# ER Diagram With Reduced Schema

## Project: Online Examination System

### 1. Introduction

This document outlines how the database schema of the Online Examination System was reduced. First, the system had originally been designed with an extensive database with redundant tables, excessive relationships, and unnecessary attributes. We reduced the database through a systematic reduction process to a well-designed and optimized schema.

### 2. Initial Database Schema

The original database design contained more tables and attributes, and that caused redundancy and inefficiency. The main issues were:

- **Duplicate Data:** Having separate tables for student_contacts, faculty_contacts, and admin_contacts when a single contact attribute would suffice.
- **Unnecessary Many-to-Many Relationships:** Additional exam_student_mapping table where it may be integrated with Exam_Attempt.
- **Redundant Attributes:** Storing exam_status in both exams and Exam_Attempt.
- **Excessive Normalization:** Questions and their choices were stored in a separate table question_options, and thus queries became difficult.
- **Unused or Occasionally Used Data:** The system had an exam_schedules table that held venue data, and it was occasionally used.

The initial schema contained the following tables:

- **College**
- **Venue**
- **Admin**
- **Faculty**
- **Student**
- **Student_Contacts**
- **Faculty_Contacts**
- **Admin_Contacts**
- **Exam**
- **Exam_Schedules**
- **Exam_Student_Mapping**
- **Question**

- **Question_Options**
- **Answer**
- **Exam_Attempt**
- **Exam_Availability**
- **System_Logs**

---

**3. Reduction Process**

The database was simplified by applying the following sequential reduction techniques:

**Step 1: Removal of Duplicate Tables**

- student_contacts, faculty_contacts, admin_contacts → Combined into their respective entities (**Student, Faculty, Admin**).
- exam_student_mapping → Merged into Exam_Attempt.
- exam_schedules → Removed because the venue details were combined in **Venue**.

**Step 2: Merging Entities**

- question_options was removed. Options (option_a, option_b, etc.) were instead included as columns in the **Question** table.

**Step 3: Elimination of Redundant Attributes**

- exam_status was not included in **Exams** since it was already included in Exam_Attempt.

**Step 4: Maximizing Many-to-Many Relations**

- The exam_student_mapping table has been removed, and the relationship has been managed by **Exam_Attempt**.

**Step 5: Normalization and Indexing**

- All primary and foreign keys were verified for consistency.
- Indexing was performed where needed (e.g., email columns, foreign keys on questions, etc.).

---

**4. Final (Optimized) Database Schema**

After applying the above optimizations, we arrived at the optimized schema, which includes:

- **College** (college_id, name, address)
- **Venue** (venue_id, name, capacity)
- **Admin** (admin_id, name, email, password, contact)
- **Faculty** (faculty_id, name, email, password, contact, college_id)
- **Student** (student_id, reg_no, name, email, password, contact, college_id)

- **Exam** (exam_id, title, subject, description, total_marks, duration, start_time, end_time, created_by)
- **Question** (question_id, exam_id, question_text, question_type, marks, option_a, option_b, option_c, option_d, correct_answer)
- **Exam_Attempt** (attempt_id, student_id, exam_id, start_time, end_time, status, total_score)
- **Answer** (answer_id, attempt_id, question_id, answer_text, is_correct, marks_obtained)
- **Exam_Availability** (availability_id, exam_id, student_group, available_from, available_until)
- **System_Logs** (log_id, user_id, user_type, activity, ip_address, created_at)

---

**Conclusion**

By reducing redundancy, merging related tables, and eliminating excessive relationships, we have developed a fast and optimized schema. This helps to provide faster queries, better data integrity, and better system performance. This article is a guide to the systematic reduction technique for use in the Online Examination System database.

---

## ER Diagram According to Final Database Schema