

***CS-242***

***Intro to Database***

***Section : BSCS-F22(B)***

***Submitted by:***

*Wajeeha Nawaz (4683-FOC/BSCS/F22)*

*Maria Saba (4650-FOC/BSCS/F22)*

*Wania khan (4677-FOC/BSCS/F22)*

*Zainab (4641-FOC/BSCS/F22)*

*Aiman Fatima (4731-FOC/BSCS/F22)*

***Submitted to:***

*Ms. Saima Abdullah*

***Topic:***

***Restaurant Management System***

**Project scope:**

A restaurant management system encompasses vital functions like menu management, order processing, and inventory control. Opting for this project for database implementation offers practical insights into real-world scenarios. Handling interconnected data entities, such as menus and orders, facilitates exploration of advanced database design strategies. Scalability considerations are essential given the dynamic nature of restaurant operations. Managing sensitive data like customer information underscores the importance of data integrity and security. Additionally, optimizing queries for efficient performance with large datasets enhances database management skills. Overall, implementing a restaurant management system provides valuable hands-on experience in database design and management, addressing practical challenges in the hospitality sector.

**Problem statements:**

1. Ensuring that all data entered the system remains accurate, consistent, and up to date across various modules such as orders, reservations, and inventory.
2. Managing simultaneous access to the database by multiple users to prevent data inconsistencies and conflicts, especially during peak operational hours.
3. Implementing robust security measures to protect sensitive customer information, financial data, and operational details from unauthorized access or cyber threats.
4. Developing a reliable backup and recovery strategy to safeguard against data loss due to system failures, human error, or external factors such as malware attacks.
5. Providing comprehensive training and support for restaurant staff to effectively utilize the database system, troubleshoot common issues, and maximize productivity.

**Proposed solutions:**

1. Implement data validation rules and constraints at the database level to ensure the accuracy and consistency of data entry. Use transactions and atomic operations to maintain data integrity during complex operations.
2. Utilize locking mechanisms such as row-level or table-level locks to control access to database resources during concurrent operations.
3. Implement role-based access control (RBAC) to restrict access to sensitive data and functionalities based on user roles and privileges. Encrypt sensitive data at rest and during transmission to protect against unauthorized access.
4. Implement role-based access control (RBAC) to restrict access to sensitive data and functionalities based on user roles and privileges. Encrypt sensitive data at rest and during transmission to protect against unauthorized access.
5. Provide comprehensive training materials and documentation for restaurant staff on how to use the database system effectively. Offer ongoing support channels such as helpdesk or online forums to address user queries and troubleshoot issues promptly.

**Requirements:**

The Redwood Bistro Restaurant Management System simplifies the management of various restaurant operations such as cooking, serving, and customer interaction. At its core, the system organizes key information across several areas: customers, reservations, orders, menu items, staff, tables, suppliers, and inventory.

**Customers**

Customers are central to the system. They interact with the system in several ways:

* Booking Reservations: They can book tables for specific dates and times, specifying the number of guests and any special requests, such as a preference for a particular table or accommodations for dietary restrictions.
* Placing Orders: Whether dining in, taking out, or requesting delivery, customers can browse the menu and place orders through the system. Each order is tracked from entry to billing.
* Loyalty Program: Customers enroll in a program where they earn points with each purchase, which can be redeemed for discounts or special offers. This program is designed to encourage repeat business and enhance customer retention.

**Orders**

Order management is a critical function:

* Tracking: Each order is assigned a unique ID and tracked through various stages: placed, prepared, served, and paid. The system updates the order status in real time, allowing staff to efficiently manage kitchen output and serving times.
* Payment and Billing: The system records the payment method (e.g., cash, credit card) and calculates the total cost, including items ordered and applicable taxes and tips.

**Menu Items**

* The menu is dynamic and interactive:
* Item Availability: Menu items are listed with details such as price and ingredients. Availability may change based on inventory levels, which the system updates in real time.
* Promotions: The system can flag certain items for promotions, applying discounts or special conditions automatically during checkout.

**Staff**

Staff management is streamlined for efficiency:

* Scheduling: The system assigns shifts, manages hours worked, and ensures that there are enough staff members to cover all operational needs. Each role (waiter, chef, manager) has specific permissions and responsibilities.
* Role Assignments: Chefs focus on food preparation, waiters on customer service, and managers oversee the entire operation, with the ability to generate reports and adjust workflows as needed.

**Tables**

Table management helps optimize the dining experience:

* Seating Arrangements: Tables are managed based on capacity and location. The system helps staff assign customers to tables in a way that maximizes space and enhances the dining experience.

**Suppliers and Inventory**

* Inventory control is tightly integrated with supplier management:
* Inventory Tracking: The system keeps track of stock levels for all ingredients and bar supplies. Alerts are generated when items fall below minimum required quantities, prompting reorders.
* Supplier Management: Details about suppliers, including contact information, reliability ratings, and types of goods supplied, help manage procurement and ensure quality and cost-effectiveness.

**Advanced Features**

* The system includes several advanced functionalities:
* Role-Based Permissions: Different levels of system access are granted based on staff roles, ensuring that sensitive information and critical functions are protected.
* Hierarchical Management Controls: Managers have oversight of sales data, staff performance, and customer feedback, enabling them to make informed decisions about operational adjustments and promotions.
* Automated Financial Calculations: The system automatically calculates financial metrics such as daily sales, cost of goods sold, and net profits, facilitating accurate and timely financial reporting.

**Description:**

**Entities and Attributes**

**Customer:**

**Attributes:** Customer ID (unique), Name, Contact Information (email, phone), Address (composite: Street, City, State, Zip Code), Loyalty Points

**Description:** Customers can dine in, order takeout, or request delivery. They can be part of a loyalty program that accrues points based on their expenditure.

**Reservation:**

**Attributes**: Reservation ID (unique), Customer ID, Date, Time, Number of Guests, Special Requests

**Description:** Customers can make reservations specifying time, date, and size of the party. Special requests (e.g., dietary restrictions) are also recorded.

**Order:**

**Attributes:** Order ID (unique), Date, Time, Status (e.g., pending, completed), Total Price (derived), Payment Type (e.g., cash, credit card), Tip Amount

**Description**: Orders encompass various food items selected by customers. The system tracks order status and aggregates pricing details.

**Food Item:**

**Attributes:** Food Item ID (unique), Name, Description, Price, Category (e.g., appetizer, entree, dessert), Available (Boolean)

**Description:** Food items are listed with detailed descriptions and categorized appropriately. Availability can change based on stock.

**Employee:**

**Attributes:** Employee ID (unique), Name, Role (e.g., waiter, chef, manager), Contact Info, Salary, Start Date, Hours Worked (per week)

**Description:** Employees are tracked by role with relevant personal and job-related details. Roles dictate their access and responsibilities within the system.

**Table:**

**Attributes:** Table ID (unique), Capacity, Location Description

**Description:** Tables are identified by capacity and location within the restaurant, important for arranging reservations and service logistics.

**Supplier:**

**Attributes:** Supplier ID (unique), Name, Contact Information, Type of Goods (e.g., seafood, vegetables), Reliability Rating

**Description:** Suppliers provide necessary kitchen and bar inventory. Their reliability and type of goods supplied are tracked for inventory management.

**Inventory Item:**

**Attributes:** Item ID (unique), Name, Quantity, Minimum Required Quantity, Supplier ID, Last Ordered Date, Price per Unit

**Description**: All ingredients and kitchen items are logged to monitor stock levels, manage reordering processes, and track costs.

**Relationships**

Customers place Orders and make Reservations.

Orders include multiple Food Items and are served by Employees.

Reservations are tied to specific Tables and managed by Employees (host/hostess).

Employees may update Inventory Items and prepare Food Items (chefs).

Suppliers provide Inventory Items critical for the preparation of Food Items.

**Inheritance:**

Employees can be specialized into Chefs, Waiters, and Managers, each with unique attributes and operations.

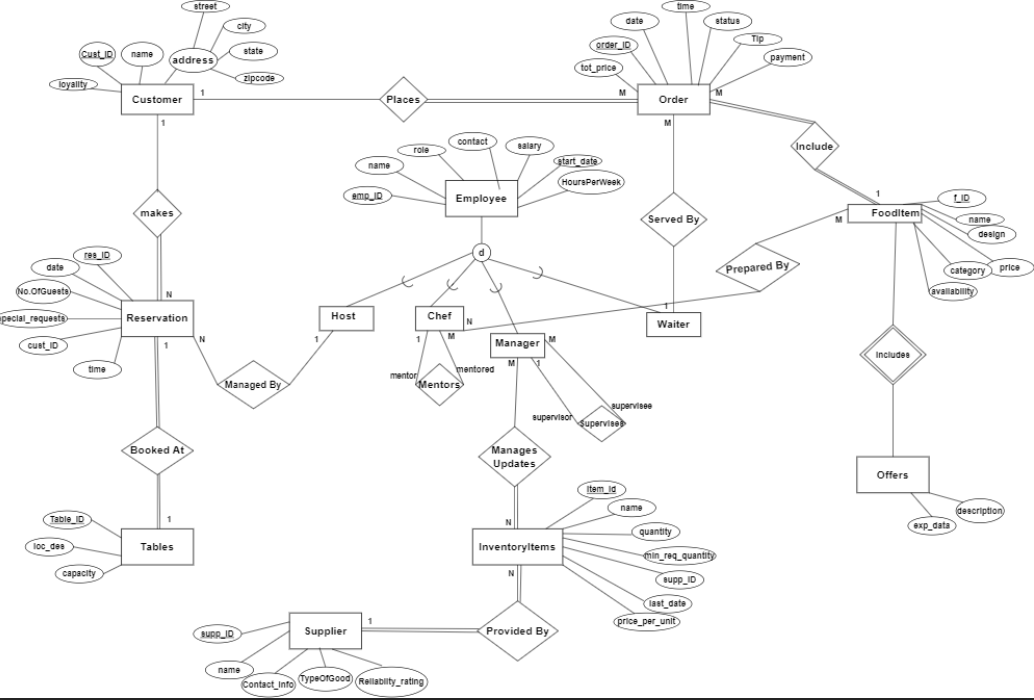
**Recursive Relationships:**

Managers supervise other Employees, and Chefs might mentor Junior Chefs.

**Weak Entities:**

Special Offers for Food Items depending on the day or event, requiring only a promotional description and expiration date, dependent on the Food Item's existence.

Total Price for Orders is calculated from the sum of all included Food Items plus taxes and tips.



**RELATIONAL SCHEMA:**

A diagram of a restaurant menu

Description automatically generated with medium confidence

A diagram of a product

Description automatically generated

A diagram of a diagram

Description automatically generated

**NORMALIZATION:**

**First Normal Form (1NF):**

To be in the First Normal Form, each table must have atomic (indivisible) values, and each record needs to be unique.

**Customer** table has atomic values for each attribute.

**Employee** table has atomic values for each attribute.

**Reservation** table has atomic values for each attribute.

**Order table** has atomic values for each attribute.

**OrderItem** table has atomic values for each attribute.

**FoodItem** table has atomic values for each attribute.

**Table** table has atomic values for each attribute.

**Performance** table has atomic values for each attribute.

**Supplier** table has atomic values for each attribute.

**Inventory Item** table has atomic values for each attribute.

**SpecialOffer** table has atomic values for each attribute.

**Schedules** table has atomic values for each attribute.

**Sales** table has atomic values for each attribute.

All tables satisfy 1NF as they have unique records and atomic values.

**Second Normal Form (2NF):**

To be in the Second Normal Form, the table must first be in 1NF, and all non-key attributes must be fully functionally dependent on the primary key.

**Customer:** All attributes depend on CustomerID.

**Employee:** All attributes depend on EmployeeID.

**Reservation:** All attributes depend on ReservationID.

**Order:** All attributes depend on OrderID.

**OrderItem:** Both OrderID and FoodItemID are required to uniquely identify each record.

**FoodItem:** All attributes depend on FoodItemID.

**Table:** All attributes depend on TableID.

**Supplier:** All attributes depend on SupplierID.

**InventoryItem:** All attributes depend on ItemID.

**SpecialOffer:** All attributes depend on OfferID.

**Schedules:** All attributes depend on ScheduleID.

**Performance:** All attributes depend on PerformanceID.

**Sales:** All attributes depend on SalesID.

All tables satisfy 2NF as all non-key attributes are fully functionally dependent on the primary key of their respective tables.

**Third Normal Form (3NF):**

To be in the Third Normal Form, the table must be in 2NF, and all the attributes must be non-transitively dependent on the primary key. This means no non-key attribute should depend on another non-key attribute.

**Customer:** No transitive dependencies exist.

**Employee:** No transitive dependencies exist.

**Reservation:** No transitive dependencies exist.

**Order:** No transitive dependencies exist.

**OrderItem:** No transitive dependencies exist.

**FoodItem:** No transitive dependencies exist.

**Table:** No transitive dependencies exist.

**Performance:** No transitive dependencies exist.

**Supplier:** No transitive dependencies exist.

**InventoryItem:** No transitive dependencies exist.

**SpecialOffer:** No transitive dependencies exist.

**Schedules:** No transitive dependencies exist.

**Sales:** No transitive dependencies exist.

All tables satisfy 3NF as there are no transitive dependencies.

**Boyce-Codd Normal Form (BCNF)**

**Customer**

Primary Key: CustomerID

Each attribute is fully functionally dependent on CustomerID, and CustomerID is a superkey.

This table is in BCNF.

**Employee**

Primary Key: EmployeeID

Each attribute is fully functionally dependent on EmployeeID, and EmployeeID is a superkey.

This table is in BCNF.

**Reservation**

Primary Key: ReservationID

Each attribute is fully functionally dependent on ReservationID, and ReservationID is a superkey.

This table is in BCNF.

**Order**

Primary Key: OrderID

Each attribute is fully functionally dependent on OrderID, and OrderID is a superkey.

This table is in BCNF.

**OrderFoodItem**

Primary Key: OrderItemID

Each attribute is fully functionally dependent on the OrderItemID, and this key is a superkey.

This table is in BCNF.

**FoodItem**

Primary Key: FoodItemID

Each attribute is fully functionally dependent on FoodItemID, and FoodItemID is a superkey.

This table is in BCNF.

**Table**

Primary Key: TableID

Each attribute is fully functionally dependent on TableID, and TableID is a superkey.

This table is in BCNF.

**Performance**

Primary Key: PerformanceID

Each attribute is fully functionally dependent on the composite key PerformanceID, and this key is a superkey.

This table is in BCNF.

**Supplier**

Primary Key: SupplierID

This table is in BCNF.

**Schedules**

Primary Key: ScheduleID

This table is in BCNF.

**InventoryItem**

Primary Key: ItemID

Each attribute is fully functionally dependent on ItemID, and ItemID is a superkey.

This table is in BCNF.

**SpecialOffer**

Primary Key: OfferID

Each attribute is fully functionally dependent on OfferID, and OfferID is a superkey.

This table is in BCNF.

**Sales**

Primary Key: SalesID

Each attribute is fully functionally dependent on SalesID, and SalesID is a superkey.

This table is in BCNF.

**CONCLUSION:**

The relational schema is normalized and in BCNF form.

Each table has atomic values, all non-key attributes are fully functionally dependent on their primary keys, and there are no transitive dependencies. This structure ensures reduced redundancy and improved data integrity.

Each table's attributes are fully functionally dependent on a superkey, ensuring there are no partial dependencies or transitive dependencies that violate BCNF.

**TABLES CREATED IN MS ACCESS:**

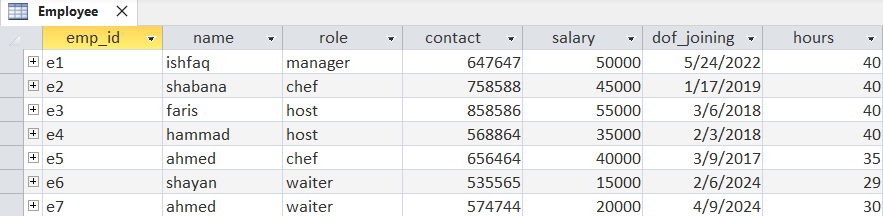
**CUSTOMER:**

A screenshot of a computer

Description automatically generated

**EMPLOYEE:**

**Query:** CREATE TABLE Employee (emp\_id TEXT PRIMARY KEY,name TEXT,role TEXT,contact NUMBER,salary NUMBER,dof\_joining DATE,hours NUMBER);

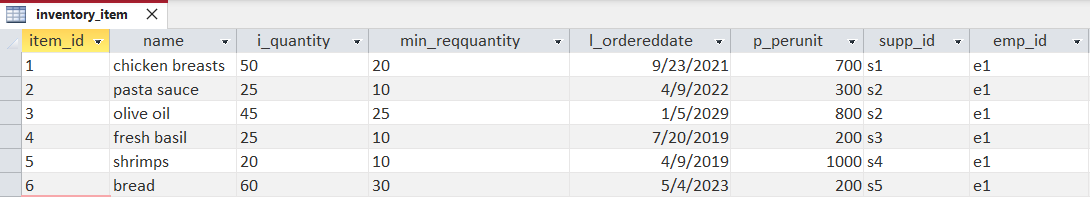


**FOODITEM:**

**A screenshot of a computer

Description automatically generated**

**INVENTORY ITEM:**

****

**ORDER:**

**A screenshot of a computer

Description automatically generated**

**ORDER ITEM:**

**A screenshot of a computer

Description automatically generated**

**PERFORMANCE:**

**QUERY:** CREATE TABLE Performance(p\_id TEXT PRIMARY KEY,p\_rating DOUBLE,emp\_id TEXT);

**A screenshot of a computer

Description automatically generated**

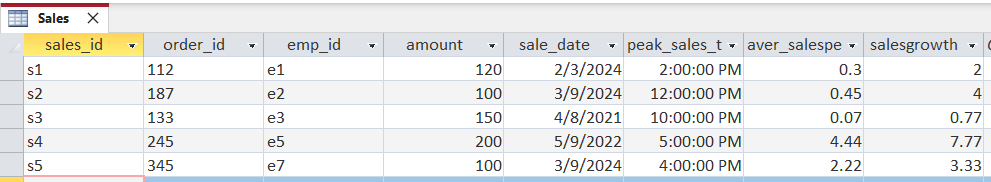
**RESERVATION:**

**A screenshot of a computer

Description automatically generated**

**SALES:**

**Query:** CREATE TABLE Sales(sales\_id TEXT PRIMARY KEY , order\_id TEXT,emp\_id TEXT,amount NUMBER,sale\_date DATE,peak\_sales\_time DATE,avg\_salesperday DOUBLE,salesgrowth DOUBLE);

****

**SCHEDULES:**

**Query:** CREATE TABLE Schedules(sc\_id TEXT PRIMARY KEY,s\_start\_time TIME,s\_end\_time TIME,s\_date DATE,emp\_id TEXT);

**A screenshot of a computer

Description automatically generated**

**SPECIAL OFFERS:**

**Query:** CREATE TABLE SpecialOffer(offer\_id TEXT PRIMARY KEY,description TEXT,exp\_date DATE,disc\_percent DOUBLE,fitem\_id TEXT);

**A screenshot of a computer

Description automatically generated**

**SUPPLIER:**

**A screenshot of a computer

Description automatically generated**

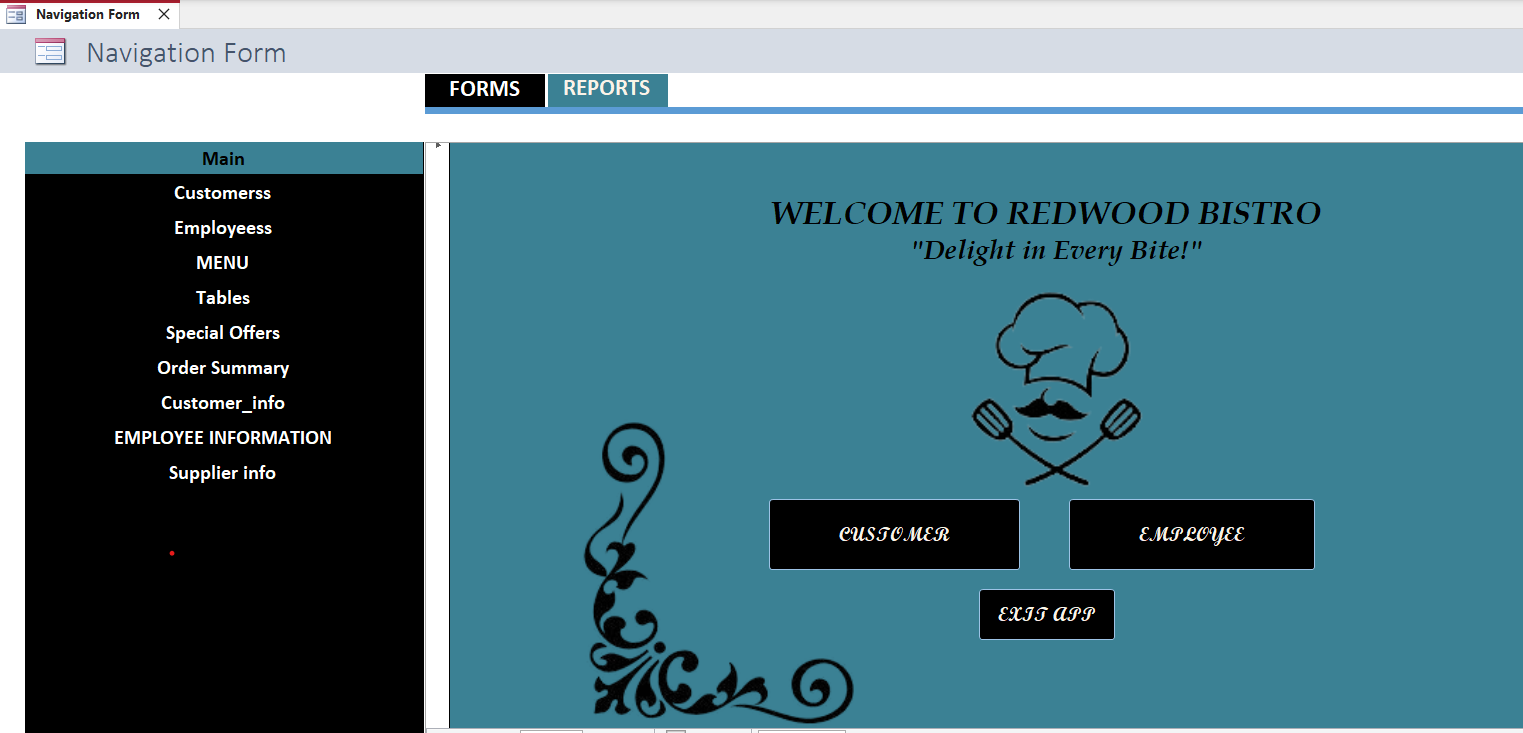
**TABLE:**

**A screenshot of a computer

Description automatically generated**

**FORMS:**

**NAVIGATION FORM:**

****

**MAIN:A blue and black website

Description automatically generated**

**CUSTOMERS: A screenshot of a menu

Description automatically generatedEMPLOYEES:A blue and black website

Description automatically generated with medium confidence**

**MENU:A screenshot of a menu

Description automatically generated**

**CUSTOMER INFO:**

**A screenshot of a computer

Description automatically generated**

**EMPLOYEE INFO:**

**A screenshot of a computer

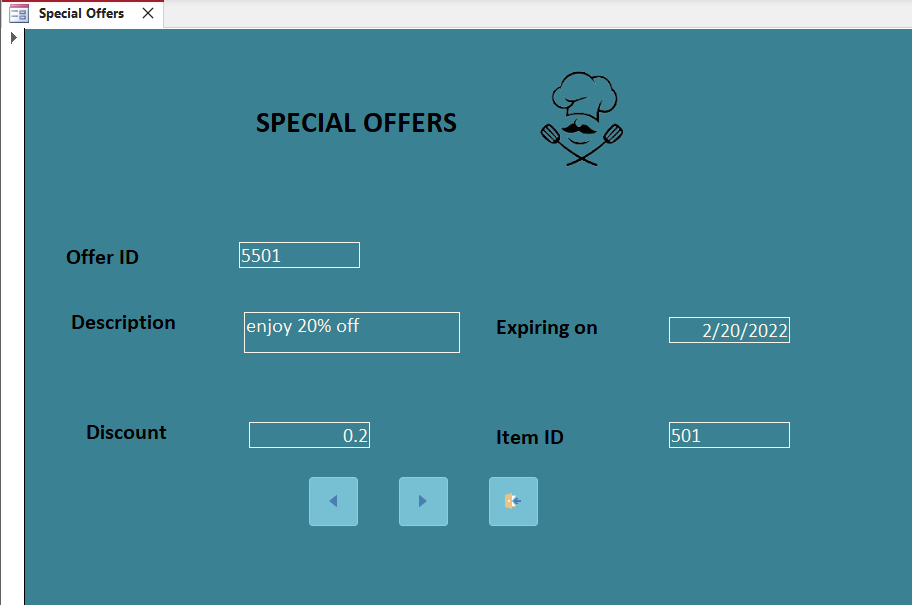
Description automatically generated**

**ORDER SUMMARY:**

A screenshot of a computer

Description automatically generated

**SPECIAL OFFERS:**



**SUPPLIER INFO:**

**A screenshot of a computer

Description automatically generated**

**TABLES:**

A screenshot of a computer

Description automatically generated

**REPORTS:**

**COMPREHENSIVE SALES ANALYSIS REPORT:**

**A screenshot of a computer

Description automatically generated**

**CUSTOMER INSIGHT REPORT:**

**A screenshot of a computer

Description automatically generated**

**EMPLOYEE EFFICIENCY REPORT:**

**A screenshot of a computer

Description automatically generated**

**ORDER FULFILLMENT AND CUSTOMER SATISFACTION REPORT:**

**A screenshot of a computer

Description automatically generated**

**RESERVATION AND TABLE UTILIZATION REPORT:**

**A screenshot of a computer

Description automatically generated**

**SHIFT SCHEDULING AND COVERAGE REPORT:**

A screenshot of a computer

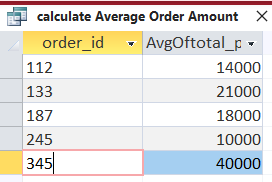
Description automatically generated

**CALCULATE AVERAGE ORDER AMOUNT:**

SELECT Order.order\_id, Avg(Order.total\_price) AS AvgOftotal\_price

FROM [Order]

GROUP BY Order.order\_id;

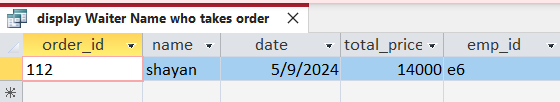
****

**DISPLAT WAITER NAME WHO TAKES ORDER:**

SELECT order\_id, Employee.name, date, total\_price, Employee.emp\_id

FROM [Order] INNER JOIN Employee ON [Order].emp\_id = Employee.emp\_id

WHERE [Order].cu\_id = [EnterCustomerID];

****

**INVENTORY LEVELS FOR ITEMS BELOW THE MINIMUM REQUIRED QUANTITY:**

SELECT name, i\_quantity, min\_reqquantity

FROM inventory\_item

WHERE i\_quantity < min\_reqquantity;

**A screenshot of a computer

Description automatically generated**

LIST ALL MENU ITEMS WITH PRICE:

SELECT fitem\_id, name, description, price

FROM fooditem;

**A screenshot of a computer

Description automatically generated**

**ORDER QUERY:**

SELECT [display Waiter Name who takes order].order\_id, [display Waiter Name who takes order].name, [display Waiter Name who takes order].date, [display Waiter Name who takes order].total\_price, [display Waiter Name who takes order].emp\_id

FROM [display Waiter Name who takes order];

**A screenshot of a computer

Description automatically generated**

**ORDER TOTAL PRICE:**

SELECT oi.order\_id, Sum(fi.price\*oi.[qty]) AS totalprice

FROM fooditem AS fi INNER JOIN orderitem AS oi ON fi.[fitem\_id]=oi.[fitem\_id]

GROUP BY oi.order\_id;

**A screenshot of a computer

Description automatically generated**

**PENDING ORDER:**

SELECT customer.name, Order.time, Order.date, Order.status

FROM customer INNER JOIN [Order] ON customer.cu\_id = Order.cu\_id

WHERE (((Order.status)='preparing'));

**A screenshot of a computer

Description automatically generated**

**PERFORMANCE QUERY:**

SELECT Performance.p\_id, Performance.p\_rating, Performance.emp\_id, Employee.name, Employee.contact, Employee.salary

FROM Employee INNER JOIN Performance ON Employee.[emp\_id] = Performance.[emp\_id];

**A screenshot of a computer

Description automatically generated**

**RETRIEVE ALL CUSTOMER DETAILS:**

SELECT Customer.cu\_id, Customer.name, Customer.phone, Customer.Email, Customer.state, Customer.city, Customer.street

FROM Customer;

**A screenshot of a computer

Description automatically generated**

**RETRIEVE ALL WAITERS:**

SELECT Employee.emp\_id, Employee.name, Employee.role, Employee.contact, Employee.contact, Employee.salary, Employee.dof\_joining, Employee.hours

FROM Employee

WHERE (((Employee.role)='waiter'));

**A screenshot of a computer

Description automatically generated**

**RETRIEVE ORDERS PLACED BY A SPECIFIC CUSTOMER:**

SELECT order\_id, date, total\_price

FROM [Order]

WHERE cu\_id = [EnterCustomerID];

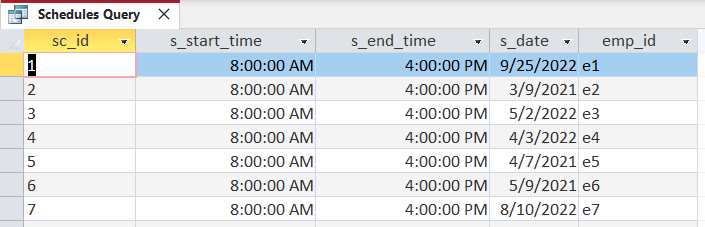
**A screenshot of a computer

Description automatically generated**

**SCHEDULE QUERY:**

SELECT Schedules.[sc\_id], Schedules.[s\_start\_time], Schedules.[s\_end\_time], Schedules.[s\_date], Schedules.[emp\_id]

FROM Schedules;

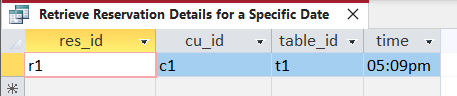
****

**RETRIEVE RESERVATION DETAILS FOR A SPECIFIC DATE:**

SELECT res\_id, cu\_id, table\_id, time

FROM reservation

WHERE date = [EnterDate];



**Conclusion:**

Implementing the Redwood Bistro Restaurant Management System in Microsoft Access has greatly enhanced our operations. The normalized database ensures data integrity and efficiency. Key improvements include better customer management, streamlined order processing, dynamic inventory control, optimized staff scheduling, and secure access control. This system has optimized operations, improved customer satisfaction, and positioned Redwood Bistro for success.