**Olympiad Registration System Documentation**

**Project Title:** Olympiad Registration System

**Members:**

* Shoaib [F22605029]
* Wasiq[F22605042]
* Fahad Ali[F22605039]

**Date:** 6th June, 2024

**Table of Contents**

1. Enhanced ERD Diagram
2. Entity Attributes
3. SQL Statements
4. Screenshots of Query Outputs
5. Normalized Tables
6. Additional Details

**Enhanced ERD Diagram**

**Entity Attributes**

**Users**

* **user\_id** (Primary Key)
* username
* password
* email
* full\_name
* role\_id (Foreign Key)
* created\_at

**Events**

* **event\_id** (Primary Key)
* event\_name
* description
* event\_date
* location
* created\_by (Foreign Key)

**Registrations**

* **registration\_id** (Primary Key)
* user\_id (Foreign Key)
* event\_id (Foreign Key)
* registration\_date

**Payments**

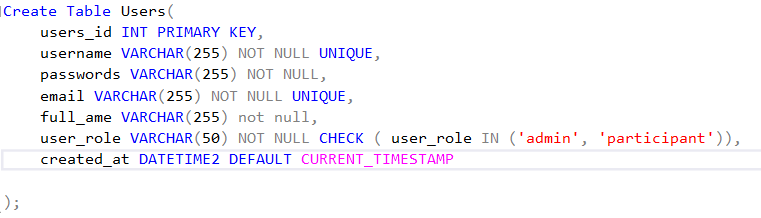
* **payment\_id** (Primary Key)
* registration\_id (Foreign Key)
* amount
* payment\_date
* status\_id (Foreign Key)

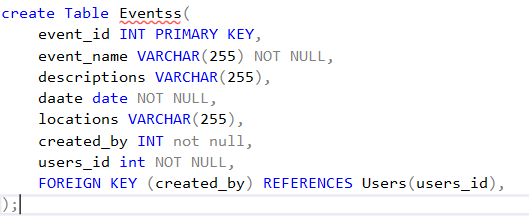
**SQL Statements**

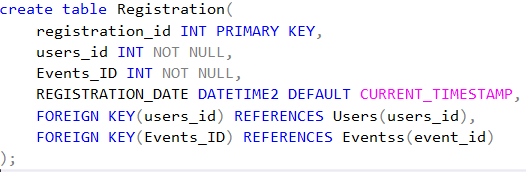
**Database Creation**

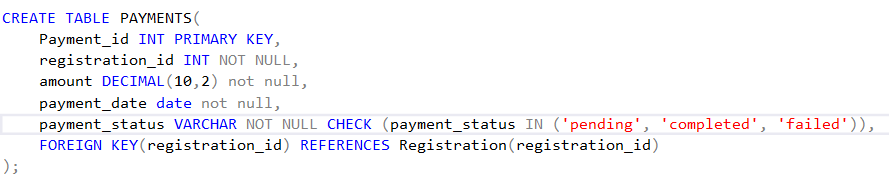


**Table Creation**





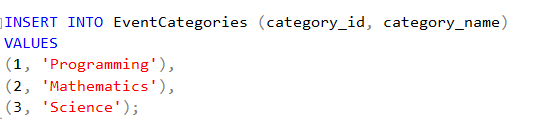


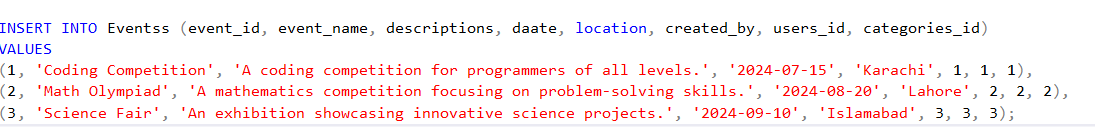


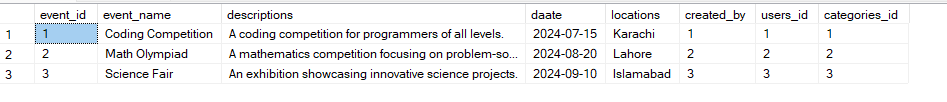
**Data Population**

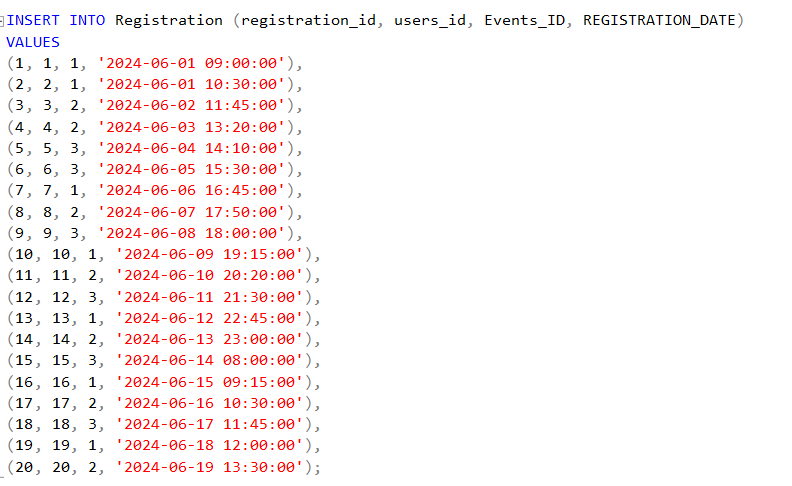


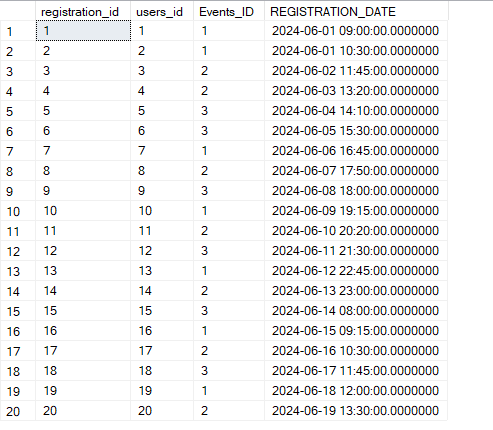


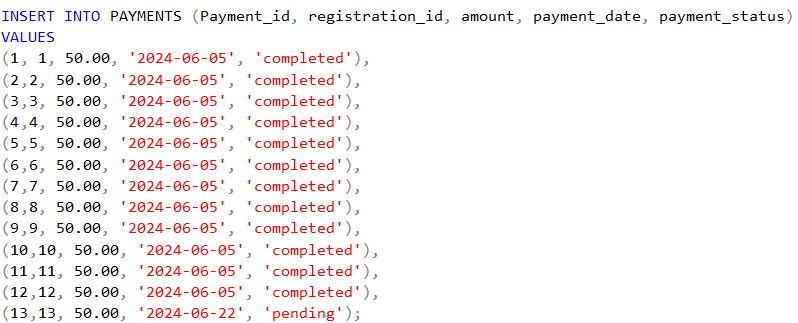


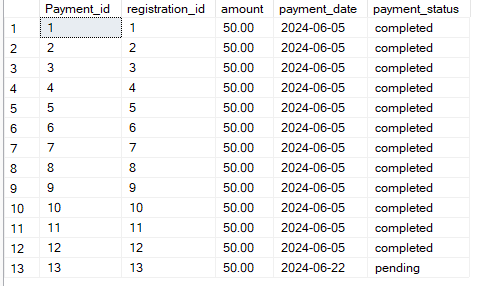








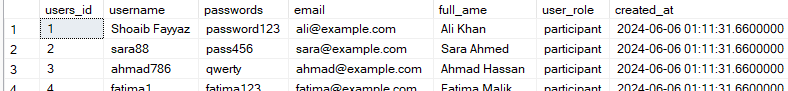


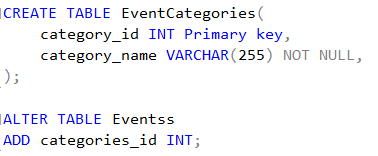


**Query Tasks & Screenshots of Query Outputs**

* 1. **Creating, Updating and Altering Tables**





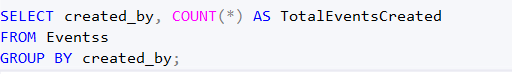


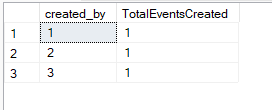
* 1. **Implementing SQL functions for data transformation or calculation.**





* 1. **Aggregation functions**

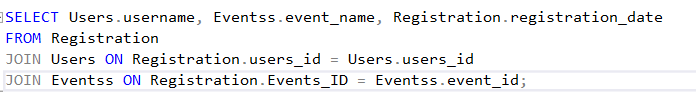


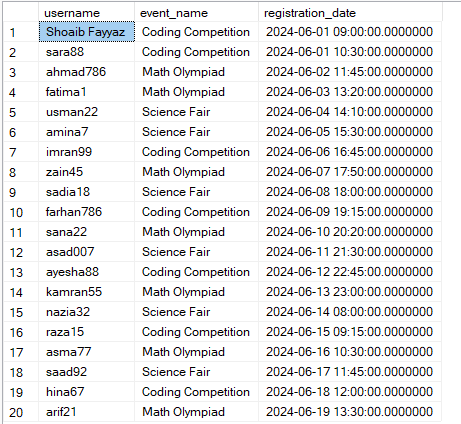






* 1. **Join functions to retrieve data from multiple tables**





**Normalization Explanation**

* **1NF**: All tables have atomic columns and unique rows.
* **2NF**: There are no partial dependencies since all non-key attributes depend on the whole primary key.
* **3NF**: There are no transitive dependencies since all attributes are only dependent on the primary key.

**Users Table**

* **1NF**: Each attribute contains atomic values, and each row is unique.
* **2NF**: No partial dependencies; all non-key attributes depend on the entire primary key (user\_id).
* **3NF**: No transitive dependencies; all attributes are only dependent on the primary key (user\_id).

**Events Table**

* **1NF**: Atomic values and unique rows.
* **2NF**: Fully dependent on the primary key (event\_id).
* **3NF**: Attributes depend only on the primary key (event\_id).

**Registrations Table**

* **1NF**: Atomic values and unique rows.
* **2NF**: Fully dependent on the composite primary key (registration\_id).
* **3NF**: Attributes depend only on the composite primary key (registration\_id).

**Payments Table**

* **1NF**: Atomic values and unique rows.
* **2NF**: Fully dependent on the primary key (payment\_id).
* **3NF**: Attributes depend only on the primary key (payment\_id).

**Additional Details**

 **Indexes**: Adding indexes to frequently queried columns (e.g., user\_id in the Registrations table, event\_id in the Payments table) can significantly improve query performance. Indexes speed up data retrieval operations by allowing the database engine to find rows more quickly.

 **Normalization Trade-offs**: While normalization reduces redundancy and improves data integrity, it can sometimes lead to complex queries and slower performance due to the need for joins. In such cases, consider denormalization balance performance and data integrity requirements.