SalesInvoice`, `Customers`, and `Articles`. Each table represents a distinct entity in the system and has relationships with other tables. Let's explore the tables and their relationships in detail.

SalesInvoice Table

The `SalesInvoice` table stores information about sales invoices generated in the application. Each invoice has a unique identifier (`InvoiceID`) and includes the following fields:

- `ID` (Primary Key): Unique identifier for the invoice.
- `customer` (Foreign Key): Identifier referencing the associated customer.
- `Invoice_Date`: Date when the invoice was created.
- `Total_price`: The total amount for the invoice.
- payment_method: the way to pay for the invoice.
- articles: Identifier referencing the associated article.

Customers Table

The `Customers` table holds information about the customers who engage in sales transactions. Each customer has a unique identifier (`CustomerID`) and includes the following fields:

- `CustomerID` (Primary Key): Unique identifier for the customer.

- `name`: Name of the customer.

- `email`; email of the customer

Articles Table

The `Articles` table contains details about the articles or products sold in the sales transactions. Each article has a unique identifier (`ArticleID`) and includes the following fields:

- `ArticleID` (Primary Key): Unique identifier for the article.

- `Name`: Name of the article.

- `Price`: Price of the article.

Relationships

The Simple Sales Application uses relationships between the tables to establish connections and enforce data integrity. Here are the relationships defined in the system:

1. One-to-Many Relationship: Customers to Invoices

   - Each customer can have multiple invoices.

   - The `CustomerID` field in the `Invoices` table is a foreign key referencing the `CustomerID` field in the `Customers` table.

2. Many-to-Many Relationship: Articles to Invoices

   - Each invoice can contain multiple articles, and each article can be included in multiple invoices.

   - A separate junction table is used to represent this relationship, named `sale_Invoice`.

   - The `InvoiceID` field in the `sale_invoice` table is a foreign key referencing the `id` field in the `Invoices` table.

   - The `id_articles` field in the `sale_invoice` table is a foreign key referencing the `id_articles` field in the `Articles` table.

Conclusion

This documentation has provided an overview of the objects and relationships in the Simple Sales Application. Understanding the structure and relationships between the tables is crucial for developing and maintaining the application. The Microsoft SQL Server RDBMS serves as the underlying storage and management system, ensuring data integrity and efficient retrieval and manipulation of sales-related information.