**SOFTWARE ENGINEERING 1 PRELIM EXAM**

**A BUSINESS ESTABLISHMENT'S DATABASE**

**NAME:Montemayor, Charles David A.**

**SECTION: UCOS 3-1**

**INSTRUCTIONS:**

* Please keep in mind that this activity should be done **INDIVIDUALLY ONLY**. This will serve as an individual assessment to identify **how much you've learned** from the prelim period. Please follow the format on the next page.
* Your goal for your prelim exam is to come up with a **software engineering scenario** and then create a database for this concept.
* For example, a restaurant asking you to create an order management system or a school needing a library management system and your goal is to create a database for the system.
* The database should have a **minimum of 5 tables**. The finals exam project consists of two parts:
  + Part 1: Data Model
  + Part 2: Creating tables, inserting records, and reading records.
* **FOR PART 1**, I want you to create the data model of your database using this website only: <https://dbdiagram.io/d>. Please highlight the relationships so that we can see which entities have one to one, one to many or many to many relationships.
* **FOR PART 2**, please show the sql queries listed below. You can use the SQL editor from <https://www.programiz.com/sql/online-compiler/>

1. SQL query for creating the tables.
2. SQL query for inserting records into the tables *(You can use AI to generate random data for you to finish this part much faster)*.
3. Three to five (3-5) SQL queries that demonstrate the use of any of these: WHERE, JOIN, GROUP BY, and aggregate functions.

***TITLE HERE***

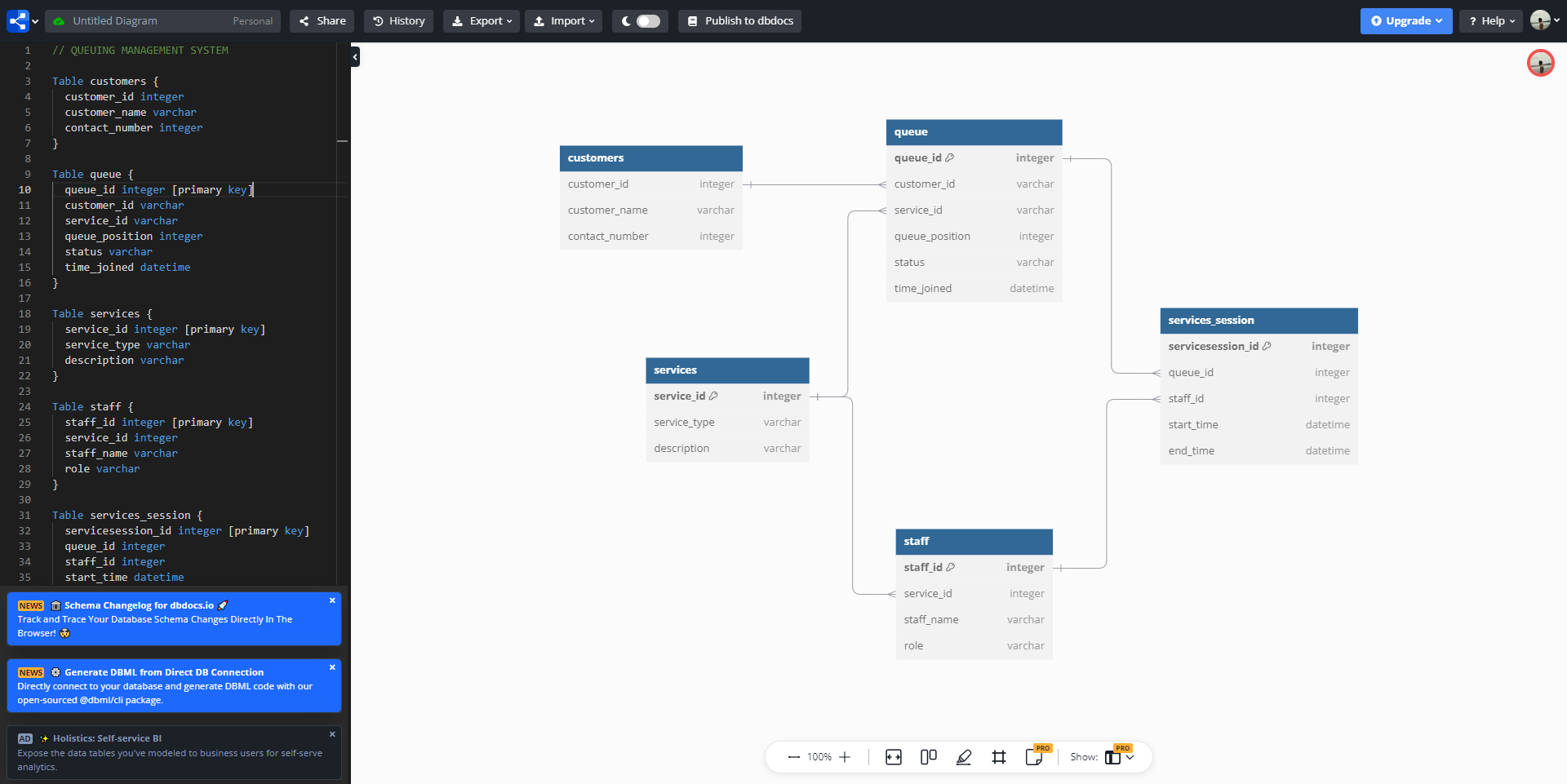
*“QUEUING MANAGEMENT SYSTEM”*

***DESCRIPTION HERE***

*The schema in my queuing management system tracks customer queue positions, service sessions, and staff assignments, efficiently managing and monitoring the flow of services.*

**PART 1: DATA MODEL**

*paste a screenshot of the data model*

**

*add a 2-3 sentence description about the data model*

* *My data model is designed for a queuing management system, featuring five tables. The Customers table records information about each customer, while the Queue table tracks customer service requests, including their status and position in the queue. The Services table defines what types of services should be done, for the Staff to be able to perform this service under the staff table which also provide the details of the personnel who provide these services.*

**PART 2: SQL QUERIES**

**SQL QUERY FOR CREATING THE TABLES**

*Paste query here*

|  |
| --- |
| CREATE TABLE customers (  customer\_id INT PRIMARY KEY,  customer\_name VARCHAR(100) NOT NULL,  contact\_number VARCHAR(15)  );  CREATE TABLE queue (  queue\_id INT PRIMARY KEY,  customer\_id INT,  service\_id INT,  queue\_position INT,  status VARCHAR(50),  time\_joined DATETIME  );  CREATE TABLE services (  service\_id INT PRIMARY KEY,  service\_type VARCHAR(100),  description VARCHAR(255)  );  CREATE TABLE staff (  staff\_id INT PRIMARY KEY,  service\_id INT,  staff\_name VARCHAR(100),  role VARCHAR(50)  );  CREATE TABLE services\_session (  servicesession\_id INT PRIMARY KEY,  queue\_id INT,  staff\_id INT,  start\_time DATETIME,  end\_time DATETIME  ); |

**SQL QUERY FOR INSERTING RECORDS IN ATABLE**

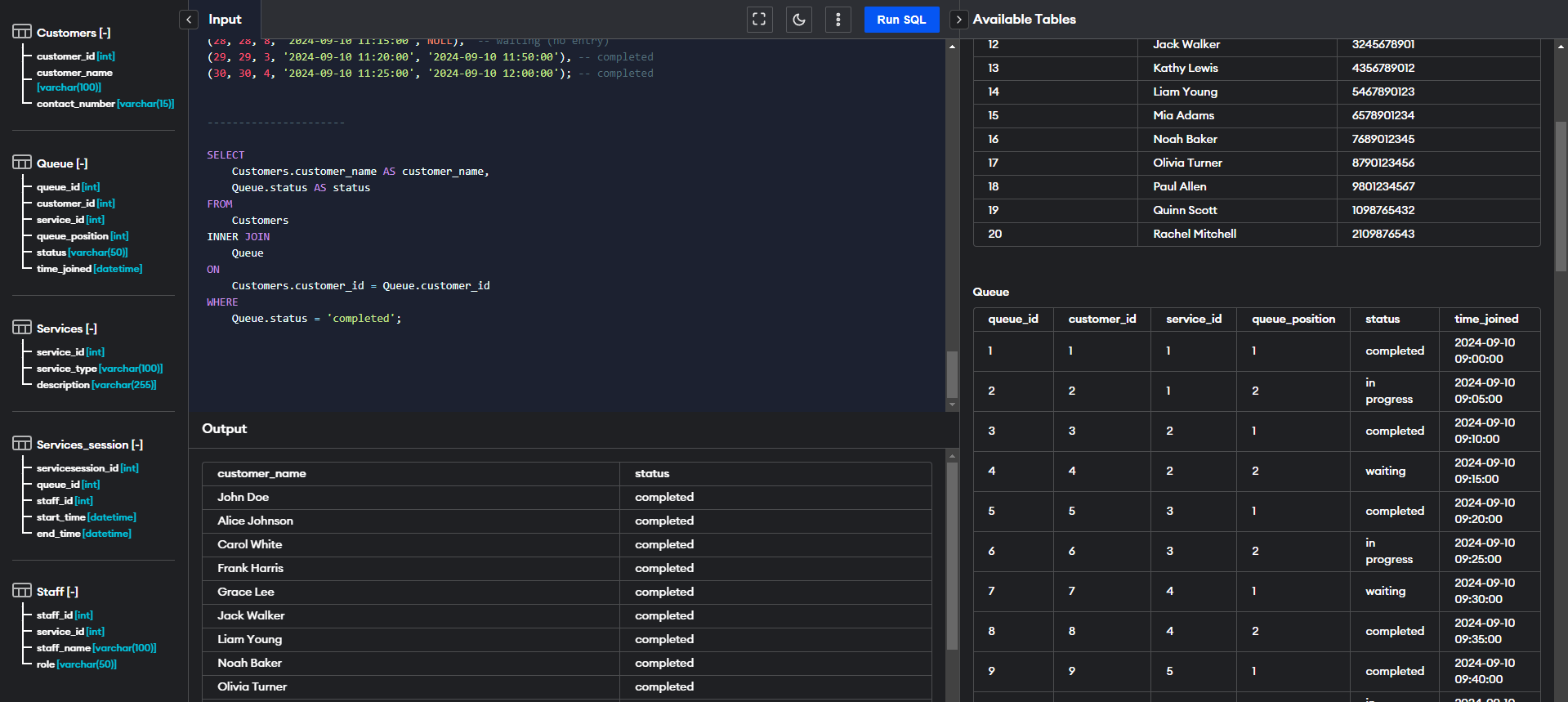
*Paste query here. Five rows of data per table can be considered enough. You can use AI to work on this part much faster.*

|  |
| --- |
| *INSERT INTO customers (customer\_id, customer\_name, contact\_number) VALUES*  *(1, 'John Doe', 1234567890),*  *(2, 'Jane Smith', 2345678901),*  *(3, 'Alice Johnson', 3456789012),*  *(4, 'Bob Brown', 4567890123),*  *(5, 'Carol White', 5678901234),*  *(6, 'David Black', 6789012345),*  *(7, 'Eva Green', 7890123456),*  *(8, 'Frank Harris', 8901234567),*  *(9, 'Grace Lee', 9012345678),*  *(10, 'Henry King', 1023456789),*  *(11, 'Ivy Clark', 2134567890),*  *(12, 'Jack Walker', 3245678901),*  *(13, 'Kathy Lewis', 4356789012),*  *(14, 'Liam Young', 5467890123),*  *(15, 'Mia Adams', 6578901234),*  *(16, 'Noah Baker', 7689012345),*  *(17, 'Olivia Turner', 8790123456),*  *(18, 'Paul Allen', 9801234567),*  *(19, 'Quinn Scott', 1098765432),*  *(20, 'Rachel Mitchell', 2109876543);*  *INSERT INTO services (service\_id, service\_type, description) VALUES*  *(1, 'Technical Support', 'Assistance with technical issues'),*  *(2, 'Billing Inquiry', 'Questions related to billing'),*  *(3, 'Account Management', 'Support with account settings'),*  *(4, 'Product Information', 'Details about products'),*  *(5, 'Customer Feedback', 'Collecting feedback from customers');*  *INSERT INTO queue (queue\_id, customer\_id, service\_id, queue\_position, status, time\_joined) VALUES*  *(1, 1, 1, 1, 'completed', '2024-09-10 09:00:00'),*  *(2, 2, 1, 2, 'in progress', '2024-09-10 09:05:00'),*  *(3, 3, 2, 1, 'completed', '2024-09-10 09:10:00'),*  *(4, 4, 2, 2, 'waiting', '2024-09-10 09:15:00'),*  *(5, 5, 3, 1, 'completed', '2024-09-10 09:20:00'),*  *(6, 6, 3, 2, 'in progress', '2024-09-10 09:25:00'),*  *(7, 7, 4, 1, 'waiting', '2024-09-10 09:30:00'),*  *(8, 8, 4, 2, 'completed', '2024-09-10 09:35:00'),*  *(9, 9, 5, 1, 'completed', '2024-09-10 09:40:00'),*  *(10, 10, 5, 2, 'in progress', '2024-09-10 09:45:00'),*  *(11, 11, 1, 3, 'waiting', '2024-09-10 09:50:00'),*  *(12, 12, 1, 4, 'completed', '2024-09-10 09:55:00'),*  *(13, 13, 2, 3, 'in progress', '2024-09-10 10:00:00'),*  *(14, 14, 2, 4, 'completed', '2024-09-10 10:05:00'),*  *(15, 15, 3, 2, 'waiting', '2024-09-10 10:10:00'),*  *(16, 16, 3, 3, 'completed', '2024-09-10 10:15:00'),*  *(17, 17, 4, 2, 'completed', '2024-09-10 10:20:00'),*  *(18, 18, 4, 3, 'in progress', '2024-09-10 10:25:00'),*  *(19, 19, 5, 2, 'waiting', '2024-09-10 10:30:00'),*  *(20, 20, 5, 3, 'completed', '2024-09-10 10:35:00'),*  *(21, 14, 1, 5, 'completed', '2024-09-10 10:40:00'),*  *(22, 3, 2, 5, 'in progress', '2024-09-10 10:45:00'),*  *(23, 2, 3, 6, 'waiting', '2024-09-10 10:50:00'),*  *(24, 9, 4, 4, 'completed', '2024-09-10 10:55:00'),*  *(25, 11, 5, 6, 'in progress', '2024-09-10 11:00:00'),*  *(26, 18, 1, 7, 'waiting', '2024-09-10 11:05:00'),*  *(27, 14, 2, 8, 'completed', '2024-09-10 11:10:00'),*  *(28, 19, 3, 9, 'in progress', '2024-09-10 11:15:00'),*  *(29, 14, 4, 10, 'completed', '2024-09-10 11:20:00'),*  *(30, 11, 5, 11, 'waiting', '2024-09-10 11:25:00');*  *INSERT INTO staff (staff\_id, service\_id, staff\_name, role) VALUES*  *(1, 1, 'Emily Davis', 'Support Specialist'),*  *(2, 2, 'Michael Clark', 'Billing Agent'),*  *(3, 3, 'Sarah Wilson', 'Account Manager'),*  *(4, 4, 'David Lewis', 'Product Specialist'),*  *(5, 5, 'Linda Martin', 'Feedback Collector'),*  *(6, 1, 'James Thompson', 'Support Specialist'),*  *(7, 2, 'Sophia White', 'Billing Agent'),*  *(8, 3, 'Daniel Harris', 'Account Manager'),*  *(9, 4, 'Olivia Scott', 'Product Specialist'),*  *(10, 5, 'Matthew Carter', 'Feedback Collector');*  *INSERT INTO services\_session (servicesession\_id, queue\_id, staff\_id, start\_time, end\_time) VALUES*  *(1, 1, 1, '2024-09-10 09:00:00', '2024-09-10 09:30:00'),*  *(2, 2, 1, '2024-09-10 09:05:00', NULL),*  *(3, 3, 3, '2024-09-10 09:10:00', '2024-09-10 09:40:00'),*  *(4, 4, 4, '2024-09-10 09:15:00', NULL),*  *(5, 5, 7, '2024-09-10 09:20:00', '2024-09-10 09:50:00'),*  *(6, 6, 5, '2024-09-10 09:25:00', NULL),*  *(7, 7, 8, '2024-09-10 09:30:00', NULL),*  *(8, 8, 6, '2024-09-10 09:35:00', '2024-09-10 10:10:00'),*  *(9, 9, 10, '2024-09-10 09:40:00', '2024-09-10 10:15:00'),*  *(10, 10, 10, '2024-09-10 09:45:00', '2024-09-10 10:20:00'),*  *(11, 11, 1, '2024-09-10 09:50:00', '2024-09-10 10:25:00'),*  *(12, 12, 2, '2024-09-10 09:55:00', '2024-09-10 10:30:00'),*  *(13, 13, 3, '2024-09-10 10:00:00', NULL),*  *(14, 14, 4, '2024-09-10 10:05:00', '2024-09-10 10:40:00'),*  *(15, 15, 2, '2024-09-10 10:10:00', NULL),*  *(16, 16, 6, '2024-09-10 10:15:00', '2024-09-10 10:50:00'),*  *(17, 17, 7, '2024-09-10 10:20:00', '2024-09-10 10:55:00'),*  *(18, 18, 6, '2024-09-10 10:25:00', '2024-09-10 11:00:00'),*  *(19, 19, 9, '2024-09-10 10:30:00', NULL),*  *(20, 20, 8, '2024-09-10 10:35:00', '2024-09-10 11:10:00'),*  *(21, 21, 1, '2024-09-10 10:40:00', '2024-09-10 11:15:00'),*  *(22, 22, 5, '2024-09-10 10:45:00', NULL),*  *(23, 23, 3, '2024-09-10 10:50:00', '2024-09-10 11:25:00'),*  *(24, 24, 1, '2024-09-10 10:55:00', '2024-09-10 11:30:00'),*  *(25, 25, 9, '2024-09-10 11:00:00', NULL),*  *(26, 26, 6, '2024-09-10 11:05:00', '2024-09-10 11:40:00'),*  *(27, 27, 2, '2024-09-10 11:10:00', '2024-09-10 11:45:00'),*  *(28, 28, 8, '2024-09-10 11:15:00', NULL),*  *(29, 29, 3, '2024-09-10 11:20:00', '2024-09-10 11:50:00'),*  *(30, 30, 4, '2024-09-10 11:25:00', '2024-09-10 12:00:00');* |

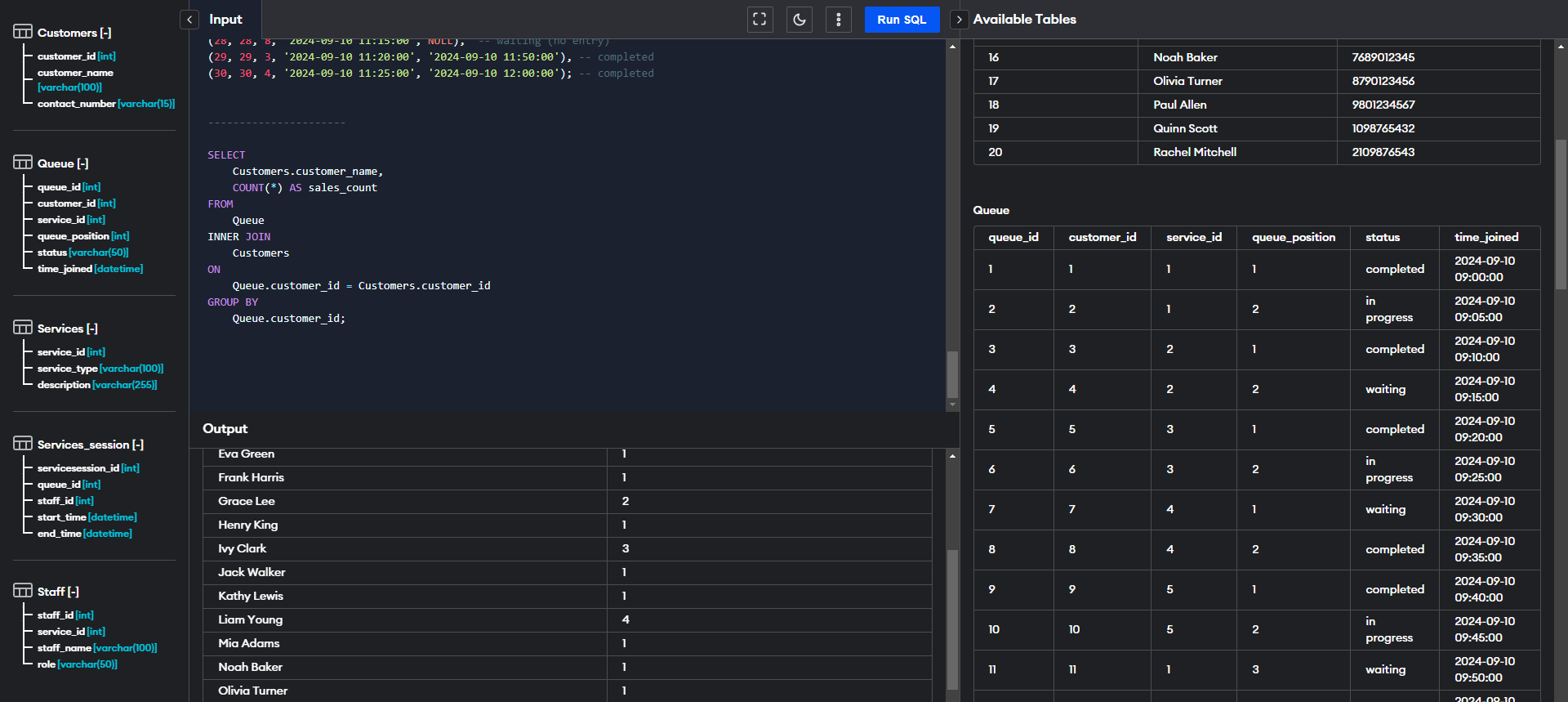
**SQL QUERIES DEMONSTRATING WHERE, JOIN, GROUP BY SQL STATEMENTS.**

*Show 3-5 SQL queries demonstrating any of the following WHERE, JOIN, GROUP BY, and aggregate functions.*

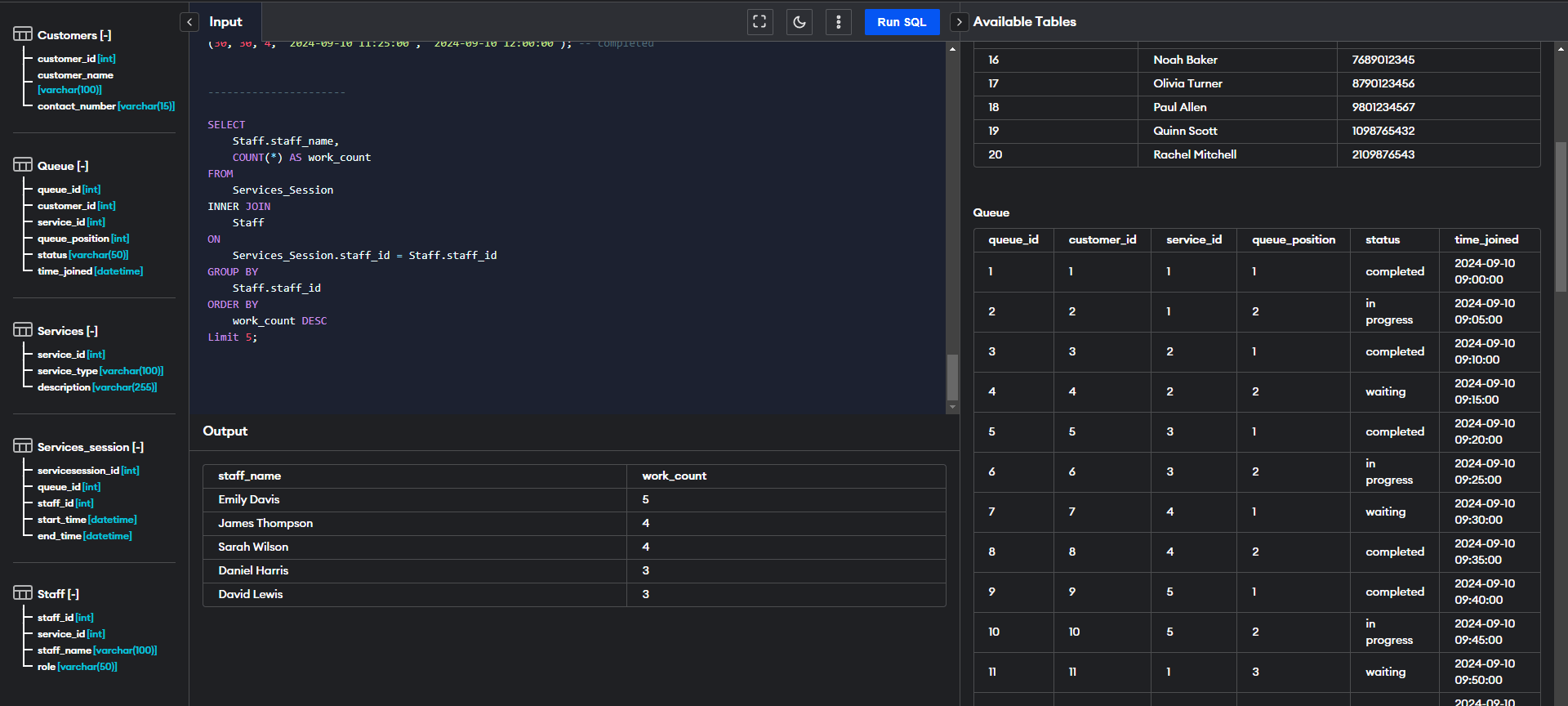
*//* *Listing the names of customers for whom the service has already been finished*

**

*//* *Total number of sales for each customer*

**

*//* *Generating the top 5 staff/employees who have the highest number of work*

**

***Github repository link (optional) if you need:***

[*https://github.com/foshoo0/3rdyrprelim\_projfinal*](https://github.com/foshoo0/3rdyrprelim_projfinal)