Sillah Phase 4

CS340: Introduction to Databases Systems

Section: 1629

Instructor: Maram Alajlan

Prepared by:

Yara Albugami - 223410190

Rose Alrakan - 223410382

Shoug Alomran - 223410392

Raghad Abdulaziz - 222511358

Table of Contents

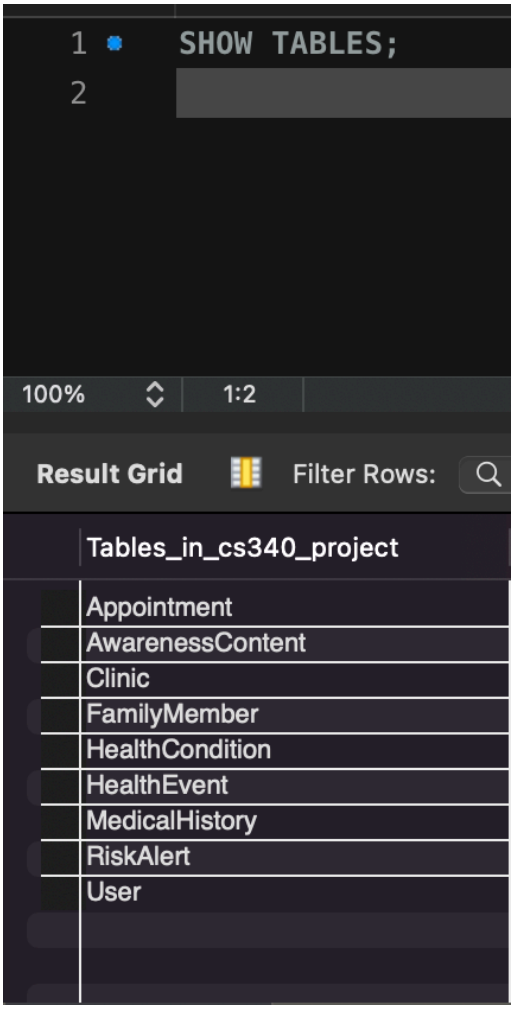
1. Database Implementation

1.1 Database Creation and Tables

The database cs340_project was successfully created and initialized.

To verify the schema structure, the SHOW TABLES; command was executed.

Presenting All Tables



The following tables were created:

- Appointment
- AwarenessContent
- Clinic

- FamilyMember
- HealthEvent
- RiskAlert
- HealthCondition
- MedicalHistory
- User

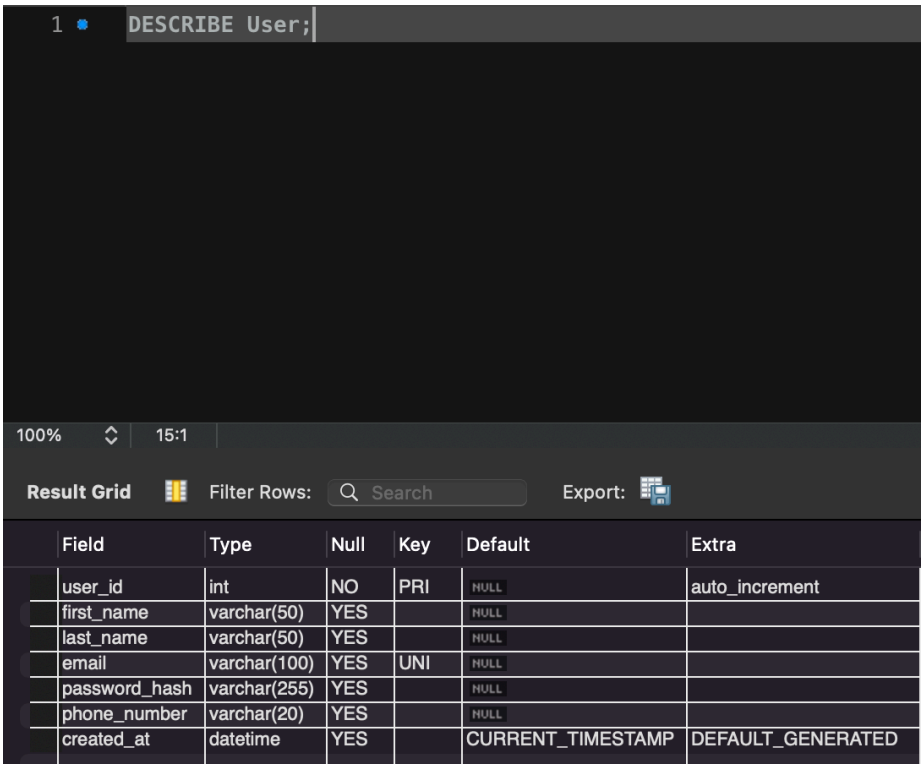
This confirms that all required entities were implemented in the relational schema.

1.2 Table Structure Verification

The structure of core tables was verified using the DESCRIBE command to confirm:

- Primary keys
- Null constraints
- Auto-increment properties
- Data types
- Default values

User Table



Field	Type	Null	Key	Default	Extra
user_id	int	NO	PRI	NULL	auto_increment
first_name	varchar(50)	YES		NULL	
last_name	varchar(50)	YES		NULL	
email	varchar(100)	YES	UNI	NULL	
password_hash	varchar(255)	YES		NULL	
phone_number	varchar(20)	YES		NULL	
created_at	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

The User table includes:

- Primary Key: user_id

- Unique constraint on email
- Automatic timestamp generation for created_at

FamilyMember Table

```
1   DESCRIBE FamilyMember;
2
```

100% 1:2

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
member_id	int	NO	PRI	NULL	auto_increment
user_id	int	NO	MUL	NULL	
first_name	varchar(50)	NO		NULL	
last_name	varchar(50)	NO		NULL	
date_of_birth	date	NO		NULL	
relationship	varchar(50)	YES		NULL	
contact_phone	varchar(20)	YES		NULL	
medical_history	text	YES		NULL	
blood_type	enum('A+','A-','B+','B-','AB+','AB-','O+','O-')	YES		NULL	
gender	enum('Male','Female')	YES		NULL	
status	varchar(30)	YES		NULL	

The FamilyMember table includes:

- Primary Key: member_id
- Foreign key reference to User
- Enum constraints for blood_type and gender
- Proper nullability rules for optional attributes

MedicalHistory Table

```
1  DESCRIBE MedicalHistory;
```

100% 25:1

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
event_id	int	NO	PRI	NULL	auto_increment
member_id	int	NO	MUL	NULL	
condition_id	int	NO	MUL	NULL	
event_date	date	NO		NULL	
event_type	varchar(50)	YES		NULL	
diagnosis	text	YES		NULL	
severity	enum('Low','Medium','High')	YES		NULL	
symptoms	text	YES		NULL	
treatment	text	YES		NULL	
outcome	text	YES		NULL	

The MedicalHistory table includes:

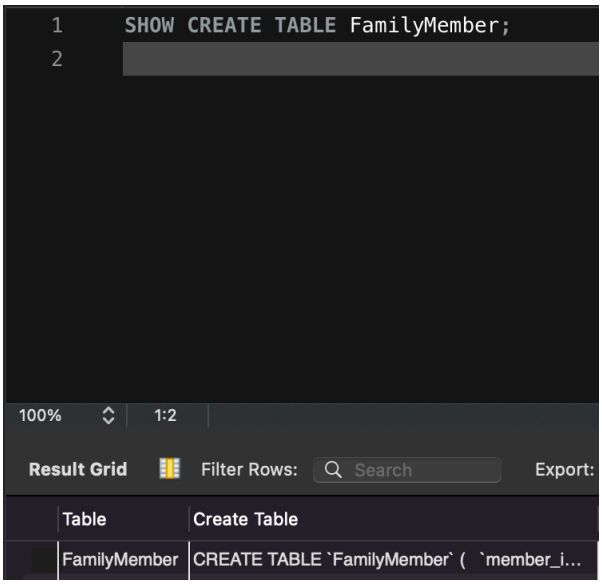
- Primary Key: event_id
- Foreign key references to:
    - FamilyMember
    - HealthCondition
- Enum constraint for severity
- Proper indexing on foreign keys

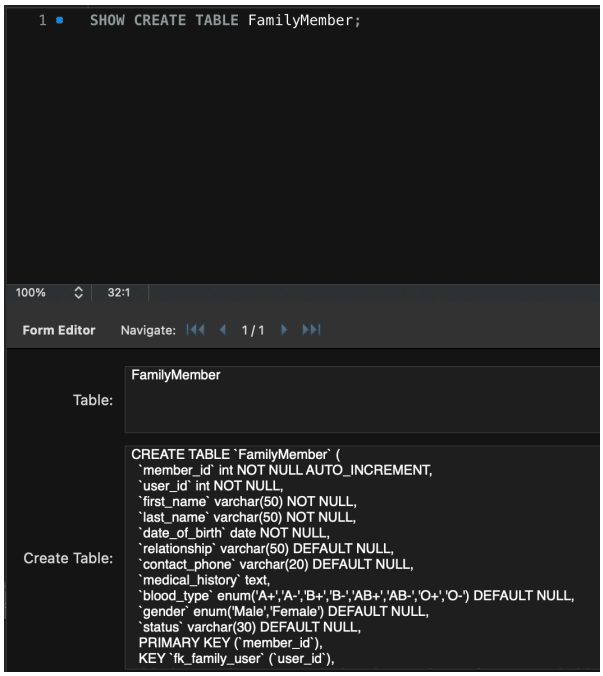
1.3 Foreign Key Constraints and Referential Integrity

Foreign key relationships were verified using the SHOW CREATE TABLE command.

This ensures referential integrity is enforced at the database level.

FamilyMember Foreign Key

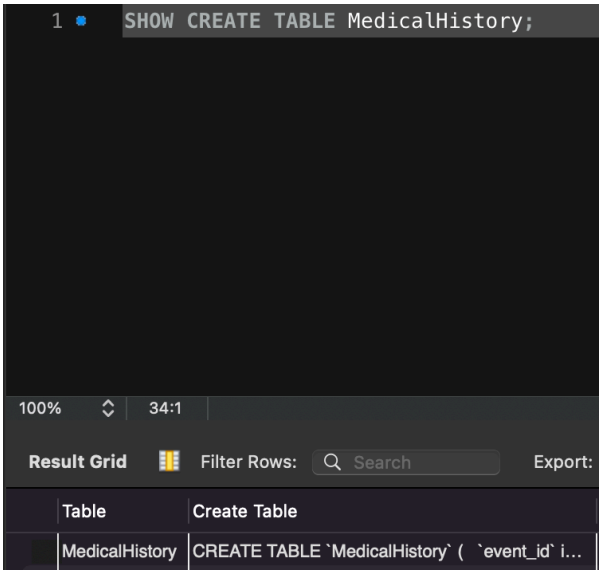


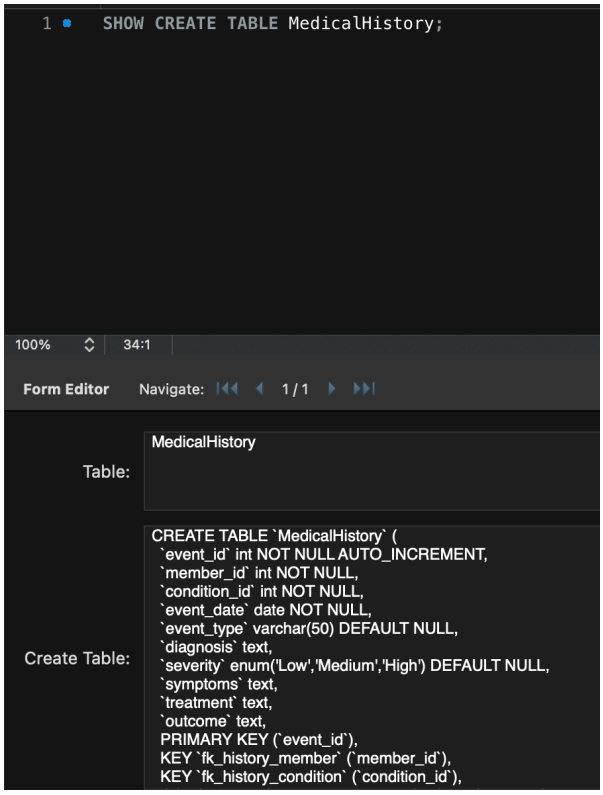


This confirms:

- FamilyMember.user_id references User.user_id
- Cascade rules are correctly applied

MedicalHistory Foreign Keys





This confirms:

- MedicalHistory.member_id references FamilyMember.member_id
- MedicalHistory.condition_id references HealthCondition.condition_id
- Referential integrity is enforced through foreign key constraints

1.4 Data Population Verification

After table creation, sample data was inserted into all tables.

To verify successful data insertion, row counts were retrieved using aggregate queries.

Row Count

```sql
SELECT
  (SELECT COUNT(*) FROM User) AS Users,
  (SELECT COUNT(*) FROM FamilyMember) AS FamilyMembers,
  (SELECT COUNT(*) FROM HealthCondition) AS HealthConditions,
  (SELECT COUNT(*) FROM MedicalHistory) AS MedicalHistory,
  (SELECT COUNT(*) FROM RiskAlert) AS RiskAlerts,
  (SELECT COUNT(*) FROM Clinic) AS Clinics,
  (SELECT COUNT(*) FROM Appointment) AS Appointments,
  (SELECT COUNT(*) FROM AwarenessContent) AS AwarenessContent,
  (SELECT COUNT(*) FROM HealthEvent) AS HealthEvents;
```

100% 1:11

Result Grid Filter Rows: Search Export:

Users	FamilyMembers	HealthConditio...	MedicalHisto...	RiskAlerts	Clinics	Appointmen...	AwarenessContent	HealthEvents
10	20	10	10	10	20	10	10	10

Row count summary:

- Users: 10
- FamilyMembers: 20
- HealthConditions: 10
- MedicalHistory: 10
- RiskAlerts: 10
- Clinics: 20
- Appointments: 10
- AwarenessContent: 10
- HealthEvents: 10

This confirms that all tables are populated and operational.

Conclusion

The database implementation satisfies the project requirements:

- All required entities were created.

- Primary and foreign key constraints were correctly implemented.
- Referential integrity is enforced.
- Appropriate data types and constraints are applied.
- The database is populated with consistent sample data.

The system is fully functional at the relational database level.