



UNIVERSITY OF  
PORTSMOUTH

## **Database Principles (DBPRIN)**

**CW**

**Group 24**

# Database Systems Development

## M21269

CW Group contribution statement

### GROUP 24

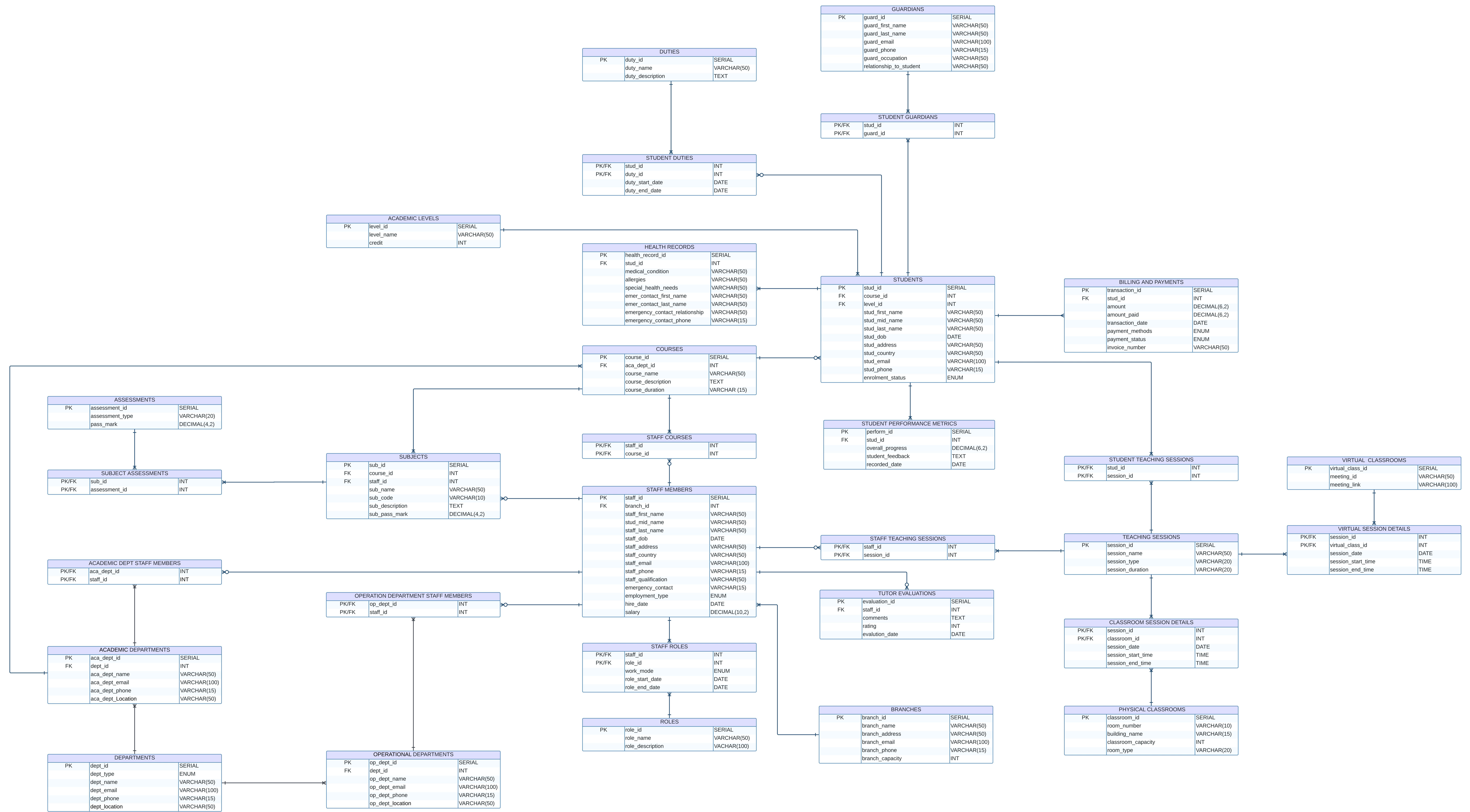
	UP2158902	UP2209423	UP2162181
ERD Design	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Assumptions	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Data Dictionary	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Coding (CREATE+INSERT)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Queries	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Theoretical Aspects	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Document writing	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Brainstorming	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
DDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Percentage Allocation	33.3%	33.3%	33.3%
<b>Signature</b>	<b>Augustine</b>	<b>Oliver</b>	<b>Abdelrhman</b>

**Note<sup>1</sup>:** The contribution is not necessary only "hands on". The contribution can be as research, brainstorming, tests, diagrams, coding etc. For DOCUMENT WRITING one person should write (to maintain the same writing style and general layout) but all members should contribute with ideas, moderation or research.

**Note<sup>2</sup>:** The Percentage Allocation between team members **cannot exceed 100%**.

**Note<sup>3</sup>:** For submission without Group Contribution Statement (or blank) it will be considered that all members are contributed equally.

# ERD



## ASSUMPTIONS

### ASSUMPTIONS FOR STELLAR EDUCATION SOLUTIONS (SES)

In creating the database for Stellar Education Solutions (SES), several important assumptions were made to ensure the system is strong, flexible, and suitable for the organization's goals. These assumptions shape the design of the database, affecting how data is connected, how rules are applied, and how information can be quickly accessed.

#### Academic Staff and Roles

Academic staff members, or tutors, are assigned by school management based on roles defined after gathering feedback from students. Consequently, the "tutor\_evaluations" table is not directly linked to the "students" table. Staff members can hold multiple roles, but each role has a defined start and end date. This aligns with SES's policy of contract-based staffing, where roles may be renewed or terminated as needed. Academic staff operate within the "Academic Department" and are directly involved with students' academic activities. By contrast, operational staff, such as IT support and lab technicians, are part of the "Operational Department" and are not directly involved in academic activities. This clear separation ensures that academic and operational roles remain distinct.

#### Student Information

The database also incorporates key assumptions regarding student information. The guardian information and emergency contact information of students are treated as separate entities, as a guardian (e.g., a parent) may differ from an emergency contact (e.g., a family friend). To support this, the "health records" table is designed to capture comprehensive health information for every student. Health records are mandatory and cannot be NULL, as this information is essential for SES's health and safety protocols. When no specific health issues exist, the field is populated with "none" to ensure data completeness. Additionally, students can only hold one role at a time, eliminating the need for a many-to-many relationship between students and roles.

#### Course and Student Relationships

Courses are designed to accommodate future growth, allowing for zero or many students. This design choice ensures that new courses can be created in the database before student enrollment. Similarly, courses can exist without subjects, allowing the institution to create placeholder course records in preparation for future curriculum development. This approach facilitates long-term scalability and flexibility as SES continues to expand its educational offerings.

#### Teaching Sessions

Each course offers two types of teaching sessions online and offline both of which are recorded for reference purposes. Sessions can exist even without enrolled students, as some courses may be available for future student enrollment. Both physical and virtual classrooms are linked to teaching sessions, ensuring a comprehensive record of

all teaching activities. This approach supports SES's blended learning model and ensures that both online and offline learning experiences are documented.

## Educational Levels

SES currently offers bachelor's (Levels 4-6) and master's (Level 7) programs. However, the database is designed to support future expansion to other educational levels. This assumption positions the system for growth, enabling the institution to offer additional programs in the future without requiring significant changes to the database structure.

## Billing and Payments

The billing and payment process at SES is designed to be flexible, reflecting real-world payment behaviors. Student billing is based not only on tuition fees but also on additional charges such as book fees. Payment records are maintained in the "billing and payments" table. Students can pay their fees after sessions have started, and the "amount\_paid" field can have a value of zero or greater. Check constraints are applied to ensure non-negative values, allowing for the identification of students with outstanding balances.

## Platform Usage

SES uses a single platform, assumed to be Steller's own system. Because of this, there is no need for an "online platform" table in the database. The platform helps manage multiple virtual links for teaching sessions, but it does not create them. There are 10 fixed meeting links available, and academic staff or coordinators schedule online sessions using these links. This approach keeps the system simple and more efficient.

## Many-to-Many Relationships

Several many-to-many relationships are essential to the database's design. These relationships are managed using join tables, ensuring data integrity and facilitating queries. The key many-to-many relationships are as follows:

- **Students and Guardians:** A student may have multiple guardians, and a guardian may be linked to multiple students. This is managed through the "student\_guardians" table.
- **Staff and Roles:** Staff members can hold multiple roles, and each role can be assigned to multiple staff members. The "staff\_roles" table captures this relationship.
- **Students and Teaching Sessions:** Students may attend multiple teaching sessions, and each session may include multiple students. This is represented by the "student\_teaching\_sessions" table.
- **Staff and Teaching Sessions:** Similar to students, staff may be assigned to multiple teaching sessions. The "staff\_teaching\_sessions" table manages this association.
- **Courses and Staff:** Staff members can be assigned to multiple courses, and each course may have multiple instructors. This many-to-many relationship is captured through the "staff\_courses" table.
- **Subjects and Assessments:** Subjects can have multiple assessments, and an assessment can be linked to multiple subjects. The "subject\_assessments" table facilitates this relationship.

- **Students and Duties:** Students can have multiple duties, and each duty can be assigned to multiple students. This many-to-many relationship is managed through a linking table that connects students with their assigned duties.
- **Academic Department and Staff Members:** Staff members can belong to multiple academic departments, and each department can have multiple staff members. This many-to-many relationship allows for flexible staff assignments across departments.
- **Operational Department and Staff Members:** Similar to academic departments, operational departments can have multiple staff members, and staff members can be part of multiple operational departments. This relationship supports dynamic staff allocation across operational units.
- **Physical Classroom and Teaching Sessions:** Teaching sessions can take place in multiple physical classrooms, and each classroom can host multiple sessions. This many-to-many relationship allows for efficient classroom scheduling and resource utilization.
- **Virtual Classroom and Teaching Sessions:** Virtual teaching sessions can use multiple virtual classrooms (or links), and each virtual classroom can be used for multiple sessions. This many-to-many relationship supports online learning flexibility and efficient use of available resources.

## Data Constraints

The database employs various constraints to ensure data integrity. Non-null constraints are applied to essential fields like health records and billing records, ensuring that crucial information is always available. Unique constraints are applied to identification fields like staff emails, guardian emails, and invoice numbers to prevent duplicates. Additionally, check constraints are applied to ensure the validity of specific fields, such as ensuring "amount\_paid" is non-negative.

## Reason for Separate Tables for Departments, Academic Departments, and Operational Departments

The **Department** table serves as a master table containing general information about all departments. The **Academic Department** and **Operational Department** tables are specialized extensions of the main Department table.

- **The Academic Department** focuses on departments directly related to teaching and learning.
- **The Operational Department** covers non-academic units that support day-to-day operations, like IT, HR, or Facilities.

This separation allows for better organization, flexibility, and easier reporting on academic vs. operational activities while maintaining a unified structure under the broader Department entity.

# DATA DICTIONARY

GUARDIANS						
<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
guard_id	PK		SERIAL			
guard_first_name			VARCHAR(50)	NOT NULL		
guard_last_name		Y	VARCHAR(50)	NOT NULL		
guard_email		Y	VARCHAR(100)	NOT NULL; UNIQUE		
guard_phone			VARCHAR(15)	NOT NULL; UNIQUE		
guard_occupation			VARCHAR(50)	NOT NULL		
relationship_to_student			VARCHAR(50)			Relationship of the guardian to the student (e.g., Parent, Aunt).

## BRANCHES

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
branch_id	PK		SERIAL			
branch_name			VARCHAR(50)	NOT NULL		
branch_address			VARCHAR(50)	NOT NULL		
branch_email		Y	VARCHAR(100)	NOT NULL; UNIQUE		
branch_phone			VARCHAR(15)	NOT NULL; UNIQUE		
branch_capacity			INT	NOT NULL		Maximum capacity of a branch

## DEPARTMENTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
dept_id	PK		SERIAL			
dept_type			ENUM	NOT NULL		('operational_dept', 'academic_dept')
dept_name			VARCHAR(50)	NOT NULL		
dept_email			VARCHAR(100)	NOT NULL; UNIQUE		
dept_phone			VARCHAR(15)	NOT NULL; UNIQUE		
dept_location			VARCHAR(50)	NOT NULL		Location of department within a branch

## ACADEMIC LEVELS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
level_id	PK		SERIAL			
level_name			VARCHAR(50)	NOT NULL		Name of an academic level (e.g., Level 4, Level 5)
credit			INT	NOT NULL		Number of credits required for an academic level

## ROLES

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
role_id	PK		SERIAL			
role_name			VARCHAR(50)	NOT NULL		Duty of a staff member
role_description			VARCHAR(100)	NOT NULL		Detailed description of the responsibilities associated with the role

## VIRTUAL CLASSROOMS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
virtual_class_id	PK		SERIAL			
meeting_id			VARCHAR(50)	NOT NULL		Unique meeting ID for the virtual session.
meeting_link			VARCHAR(100)	NOT NULL		URL/link to access the virtual classroom.

## ASSESSMENTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
assessment_id	PK		SERIAL			
assessment_type			VARCHAR(20)	NOT NULL		Type of assessment (e.g., Quiz, Exam).
pass_mark			DECIMAL(4,2)	NOT NULL		Minimum passing mark for an assessment.

## DUTIES

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
duty_id	PK		SERIAL			
duty_name			VARCHAR(50)	NOT NULL		Name of the role assigned to a student.
duty_description			TEXT	NOT NULL		Detailed description of the responsibilities associated with the duty

## ACADEMIC DEPARTMENTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
aca_dept_id	PK		SERIAL			
dept_id	FK		INT	NOT NULL	departments.dept_id	
aca_dept_name			VARCHAR(50)	NOT NULL		
aca_dept_email		Y	VARCHAR(100)	NOT NULL; UNIQUE		
aca_dept_phone			VARCHAR(15)	NOT NULL; UNIQUE		
aca_dept_location			VARCHAR(50)	NOT NULL		Location of an academic department

## COURSES

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
course_id	PK		SERIAL			
aca_dept_id	FK		INT	NOT NULL	academic_departments.aca_dept_id	
course_name			VARCHAR(50)	NOT NULL		
course_description			TEXT	NOT NULL		Detailed description of the course
course_duration			VARCHAR(15)	NOT NULL		Duration of the course in weeks, months, or years

## STAFF MEMBERS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
staff_id	PK		SERIAL			
branch_id	FK		INT	NOT NULL	branches.branch_id	
staff_first_name		Y	VARCHAR(50)	NOT NULL		
staff_mid_name			VARCHAR(50)			
staff_last_name		Y	VARCHAR(50)	NOT NULL		
staff_dob			DATE	NOT NULL		
staff_address			VARCHAR(50)	NOT NULL		
staff_country			VARCHAR(50)	NOT NULL		
staff_email		Y	VARCHAR(100)	NOT NULL; UNIQUE		
staff_phone			VARCHAR(15)	NOT NULL; UNIQUE		
staff_qualification			VARCHAR(50)	NOT NULL		Educational qualifications or degrees held by a staff member
emergency_contact			VARCHAR(15)	NOT NULL		Emergency phone number for a staff
employment_type			ENUM	NOT NULL		Type of employment (e.g., full-time, part-time)
hire_date			DATE	NOT NULL		Date a staff member was hired
salary			DECIMAL(10,2)	NOT NULL		Salary of a staff member

## PHYSICAL CLASSROOMS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
classroom_id	PK		SERIAL			
room_number			VARCHAR(10)	NOT NULL		Room number for each classroom
building_name			VARCHAR(15)	NOT NULL		Location of a building within a branch
classroom_capacity			INT	NOT NULL		Maximum number of occupants a classroom can accommodate
room_type			VARCHAR(20)	NOT NULL		Type of classroom (e.g., Lecture Hall, Lab).

## SUBJECTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
sub_id	PK		SERIAL			
course_id	FK		INT	NOT NULL	courses.course_id	
staff_id	FK		INT	NOT NULL	staff_members.staff_id	
sub_name			VARCHAR(50)	NOT NULL		
sub_code			VARCHAR(10)	NOT NULL		Unique code for each subject
sub_description			TEXT	NOT NULL		Detailed description of each subject
sub_passmark			DECIMAL(4,2)	NOT NULL		Minimum passing mark for each subject

## OPERATIONAL DEPARTMENTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
op_dept_id	PK		SERIAL			
dept_id	FK		INT	NOT NULL	departments.dept_id	
op_dept_name			VARCHAR(50)	NOT NULL		
op_dept_email		Y	VARCHAR(100)	NOT NULL; UNIQUE		
op_dept_phone			VARCHAR(15)	NOT NULL; UNIQUE		
op_dept_location			VARCHAR(50)	NOT NULL		Location of an operational department

## STUDENTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
stud_id	PK		SERIAL			
course_id	FK		INT	NOT NULL	courses.course_id	
level_id	FK		INT	NOT NULL	academic_level.level_id	
stud_first_name		Y	VARCHAR(50)	NOT NULL		
stud_mid_name			VARCHAR(50)			
stud_last_name		Y	VARCHAR(50)	NOT NULL		
stud_dob			DATE	NOT NULL		Date of birth of a student
stud_address			VARCHAR(50)	NOT NULL		
stud_country			VARCHAR(50)	NOT NULL		
stud_email		Y	VARCHAR(100)	NOT NULL; UNIQUE		
stud_phone			VARCHAR(15)	NOT NULL; UNIQUE		Student's phone number
enrolment_status			ENUM	NOT NULL		('Enrolled', 'On Hold', 'Withdrawn')

## TEACHING SESSIONS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
session_id	PK		SERIAL			
session_name			VARCHAR(50)	NOT NULL		
session_type			VARCHAR(20)	NOT NULL		Type of session (e.g., Lecture, Lab, Workshop, Practical, Seminar).
session_duration			VARCHAR(20)	NOT NULL		Duration of a session

## TUTOR EVALUATIONS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
evaluation_id	PK		SERIAL			
staff_id	FK		INT	NOT NULL	staff_members.staff_id	
comments			TEXT	NOT NULL		Comments or feedback provided about a tutor / lecturer (after a certain period of time e.g termly)
rating			INT	NOT NULL		Numeric rating for a tutor performance (e.g., 1-10 scale)
evaluation_date			DATE	NOT NULL		

## STUDENT PERFORMANCE METRICS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
perform_id	PK		SERIAL			
stud_id	FK		INT	NOT NULL	students.stud_id	
overall_progress			DECIMAL(6,2)	NOT NULL		Overall progress of a student (e.g., percentage)
student_feedback			VARCHAR(100)	NOT NULL		General aspects of student's academic life
recorded_date			DATE	NOT NULL		

## HEALTH RECORDS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
health_record_id	PK		SERIAL			
stud_id	FK		INT	NOT NULL	students.stud_id	
medical_condition			VARCHAR(50)	NOT NULL		Specific medical conditions of a student
allergies			VARCHAR(50)	NOT NULL		
special_health_needs			VARCHAR(50)	NOT NULL		Unique health requirement(s) or condition(s) a student may have or need
emer_contact_first_name		Y	VARCHAR(50)	NOT NULL		
emer_contact_last_name			VARCHAR(50)	NOT NULL		
emergency_contact_relationship			VARCHAR(50)	NOT NULL		
emergency_contact_phone		Y	VARCHAR(15)	NOT NULL; UNIQUE		

## BILLING AND PAYMENTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
transaction_id	PK		SERIAL			
stud_id	FK		INT	NOT NULL	students.stud_id	
amount			DECIMAL(6,2)	NOT NULL		Total amount to be paid
amount_paid			DECIMAL(6,2)	NOT NULL		Amount already paid by each student (most up-to-date amount)
transaction_date			DATE	NOT NULL		
payment_methods			ENUM	NOT NULL		('Credit Card', 'Debit Card', 'Bank Transfer')
payment_status			ENUM	NOT NULL		('Pending', 'Completed')
invoice_number		Y	VARCHAR(50)	NOT NULL; UNIQUE		Unique invoice number for tracking

## STAFF ROLES

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
staff_id	PK/FK		INT	NOT NULL	staff_members.staff_id	
role_id	PK/FK		INT	NOT NULL	roles.role_id	
work_mode			ENUM	NOT NULL; DEFAULT		ENUM to specify if the role is Online, Offline, or Hybrid.
role_start_date			DATE	NOT NULL		
role_end_date			DATE	NOT NULL		

## STAFF COURSES

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
staff_id	PK/FK		INT	NOT NULL	staff_members.staff_id	
course_id	PK/FK		INT	NOT NULL	courses.course_id	

## STUDENT GUARDIANS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
stud_id	PK/FK		INT	NOT NULL	students.stud_id	
guard_id	PK/FK		INT	NOT NULL	guardians.guard_id	

## STUDENT TEACHING SESSIONS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
stud_id	PK/FK		INT	NOT NULL	students.stud_id	
session_id	PK/FK		INT	NOT NULL	sessions.session_id	

## STAFF TEACHING SESSIONS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
staff_id	PK/FK		INT	NOT NULL	staff_members.staff_id	
session_id	PK/FK		INT	NOT NULL	sessions.session_id	

## ACADEMIC DEPT STAFF MEMBERS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
aca_dept_id	PK/FK		INT	NOT NULL	academic_departments.aca_dept_id	
staff_id	PK/FK		INT	NOT NULL	staff_members.staff_id	

## OPERATION DEPARTMENT STAFF MEMBERS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
op_dept_id	PK/FK		INT	NOT NULL	operational_departments.op_dept_id	
staff_id	PK/FK		INT	NOT NULL	staff_members.staff_id	

## STUDENT DUTIES

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
stud_id	PK/FK		INT	NOT NULL	students.stud_id	
duty_id	PK/FK		INT	NOT NULL	duties.duty_id	
duty_start_date			DATE	NOT NULL		
duty_end_date			DATE	NOT NULL		

## SUBJECT ASSESSMENTS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
sub_id	PK/FK		INT	NOT NULL	subjects.sub_id	
assessment_id	PK/FK		INT	NOT NULL	assessments.assessment_id	

## CLASSROOM SESSION DETAILS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
session_id	PK/FK		INT	NOT NULL	teaching_sessions.session_id	
classroom_id	PK/FK		INT	NOT NULL	physical_classrooms.classroom_id	
session_date			DATE	NOT NULL		Date when a physical session occurs
session_start_time			TIME	NOT NULL		Start time of a session
session_end_time			TIME	NOT NULL		End time of a session

## VIRTUAL SESSION DETAILS

<i>Attribute_Name</i>	<i>KEY</i>	<i>INDEX</i>	<i>Data Type &amp; Size</i>	<i>Domains &amp; Constraints</i>	<i>FK Reference</i>	<i>Description</i>
session_id	PK/FK		INT	NOT NULL	teaching_sessions.session_id	
virtual_class_id	PK/FK		INT	NOT NULL	virtual_classrooms.virtual_class_id	
session_date			DATE	NOT NULL		Date when an online session occurs
session_start_time			TIME	NOT NULL		Start time of a session
session_end_time			TIME	NOT NULL		End time of a session

## QUERY OUTPUT AND DESCRIPTION

### Query 1:

#### Description:

The query supports financial decisions by providing data on fees paid, outstanding balances, and payment statuses, enabling timely follow-ups and improved cash flow management. It supports operational decisions by highlighting course enrollment trends and student statuses, aiding in resource allocation and course planning.

```
--Query 1
-- Retrieves student enrollment, course, and payment details to support financial and
operational decisions

SELECT
    CONCAT(s.stud_first_name, ' ', s.stud_last_name) AS "Students",
    c.course_name AS "Students' Courses",
    s.enrolment_status AS "Enrolment Status",
    bp.amount AS "Total Fee",
    bp.amount_paid AS "Amount Paid",
    bp.transaction_date AS "Payment Date",
    bp.payment_status AS "Payment Status",
    (bp.amount - bp.amount_paid) AS "Amount Left"
FROM
    students s
INNER JOIN courses c ON s.course_id = c.course_id
INNER JOIN billing_and_payments bp ON s.stud_id = bp.stud_id
WHERE
    EXISTS (
        SELECT 1
        FROM academic_levels al
        WHERE al.level_id = s.level_id
        AND al.level_name ILIKE 'level 4'
    )
ORDER BY
    "Students" ASC;
```

**Output:**

Students	Students' Courses	Enrolment Status	Total Fee	Amount Paid	Payment Date	Payment Status	Amount Left
Amelia Garcia	Cybersecurity	Enrolled	1000.00	500.00	2024-02-10	Pending	500.00
Archer Lewis	Pharmacy	Enrolled	1500.00	1500.00	2024-02-12	Completed	0.00
Augustine Gyamprah	Software Engineering	Enrolled	3000.00	3000.00	2024-02-01	Completed	0.00
Ava Thompson	Data Science	Enrolled	1500.00	1500.00	2024-02-14	Completed	0.00
Charlotte Brown	Software Engineering	Enrolled	3000.00	3000.00	2024-02-05	Completed	0.00
Ethan Perez	Civil Engineering	Enrolled	4500.00	4500.00	2024-02-09	Completed	0.00
Grace Gyamprah	Nursing	Enrolled	2500.00	1300.00	2024-02-04	Pending	1200.00
Isabella Thomas	Data Science	Enrolled	1500.00	1500.00	2024-02-11	Completed	0.00
Jackson Lee	English Literature	Enrolled	3000.00	2000.00	2024-02-08	Pending	1000.00
Jaxon Carter	Nursing	Enrolled	2500.00	2000.00	2024-02-12	Pending	500.00
Leo Harris	Pharmacy	Enrolled	1500.00	1000.00	2024-02-13	Pending	500.00
Lily Johnson	Political Science	Enrolled	1500.00	950.00	2024-02-15	Pending	550.00
Lily Scott	Software Engineering	Enrolled	3000.00	1500.00	2024-02-02	Pending	1500.00
Logan King	Political Science	Enrolled	1500.00	1500.00	2024-02-10	Completed	0.00
Luca Scott	Nursing	Withdrawn	2500.00	2500.00	2024-02-07	Completed	0.00
Lucas Anderson	Data Science	Withdrawn	1500.00	1500.00	2024-02-10	Completed	0.00
Luna White	Renewable Energy	Enrolled	2000.00	1000.00	2024-02-04	Pending	1000.00
Mason Jackson	Renewable Energy	Enrolled	2000.00	2000.00	2024-02-06	Completed	0.00
Mia Gonzalez	Civil Engineering	Enrolled	4500.00	4500.00	2024-02-12	Completed	0.00
Noah Jones	Cybersecurity	Enrolled	1000.00	1000.00	2024-02-10	Completed	0.00
Oliver Martinez	Cybersecurity	Enrolled	1000.00	700.00	2024-02-07	Pending	300.00
Oliver Williams	Software Engineering	Enrolled	3000.00	3000.00	2024-02-01	Completed	0.00
Sofia Young	Political Science	Enrolled	1500.00	1500.00	2024-02-16	Completed	0.00
Sophia Rodriguez	English Literature	Enrolled	3000.00	3000.00	2024-02-05	Completed	0.00
Zoe Clark	Pharmacy	On Hold	1500.00	1500.00	2024-02-03	Completed	0.00
(25 rows)							

**Query 2:**
**Description:**

The query segregates staff members based on their salary, helping the business to understand payroll distribution and identify high-cost or underpaid talent, which supports decisions on salary adjustments and payroll redistribution.

```
--Query 2 - view
-- Retrieves staff details, including roles, departments, and salary categories, to support
workforce and payroll management decisions
CREATE VIEW StaffDetails AS
SELECT
    CONCAT(sm.staff_first_name, ' ', sm.staff_last_name) AS "Staff Name",
    sm.staff_email AS "Email",
    sm.staff_qualification AS "Qualification",
    sm.salary AS "Salary ($)",
    STRING_AGG(DISTINCT ad.aca_dept_name, ', ') AS "Departments",
    STRING_AGG(DISTINCT r.role_name, ', ') AS "Roles",
    b.branch_name AS "Branch Name",
CASE
    WHEN sm.salary > 6000 THEN 'High Earner'
    WHEN sm.salary BETWEEN 5000 AND 5500 THEN 'Mid Earner'
    ELSE 'Low Earner'
END AS "Salary Category"
FROM staff_members sm
JOIN academic_dept_staff_members am ON sm.staff_id = am.staff_id
JOIN academic_departments ad ON am.aca_dept_id = ad.aca_dept_id
JOIN staff_courses sc ON sm.staff_id = sc.staff_id
JOIN courses c ON sc.course_id = c.course_id
JOIN staff_roles sr ON sm.staff_id = sr.staff_id
```

```

JOIN roles r ON sr.role_id = r.role_id
JOIN branches b ON sm.branch_id = b.branch_id
GROUP BY sm.staff_id, b.branch_name
ORDER BY
    "Staff Name";

-- Retrieve View Table
SELECT * FROM StaffDetails;

```

### **Output:**

Staff Name	Email	Qualification	Salary (\$)	Departments	Roles	Branch Name	Salary Category
Adam Evans	adam.evans@ses.co.uk	Master of Business Administration	5300.00	Department of Business	Online Instructor, Tutor	Manchester Branch	Mid Earner
Alexander Gray	alexander.gray@ses.co.uk	Master of Business Administration	5300.00	Department of Business, Department of Mathematics	Lecturer, Manager	Leeds Branch	Mid Earner
Alice Taylor	alice.taylor@ses.co.uk	PhD in Artificial Intelligence	6200.00	Department of Computer Science, Department of Engineering	Lecturer, Manager	Portsmouth Branch	High Earner
Chloe Richmond	chloe.richmond@ses.co.uk	PhD in Chemistry	6100.00	Department of Economics	Lab Technician	Birmingham Branch	High Earner
Daniel Wilson	daniel.wilson@ses.co.uk	Master of Finance	5400.00	Department of Business, Department of Economics	Online Instructor, Lecturer	London Branch	Mid Earner
Elijah Taylor	elijah.taylor@ses.co.uk	Bachelor of Engineering	3300.00	Department of Computer Science, Department of Mathematics	Online Instructor, Tutor	Portsmouth Branch	Low Earner
Ethan King	ethan.king@ses.co.uk	PhD in Marine Biology	6000.00	Department of Natural Sciences	Lecturer, Tutor	Birmingham Branch	Low Earner
Grace Brown	grace.brown@ses.co.uk	MSc in Nursing	5300.00	Department of Health Sciences	Lecturer, Manager	Manchester Branch	Mid Earner
Gwen Williams	gwen.williams@ses.co.uk	MSc in Veterinary Science	4800.00	Department of Environmental Studies	Project Manager, Lecturer, Tutor	London Branch	Mid Earner
James Clark	james.clark@ses.co.uk	PhD of Chemistry	6000.00	Department of Health Sciences	Head Teacher, Lecturer	Manchester Branch	Low Earner
John Lee	john.lee@ses.co.uk	Diploma in Maintenance Operations	3600.00	Department of Arts and Humanities, Department of Natural Sciences	Event Coordinator, Head Teacher	London Branch	Low Earner
Joshua Patterson	joshua.patterson@ses.co.uk	MSc in Information Systems	3200.00	Department of Medicine	Course Coordinator	Birmingham Branch	Low Earner
Liam Allen	liam.allen@ses.co.uk	MSc in Psychology	6200.00	Department of Computer Science, Department of Engineering	Lecturer, Manager	London Branch	Mid Earner
Lucas Bailey	lucas.bailey@ses.co.uk	MSc in Environmental Studies	3500.00	Department of Environmental Studies	Tutor	Manchester Branch	Low Earner
Matthew Perry	matthew.perry@ses.co.uk	Bachelor of Biology	5500.00	Department of Medicine, Department of Natural Sciences	Lab Technician, Research Assistant	Birmingham Branch	Mid Earner
Michael Brown	michael.brown@ses.co.uk	Master in Software Engineering	5300.00	Department of Computer Science	Lecturer, Research Assistant	Portsmouth Branch	Mid Earner
Oliver Brooks	oliver.brooks@ses.co.uk	Bachelor of Sustainable Energy Engineering	5200.00	Department of Environmental Studies	Project Manager, Lecturer	Birmingham Branch	Mid Earner
Ryan Morris	ryan.morris@ses.co.uk	Bachelor of Chemistry	4200.00	Department of Environmental Studies, Department of Natural Sciences	Lecturer, Tutor	Manchester Branch	Low Earner
Sophia Brown	sophia.brown@ses.co.uk	Master of Education	3100.00	Department of Arts and Humanities, Department of Social Sciences	Lecturer, Tutor	Portsmouth Branch	Low Earner
Sophia Taylor	sophia.taylor@ses.co.uk	Master of Education	4900.00	Department of Economics	Lecturer, Tutor	London Branch	Low Earner
William Stone	william.stone@ses.co.uk	Bachelor of Engineering	5500.00	Department of Computer Science, Department of Engineering	Course Coordinator, Head Teacher	Portsmouth Branch	Mid Earner
(21 rows)							

### **Query 3:**

#### **Description:**

By Comparing the branch capacity and total salary disbursed based on the number of staff and their salaries per branch per branch, the business can monitor and manage payment allocation for branch expenses.

```

-- Query 3
-- Retrieves branch details, including capacity, staff count, and payroll costs, to support
branch resource and financial management decisions
CREATE VIEW BranchStaffDetails AS
SELECT
    b.branch_name AS "Branches",
    b.branch_capacity AS "Capacity",
    COUNT(sm.staff_id) AS "Number of Staff",
    SUM(sm.salary) AS "Total Amount to be Disbursed"
FROM
    branches b
JOIN staff_members sm ON b.branch_id = sm.branch_id
WHERE EXISTS (
    SELECT 1
    FROM staff_members sm_sub
    WHERE sm_sub.branch_id = b.branch_id
)
GROUP BY b.branch_name, b.branch_capacity
ORDER BY
    "Total Amount to be Disbursed";

```

```
-- Retrieve View Table
SELECT * FROM BranchStaffDetails;
```

**Output:**

```
ses_cw2=# -- Retrieve View Table
SELECT * FROM BranchStaffDetails;
   Branches      | Capacity | Number of Staff | Total Amount to be Disbursed
-----+-----+-----+-----
Leeds Branch    |    400   |        9          | 40000.00
London Branch   |    500   |        9          | 43100.00
Birmingham Branch | 400   |        9          | 43500.00
Manchester Branch | 450   |        9          | 45100.00
Portsmouth Branch | 400   |       10          | 48000.00
(5 rows)
```

#### Query 4:

##### Description:

By identifying the courses with high or low enrollment, administrators can make informed decisions about resource allocation, such as hiring instructors, scheduling classes, or managing classroom space.

```
-- Query 4
-- Retrieves course details, academic levels, and student enrollment statistics to support
-- academic planning and resource management decisions.

SELECT
  c.course_name AS "Course",
  c.course_duration AS "Course Duration",
  l.level_name AS "Level",
  COUNT(s.stud_id) AS "Number of Students in each Course"
FROM students s
INNER JOIN courses c ON s.course_id = c.course_id
INNER JOIN academic_levels l ON s.level_id = l.level_id
WHERE EXISTS (
  SELECT 1
  FROM students s_sub
  WHERE s_sub.course_id = c.course_id
)
GROUP BY c.course_name, c.course_duration, l.level_name
ORDER BY
  "Level" ASC;
```

**Output:**

Course	Course Duration	Level	Number of Students in each Course
Civil Engineering	4 Years	Level 4	2
Cybersecurity	3 Years	Level 4	3
Data Science	3 Years	Level 4	3
English Literature	3 Years	Level 4	2
Nursing	3 Years	Level 4	3
Pharmacy	4 Years	Level 4	3
Political Science	3 Years	Level 4	3
Renewable Energy	3 Years	Level 4	2
Software Engineering	3 Years	Level 4	4
Civil Engineering	4 Years	Level 5	2
Cybersecurity	3 Years	Level 5	3
Data Science	3 Years	Level 5	3
English Literature	3 Years	Level 5	2
Nursing	3 Years	Level 5	3
Pharmacy	4 Years	Level 5	3
Political Science	3 Years	Level 5	3
Renewable Energy	3 Years	Level 5	2
Software Engineering	3 Years	Level 5	4
Civil Engineering	4 Years	Level 6	2
Cybersecurity	3 Years	Level 6	3
Data Science	3 Years	Level 6	3
English Literature	3 Years	Level 6	2
Nursing	3 Years	Level 6	3
Pharmacy	4 Years	Level 6	3
Political Science	3 Years	Level 6	3
Renewable Energy	3 Years	Level 6	2
Software Engineering	3 Years	Level 6	4
Civil Engineering	4 Years	Level 7	2
Cybersecurity	3 Years	Level 7	3
Data Science	3 Years	Level 7	3
English Literature	3 Years	Level 7	2
Nursing	3 Years	Level 7	3
Pharmacy	4 Years	Level 7	3
Political Science	3 Years	Level 7	3
Renewable Energy	3 Years	Level 7	2
Software Engineering	3 Years	Level 7	4

(36 rows)

**Query 5:**
**Description:**

This supports educational institutions in monitoring student performance, identifying high achievers and those needing improvement, and enhancing academic support and resource allocation.

```
--Query 5
-- Tracks Level 6 and Level 7 student performance, enrollment, and progress to support
academic evaluation and targeted support.

CREATE VIEW StudentPerformance AS
SELECT
    CONCAT(s.stud_first_name, ' ', s.stud_mid_name, ' ', s.stud_last_name) AS "Student
Name",
    s.stud_email AS "Email",
    s.enrolment_status AS "Status",
    al.level_name AS "Level",
    c.course_name AS "Course",
    spm.overall_progress AS "Progress",
    CASE
        WHEN spm.overall_progress >= 85 THEN 'Excellent'
        WHEN spm.overall_progress BETWEEN 70 AND 84 THEN 'Good'
        WHEN spm.overall_progress BETWEEN 50 AND 69 THEN 'Satisfactory'
        ELSE 'Needs Improvement'
    END AS "Performance Category"
FROM
    students s
INNER JOIN academic_levels al ON s.level_id = al.level_id
INNER JOIN courses c ON s.course_id = c.course_id
INNER JOIN student_performance_metrics spm ON s.stud_id = spm.stud_id
WHERE
    (al.level_name ILIKE 'Level 6' OR al.level_name ILIKE 'Level 7')
    AND EXISTS (
        SELECT 1
        FROM student_performance_metrics spm_sub
        WHERE spm_sub.stud_id = s.stud_id
    )
GROUP BY
    s.stud_id, al.level_name, c.course_name, spm.overall_progress
ORDER BY "Level";

-- Retrieve View Table
SELECT * FROM StudentPerformance;
```

### *Output:*

## SECURITY

### DATABASE SECURITY FOR SES

#### Introduction

Database security is important for protecting sensitive information and preventing unauthorized access in the Stellar Education Solutions (SES) system. As SES expands, the database must efficiently manage access with clearly defined roles and permissions.

#### Security Considerations

To secure the SES database, the following security aspects were considered:

##### 1. Role - Based Access Control(RBAC):

- Assigning roles based on user responsibilities or duties (e.g., managers, students).
- Restricting access to some specific data based on the user's role helps data integrity and reduce risks.

##### 2. Privileges:

- Permissions are applied to tables and views based on operational requirements.
- Sensitive data (e.g., staff\_salary, emergency contacts) is only accessible to authorized roles.

#### Why These Measures Are Necessary

As SES expands, the database must serve different users like managers and tutors, each needing specific data access. Without proper controls, sensitive information could be exposed or altered. Security measures ensure:

- **Confidentiality:** Users only access authorized data.
- **Data Integrity:** Prevents accidental or malicious changes.
- **Efficiency:** Role-based access makes tasks easier and reduces unnecessary data sharing.

#### Roles and Privileges Defined

Two key roles were implemented in the SES database: **ses\_manager** and **ses\_tutor**. Each role was assigned specific privileges.

#### Role: Manager (ses\_manager)

##### 1. Role Created

The **ses\_manager** role supports managers in taking charge over branch operations, managing staff data, session details, and viewing branch insights.

```
-- Role Manager
CREATE ROLE ses_manager WITH LOGIN PASSWORD 's1-ses-1M';
```

Output:

```
CREATE ROLE ses_manager WITH LOGIN PASSWORD 's1-ses-1M';
CREATE ROLE
```

## 2. Privileges Assigned

The **ses\_manager** role was assigned specific privileges to access and modify necessary tables:

**Staff Table:** The manager can read and write data from the staff table

```
-- Privileges on Staff Table
GRANT SELECT, UPDATE ON staff_members TO ses_manager;
```

**Sessions Table:** The manager can read student details.

```
-- Privileges on Students Table
GRANT SELECT ON students TO ses_manager;
```

**Branches Table:** The manager can only view branch details without making changes.

```
-- Privileges on Branches Table (Read-Only)
GRANT SELECT ON branches TO ses_manager;
```

**Staff Details View:** The manager has read-only access to a consolidated view of staff data.

```
-- Grant SELECT privilege on the view to SES_Manager
GRANT SELECT ON StaffDetails TO SES_Manager;
```

*Following the creation of privileges*

```
ses_cw=# -- Privileges on Staff Table
GRANT SELECT, UPDATE ON staff_members TO ses_manager;
-- Privileges on Students Table
GRANT SELECT ON students TO ses_manager;
-- Privileges on Billing and Payment Table
GRANT SELECT, UPDATE ON billing_and_payments TO ses_manager;
-- Privileges on Branches Table (Read-Only)
GRANT SELECT ON branches TO ses_manager;
-- Grant SELECT privilege on the view to SES_Manager
GRANT SELECT ON StaffDetails TO SES_Manager;
GRANT
GRANT
GRANT
GRANT
GRANT
```

## 3. Testing Privileges

Testing if **ses\_manager** can access tables without granting permission.

```
ses_cw=> SELECT * FROM staff_members;
ERROR: permission denied for table staff_members
```

## Testing Read and Update on Staff Table

staff_id	branch_id	staff_first_name	staff_mid_name	staff_last_name	staff_dob	staff_address	staff_country	staff_email	staff_phone	staff_qualification	emergency_contact	employment_type	hire_date	salary
1	1	Alice	Marie	Taylor	1987-03-21	78 Commercial Road, Portsmouth	UK	alice.taylor@ses.co.uk	+4423923455678	PhD in Artificial Intelligence	+447123456789	Full Time	2015-06-01	\$2600.00
2	1	William	Andrew	Stone	1985-05-12	98 Elm Street, Portsmouth	UK	william.stone@ses.co.uk	+442392765432	Bachelor of Engineering	+447898123456	Full Time	2015-06-03	\$500.00
3	1	Grace	Natalie	Evans	1990-01-05	123 Piccadilly, Manchester	UK	grace.evans@ses.co.uk	+44239286779	Master in Management	+447654987654	Part Time	2016-03-01	\$300.00
4	1	Daniel	Christopher	Harris	1978-08-30	123 Cedar Road, Portsmouth	UK	daniel.harris@ses.co.uk	+442392867543	Bachelor of Science	+447654987298	Full Time	2016-03-28	\$400.00
5	1	Olivia	Thomas	Taylor	1990-12-05	67 Pine Street, Portsmouth	UK	olivia.taylor@ses.co.uk	+442392875434	Master in IT Support	+4476549887123	Part Time	2016-03-15	\$300.00
6	1	James	Edward	Johns	1988-09-20	45 Regent Street, London	UK	james.johns@ses.co.uk	+442392875435	Bachelor of Engineering	+447654988755	Part Time	2016-11-15	\$400.00
7	1	Ava	Sophia	King	1988-06-18	67 Park Lane, London	UK	ava.king@ses.co.uk	+442392875432	Bachelor of Education	+447898654521	Full Time	2015-07-01	\$800.00
8	1	Isabella	Henry	Reed	1975-11-15	12 Victoria Street, London	UK	isabella.reed@ses.co.uk	+442392875433	Bachelor of Marketing	+447654987293	Part Time	2016-03-25	\$150.00
9	1	Isla	Isabel	Allen	1979-02-01	56 Piccadilly, Manchester	UK	isla.allen@ses.co.uk	+442392875432	RBA in Operations Management	+447543216579	Full Time	2005-06-18	\$600.00
10	1	Lucas	Flora	Young	1991-07-25	99 Almond Street, Manchester	UK	lucas.young@ses.co.uk	+441612987654	Bachelor of English	+447654987321	Part Time	2016-06-29	\$1100.00
11	2	Tom	Flora	Young	1991-07-25	99 Almond Street, Manchester	UK	grace.evans@ses.co.uk	+441612789012	Master of Library Science	+447654987321	Full Time	2016-04-15	\$800.00
12	2	Grace	Lillian	Evans	1998-12-15	78 Oxford Road, Manchester	UK	lillian.evans@ses.co.uk	+441612789013	Master of Business Administration	+447654987321	Full Time	2016-04-15	\$800.00
13	2	Isaac	Isaac	Evans	1998-12-15	78 Oxford Road, Manchester	UK	lucia.clark@ses.co.uk	+441612095765	PhD in Renewable Energy	+447898654218	Full Time	2016-08-05	\$2600.00
14	2	Lucas	William	Clark	1978-06-28	12 New Street, Manchester	UK	lucia.clark@ses.co.uk	+441612095765	PhD in Renewable Energy	+447898654218	Full Time	2016-08-05	\$2600.00
15	2	Sophia	Emma	Taylor	1998-07-12	56 Oak Street, Birmingham	UK	sophia.taylor@ses.co.uk	+442134567890	Master of Education	+4476549872113	Part Time	2016-03-15	\$400.00
16	2	James	James	Evans	1989-01-01	123 High Street, Birmingham	UK	james.evans@ses.co.uk	+442134567891	Master of Business Management	+4476549872113	Part Time	2016-03-15	\$400.00
17	2	Grace	Isabella	Evans	1987-05-10	45 Commercial Road, Birmingham	UK	grace.isabella@ses.co.uk	+4421124765432	Master of Business Administration	+4478986232789	Full Time	2016-07-18	\$2600.00
18	2	John	Alexander	Lee	1988-10-25	78 New Street, Birmingham	UK	john.lees@ses.co.uk	+4421124321984	Diploma in Maintenance Operations	+447312676432	Part Time	2015-11-01	\$300.00
19	2	Emily	Ashley	Jones	1990-01-01	123 High Street, Birmingham	UK	emily.jones@ses.co.uk	+4421124321985	Diploma in Business Administration	+4476549872113	Part Time	2016-03-15	\$400.00
20	2	Daniel	Lucas	Walker	1993-06-10	98 King Street, Leeds	UK	daniel.walker@ses.co.uk	+441132987653	Bachelor of Business Administration	+4478986232789	Part Time	2015-01-28	\$2600.00
21	2	Olivia	Sophia	Walker	1982-09-14	123 High Street, Liverpool	UK	olivia.walker@ses.co.uk	+441132987654	Diploma in Business Administration	+447312676432	Part Time	2015-01-15	\$400.00
22	2	James	Thomas	Evans	1982-09-14	123 High Street, Liverpool	UK	james.evans@ses.co.uk	+441132987655	Diploma in Business Administration	+4476549872113	Part Time	2016-03-15	\$400.00
23	3	James	Henry	Clark	1978-09-25	56 Bold Street, Liverpool	UK	james.clark@ses.co.uk	+441516789012	PhD of Chemistry	+447654321876	Full Time	2010-04-28	\$600.00
24	3	James	Henry	Clark	1978-09-25	56 Bold Street, Liverpool	UK	james.clark@ses.co.uk	+441516789013	PhD of Chemistry	+447654321876	Full Time	2010-04-28	\$600.00
25	3	Grace	Matthew	Brown	1992-05-16	49 Oxford Road, Liverpool	UK	matthew.brown@ses.co.uk	+441516894765	MSc in Environmental Studies	+447898654123	Part Time	2015-07-29	\$300.00
26	3	Grace	Amelia	Brown	1992-05-16	49 Oxford Road, Liverpool	UK	grace.amelia@ses.co.uk	+441516894765	MSc in Nursing	+447898654123	Part Time	2015-07-29	\$300.00
27	3	Grace	Amelia	Morris	1981-08-05	67 Park Lane, Liverpool	UK	ryan.morris@ses.co.uk	+441516987654	Bachelor of Chemistry	+4476549887123	Part Time	2017-01-01	\$2600.00
28	3	Ryan	Sophia	Patterson	1993-07-08	67 Holywell Street, Oxford	UK	sophia.patterson@ses.co.uk	+441865987654	Diploma in Business Management	+447312676432	Part Time	2016-03-15	\$400.00
29	4	Joshua	Daniel	Patterson	1995-07-08	67 Holywell Street, Oxford	UK	joshua.patterson@ses.co.uk	+441865987654	MSc in Information Systems	+447312676432	Part Time	2020-06-01	\$2600.00
30	4	Sophia	Louise	Griffiths	1987-10-22	78 Woodstock Road, Oxford	UK	sophia.griffiths@ses.co.uk	+441865765432	Master of Business Administration	+4478986232789	Full Time	2019-02-18	\$400.00

Update

```
ses_cw=> UPDATE staff_members
SET
    staff_email = 'alice.t.new@ses.co.uk',
    salary = 6500.00,
    staff_address = '90 High Street, Portsmouth'
WHERE
    staff_first_name = 'Alice'
    AND staff_last_name = 'Taylor'
    AND branch_id = 1;
UPDATE 1
```

## Testing Read-Only Access to Branches Table

branch_id	branch_name	branch_address	branch_email	branch_phone	branch_capacity
1	Portsmouth Branch	78 Commercial Road, Portsmouth PO1 1HG	portsmouth@ses.co.uk	+4423923455678	400
2	London Branch	123 Oxford Street, London W1D 1LT	london@ses.co.uk	+442071234567	500
3	Manchester Branch	56 Piccadilly, Manchester M1 2GH	manchester@ses.co.uk	+441612345678	450
4	Birmingham Branch	12 New Street, Birmingham B2 4QA	birmingham@ses.co.uk	+441214567890	400
5	Leeds Branch	78 Briggate, Leeds LS1 6LH	leeds@ses.co.uk	+441132348765	400

(5 rows)

## Testing View Access of the StaffDetails view.

Staff Name	Email	Qualification	Salary (\$)	Departments	Roles	Branch Name	Salary Category
Adam Evans	adam.evans@ses.co.uk	Master of Business Administration	\$3000.00	Department of Business	Online Instructor, Tutor	Manchester Branch	Mid Earner
Alexander Gray	alexander.gray@ses.co.uk	Master of Business Administration	\$3000.00	Department of Business, Department of Mathematics	Lecturer, Manager	Leeds Branch	Mid Earner
Alice Taylor	alice.t.new@ses.co.uk	PhD in Artificial Intelligence	\$6000.00	Department of Computer Science, Department of Engineering	Lecturer, Manager	Portsmouth Branch	High Earner
Chloe Richmond	chloe.richmond@ses.co.uk	PhD in Chemistry	\$6000.00	Department of Economic	Lab Technician	Birmingham Branch	High Earner
David Williams	dwight.williams@ses.co.uk	Master of Science	\$3000.00	Department of Environmental Sciences	Lab Technician, Lecturer	Portsmouth Branch	Mid Earner
Eliah Taylor	eliah.taylor@ses.co.uk	Bachelor of Engineering	\$3000.00	Department of Computer Sciences, Department of Mathematics	Online Instructor, Tutor	Birmingham Branch	Low Earner
Ethan King	ethan.king@ses.co.uk	PhD in Marine Biology	\$6000.00	Department of Natural Sciences	Lecturer, Tutor	Manchester Branch	Low Earner
Grace Brown	grace.brown@ses.co.uk	MSc in Nursing	\$3000.00	Department of Health Sciences	Lecturer, Manager	Leeds Branch	Mid Earner
Grace Evans	grace.evans@ses.co.uk	Master of Library Science	\$3000.00	Department of Environmental Studies	Department Head, Lecturer, Tutor	London Branch	Mid Earner
Grace Clark	grace.clark@ses.co.uk	PhD of Chemistry	\$6000.00	Department of Health Sciences	Head Teacher, Lecturer	Manchester Branch	Low Earner
John Lee	john.lee@ses.co.uk	Diploma in Maintenance Operations	\$3000.00	Department of Arts and Humanities, Department of Natural Sciences	Event Coordinator, Head Teacher	London Branch	Low Earner
Joshua Patterson	joshua.patterson@ses.co.uk	MSc in Engineering	\$2000.00	Department of Medicine	Course Coordinator	Birmingham Branch	Low Earner
Lucas Allen	lucas.allen@ses.co.uk	PhD in Environmental Studies	\$3500.00	Department of Computer Science, Department of Engineering	Lecturer, Manager	London Branch	High Earner
Lucas Bailey	lucas.bailey@ses.co.uk	MSc in Environmental Studies	\$3500.00	Department of Environmental Studies	Tutor	Manchester Branch	Low Earner

**Testing View Access** the **ses\_manager** role could retrieve data from the **billing and payment**.

transaction_id	stud_id	amount	amount_paid	transaction_date	payment_methods	payment_status	invoice_number
1	1	3000.00	3000.00	2024-02-01	Credit Card	Completed	INV001
2	2	3000.00	1500.00	2024-02-02	Debit Card	Pending	INV002
3	3	3000.00	3000.00	2024-02-01	Bank Transfer	Completed	INV003
4	4	3000.00	3000.00	2024-02-05	Credit Card	Completed	INV004
5	5	1000.00	1000.00	2024-02-10	Debit Card	Completed	INV005
6	6	1000.00	500.00	2024-02-10	Bank Transfer	Pending	INV006
7	7	1000.00	700.00	2024-02-07	Credit Card	Pending	INV007
8	8	3000.00	3000.00	2024-02-05	Debit Card	Completed	INV008
9	9	3000.00	2000.00	2024-02-08	Bank Transfer	Pending	INV009
10	10	4500.00	4500.00	2024-02-12	Credit Card	Completed	INV010
11	11	4500.00	4500.00	2024-02-09	Debit Card	Completed	INV011
12	12	1500.00	1500.00	2024-02-14	Bank Transfer	Completed	INV012
13	13	1500.00	1500.00	2024-02-10	Credit Card	Completed	INV013
14	14	1500.00	1500.00	2024-02-11	Debit Card	Completed	INV014
15	15	2000.00	2000.00	2024-02-06	Bank Transfer	Completed	INV015
16	16	2000.00	1000.00	2024-02-04	Credit Card	Pending	INV016
17	17	1500.00	1000.00	2024-02-13	Debit Card	Pending	INV017
18	18	1500.00	1500.00	2024-02-03	Bank Transfer	Completed	INV018
19	19	1500.00	1500.00	2024-02-12	Credit Card	Completed	INV019
20	20	1500.00	950.00	2024-02-15	Debit Card	Pending	INV020
21	21	1500.00	1500.00	2024-02-16	Bank Transfer	Completed	INV021
22	22	1500.00	1500.00	2024-02-10	Credit Card	Completed	INV022
23	23	2500.00	2500.00	2024-02-07	Debit Card	Completed	INV023
24	24	2500.00	1300.00	2024-02-04	Bank Transfer	Pending	INV024
25	25	2500.00	2000.00	2024-02-12	Credit Card	Pending	INV025
26	26	3000.00	3000.00	2024-02-14	Debit Card	Completed	INV026
27	27	3000.00	1500.00	2024-02-13	Bank Transfer	Pending	INV027
28	28	3000.00	1750.00	2024-02-10	Credit Card	Pending	INV028
29	29	3000.00	3000.00	2024-02-14	Debit Card	Completed	INV029
30	30	1000.00	500.00	2024-02-12	Bank Transfer	Pending	INV030

## Role: Tutor (ses\_tutor)

### 1. Role Created

The **ses\_tutor** role was created to enable teaching staff to access course-related data and record student performance.

Code:

```
-- Role Tutor
CREATE ROLE ses_tutor WITH LOGIN PASSWORD 'H1-ses-1T';
```

Output:

```
CREATE ROLE
ses_cw=#
```

### 2. Privileges Assigned

The ses\_tutor role was granted the following permissions:

**Students Table:** The tutor can read or retrieve student information but cannot modify it.

```
-- Grant read-only access to the Students table
GRANT SELECT ON students TO Tutor;
```

**StudentPerformance View:** The tutor can view summarized performance data for students.

```
-- Grant SELECT (read-only) access to the StudentPerformance view
GRANT SELECT ON StudentPerformance TO Tutor;
```

**Courses Table:** The tutor can read course details relevant to their teaching assignments

```
-- Grant read-only access to the Course table
GRANT SELECT ON courses TO Tutor;
```

**Teaching Sessions Table:** The tutor can view session schedules and other teaching-related data.

```
-- Grant read-only access to the Sessions table
GRANT SELECT ON teaching_sessions TO Tutor;
```

**Student Performance Metrics Table:** The tutor can view and update student performance data, including adding feedback and progress reports.

```
-- Grant SELECT (read) access to the student_performance table
GRANT SELECT ON student_performance_metrics TO Tutor;
-- Grant INSERT (add data) access to the student_performance table
GRANT INSERT ON student_performance_metrics TO Tutor;
GRANT USAGE, SELECT, UPDATE ON SEQUENCE student_performance_metrics_perform_id_seq TO ses_tutor;
```

### *Following the creation of privileges*

```
ses_cw=# --1 Grant read-only access to the Students table
GRANT SELECT ON students TO ses_tutor;
--2 Grant SELECT read-only access to the StudentPerformance view
GRANT SELECT ON StudentPerformance TO ses_tutor;
--3 Grant read-only access to the Course table
GRANT SELECT ON courses TO ses_tutor;
--4 Grant read-only access to the Sessions table
GRANT SELECT ON teaching_sessions TO ses_tutor;
--5 Grant SELECT (read) access to the student_performance table
GRANT SELECT ON student_performance_metrics TO ses_tutor;
--6 Grant INSERT (add data) access to the student_performance table
GRANT INSERT ON student_performance_metrics TO ses_tutor;
GRANT
GRANT
GRANT
GRANT
GRANT
GRANT
```

## Testing Privileges

Testing if ses\_tutor can access tables without explicit grants

```
ses_cw=> SELECT * FROM students;
ERROR: permission denied for table students
```

## Testing Access to StudentsTable.

```
SELECT * FROM students;
```

stud_id	course_id	level_id	stud_first_name	stud_mid_name	stud_last_name	stud_dob	stud_address	stud_country	stud_email	stud_phone	enrolment_s
1	19	1	Augustine	Somiah	Gyamprah	1990-03-12	123 High Street	United Kingdom	gym.auga@email.com	+447911145612	Enrolled
2	19	1	Lily	Grace	Scott	2000-11-09	456 Oak Avenue	Canada	lily.johnson04@email.com	+4419111234567	Enrolled
3	19	1	Oliver	O.	Williams	1997-04-24	789 Maple Road	Australia	oliver.williams@email.com	+4479111345678	Enrolled
4	19	1	Charlotte	Evelyn	Brown	2001-07-14	101 Pine Crescent	United States	charlotte.brown@email.com	+447912456789	Enrolled
5	20	1	Noah	Alexander	Jones	1999-12-17	102 Cedar Street	India	noah.jones@email.com	+44791146789	Enrolled
6	20	1	Amelia		Garcia	2000-02-01	203 Elm Lane	Spain	amelia.garcia02@email.com	+09791167498	Enrolled
7	20	1	Oliver	Benjamin	Martinez	1998-08-22	304 Birch Drive	Mexico	oliver.martinez@email.com	+447981789012	Enrolled
8	1	1	Sophia	Rose	Rodriguez	1999-01-10	505 Fir Road	Germany	sophia.rodriguez0@email.com	+447901890123	Enrolled
9	1	1	Jackson	Liam	Lee	2001-04-03	606 Redwood Avenue	Brazil	jackson.lee@email.com	+497911901234	Enrolled
10	7	1	Mia	Isabelle	Gonzalez	2000-09-19	707 Willow Street	Italy	mia.gonzalez@email.com	+417911012345	Enrolled
11	7	1	Ethan	Matthew	Perez	1997-12-13	888 Chestnut Lane	France	ethan.perez@email.com	+447909123456	Enrolled
12	21	1	Ava	Charlotte	Thompson	1998-05-25	909 Birch Lane	South Africa	ava.thompson@email.com	+4479111234567	Enrolled
13	21	1	Lucas	Henry	Anderson	1999-03-15	1001 Oak Park	Sweden	lucas.anderson@email.com	+4479011345678	Withdrawn
14	21	1	Isabella	Sophie	Thomas	2001-10-15	1111 Pine Street	Finland	isabella.thomas@email.com	+447981456789	Enrolled
15	18	1	Mason	Daniel	Jackson	2000-03-20	1212 Cedar Drive	Denmark	mason.jackson@email.com	+447911578901	Enrolled
16	18	1	Luna	White	White	1998-09-17	1313 Cedar Avenue	Norway	luna.white@email.com	+447911578901	Enrolled
17	12	1	Leo	Sebastian	Harris	1999-02-28	1414 Willow Road	Australia	leo.harris@email.com	+179911789002	Enrolled
18	12	1	Zoe	Claire	Clark	2000-07-25	1515 Birch Crescent	Italy	zoe.clark@email.com	+747911890123	On Hold
19	12	1	Archer	Finn	Lewis	1998-05-13	1616 Redwood Lane	New Zealand	archer.lewis@email.com	+647911012345	Enrolled
20	3	1	Lily	Sophie	Johnson	2001-03-17	1717 Pine Crescent	Russia	lily.walker@email.com	+067911019345	Enrolled
21	3	1	Sofia	Eleanor	Young	1999-11-30	1818 Elm Avenue	Brazil	sofia.young@email.com	+447911363456	Enrolled
22	3	1	Logan	Jacob	King	2000-06-07	1919 Oak Road	France	logan.king@email.com	+447911234568	Enrolled
23	11	1	Luca		Scott	1997-04-04	2020 Pine Lane	South Korea	luca.scott@email.com	+547911345678	Withdrawn

## Testing Access to Courses Table

```
SELECT * FROM courses;
```

ses_cw=> SELECT * FROM courses;	course_id	aca_dept_id	course_name	course_description	course_duration
1	1	1	English Literature	A study of classic and contemporary literary works.	3 Years
2	1	1	History and Culture	Explores historical events and their cultural impacts.	3 Years
3	2	2	Political Science	Explores political theories, government systems, and public policy.	3 Years
4	2	2	Criminology	Study of crime, justice systems, and criminal behavior.	3 Years
5	3	3	Biology	Explores living organisms and ecosystems.	3 Years
6	3	3	Physics	Focuses on the principles of matter, energy, and the universe.	3 Years
7	4	4	Civil Engineering	Learn about construction, infrastructure, and structural design.	4 Years
8	4	4	Electrical Engineering	A program on power systems, electronics, and communication.	4 Years
9	5	5	Business Administration	Covers principles of management, marketing, and finance.	3 Years
10	5	5	Marketing and Advertising	A course on branding, market research, and advertising strategies.	3 Years
11	6	6	Nursing	Prepares students for careers in clinical nursing and patient care.	3 Years
12	6	6	Pharmacy	Study of drug formulation, pharmacology, and patient counseling.	4 Years
13	7	7	Financial Economics	Focus on investment, risk, and financial markets.	3 Years
14	7	7	Public Economics	Examines the role of government in the economy.	3 Years
15	8	8	General Medicine	Comprehensive training for medical diagnosis and treatment.	6 Years
16	8	8	Surgery	Prepares students for careers in surgical fields.	6 Years
17	9	9	Environmental Science	Study of the environment and sustainability practices.	3 Years
18	9	9	Renewable Energy	Covers renewable energy technologies and policies.	3 Years
19	10	10	Software Engineering	A program focused on designing, developing, and testing software systems.	3 Years
20	10	10	Cybersecurity	Learn to protect computer systems and networks from cyber threats.	3 Years
21	10	10	Data Science	An interdisciplinary program that covers data analysis, machine learning, and big data.	3 Years
22	10	10	Artificial Intelligence	Explores AI principles, algorithms, and applications.	3 Years
23	11	11	Pure Mathematics	Explores abstract mathematical theories and principles.	3 Years
24	11	11	Discrete Mathematics	Explores mathematical structures like graphs and algorithms.	3 Years

(24 rows)

Testing to see if ses\_tutor can view and update performance metrics.

```
SELECT * FROM StudentPerformance;
```

perform_id	stud_id	overall_progress	student_feedback	recorded_date
1	1	95.00	Very strong performance academically. Demonstrates leadership skills and excellent communication in group settings.	2024-11-12
2	1	100.00	Exceptional academic performance. Demonstrates excellent leadership, teamwork, and communication skills in all tasks.	2024-11-05
3	2	65.00	Decent performance overall. Needs to work more on group projects and task completion within time.	2024-11-07
4	3	55.00	Decent academic progress. Could work on communication skills and completing assignments on time.	2024-12-12
5	3	77.00	Strong academic progress. Could benefit from more active participation and leadership in group activities.	2024-11-10
6	4	22.00	Major improvement needed. Struggles with participation and completing tasks. Time management needs focus.	2024-12-12
7	4	62.00	Decent performance. Needs to work on consistency and ensuring tasks are completed on time.	2024-11-15
8	5	89.00	Good academic performance. Needs to work on improving time management and collaboration in group settings.	2024-12-12
9	5	80.00	Very good performance academically. Should continue improving teamwork and leadership roles.	2024-11-18
10	6	28.00	Very low performance. Needs immediate improvement in organization, participation, and completing assignments on time.	2024-12-12
11	6	92.00	Excellent performance. Consistently delivers high-quality work and demonstrates leadership skills in group settings.	2024-11-10
12	7	72.00	Good performance. Needs to work on organization of group projects.	2024-11-22
13	8	56.00	Fair performance. Needs to focus more on applying knowledge practically and manage deadlines effectively.	2024-11-23
14	8	40.00	Performance is below expectations. Needs to focus on time management, increasing class participation, and improving task completion consistency.	2024-12-12
15	9	41.00	Below average performance. Needs to work on time management and consistently meeting deadlines.	2024-11-23
16	10	66.00	Solid academic progress. Could benefit from more involvement in class discussions and group tasks.	2024-11-25
17	11	50.00	Average performance. Needs to improve in completing tasks on time and focusing during practical exercises.	2024-11-28
18	12	35.00	Low performance. Immediate improvement needed in engagement and organization to meet expectations.	2024-12-01
19	13	88.00	Very good performance. Needs to focus on taking more leadership roles and improving collaboration in teams.	2024-12-03

Update

```
ses_cw=> INSERT INTO student_performance_metrics (stud_id, overall_progress, student_feedback, recorded_date)
VALUES
(2, 95, 'Very strong performance academically. Demonstrates leadership skills and excellent co mmunication in group settings.', '2024-11-13');
INSERT 0 1
```

## Testing Access to the StudentPerformance View

# OPTIMISATION

## THREE performance-sensitive transactions

1. **Retrieve Tutor Evaluation Records**
  2. **Enrol new students (Add new student)**
  3. **Update Student Payment Status (Amount paid)**

## **TRANSACTION MATRIX TABLE**

## OPTIMISATION

According to Colley, D. A. (2021), one could argue that relational database speed is still a concern today due to extensive use of relational databases in various applications, therefore optimizing performance-sensitive transactions in modern database systems is important for efficient operations. This report assesses the performance demands of three core transactions and proposes techniques to reduce load, improve response-time, and user experience.

### 1. Enroll New Students

Description: Adding a new student to the database involves inserting data into the student table, along with related tables that contain relevant details.

Performance Sensitivity: Adding new students significantly impacts our database during enrolment periods, as Stellar tutors must concurrently add many students, affecting database operations across related tables. These changes need to be immediate to prevent concurrency issues. Moreover, failing to enforce referential integrity can result in invalid records and compromised data integrity.

#### Optimization Strategies

##### 1. Indexing:

Indexes will be created on frequently accessed columns, such as course\_id and stud\_id, to enhance query performance, particularly for insert and retrieval operations. This significantly reduces the time required to locate specific rows during high-volume transactions.

It is important to note that the primary purpose of adding a record is to retrieve it efficiently when needed. By identifying the attributes most frequently queried, appropriate indexes can be applied. Additionally, when a new row is added to the database, the presence of an index ensures that the retrieval process is optimized from the outset. This approach balances the efficiency of data inserts with the speed of subsequent queries, ensuring optimal performance.

### 2. Update Student Payment Status(Amount Paid)

Description: Updating payment records for students, ensuring financial data accuracy and reflecting payment status in real-time.

Performance Sensitivity: During times of high transaction activity, payment records are updated frequently, and when not frequent, we need to make sure that the effects of this transaction are absolutely accurate and performed in real-time, as financial records are not only sensitive but also critical to the business (clients and users alike).

#### Optimization Strategies:

- **Indexes** will be created on frequently accessed columns, such as student\_id and payment\_status, to enhance both search and update performance. Specifically, indexes will be applied to the 'students' table for attributes like 'name' and 'email', which are commonly queried. Additionally, indexes will be placed on key columns of the 'billing\_and\_payment' table to further optimize query performance and facilitate efficient data retrieval.
  
- **Triggers:**

- Database triggers will be implemented to automatically update related fields, such as total\_paid in financial summaries, whenever a payment record is modified.
- This automation ensures consistency across the database and reduces the need for recalculating summaries during queries, thereby improving overall system efficiency.

### 3. Retrieve Tutor Evaluation Record

Description: Fetching tutor evaluation data, including scores and feedback, for performance analysis and review purposes.

Performance Sensitivity: Queries related to retrieving tutors' evaluations would have high performance effects as it would involve large data sets, having many joins (e.g with evaluation\_date and session\_id) and aggregations. High demand for related records can also slow down response times.

#### Optimization Strategies:

1. **Views:** Views would be created (and stored) for this record on the database for faster retrieval, considering its a frequently accessed record.
2. **Indexing:** Add composite indexes on (tutor\_id and evaluation\_date) to enhance filtering and sorting performance.
3. **Query Optimization:** Use EXPLAIN to analyze query plans and optimize join orders, ensuring efficient execution.

## PROFESSIONAL, LEGAL AND ETHICAL ISSUES

### PROFESSIONAL, LEGAL, AND ETHICAL ISSUES

When developing a database system, it is crucial to address professional, legal, and ethical considerations to ensure its functionality, compliance, and fairness in real-world applications. Below, we outline key aspects of these considerations.

#### Professional Issues

Professional issues focus on technical and operational standards to create a functional, reliable database system.

- **Data Integrity:** As Adrienne Watt emphasizes, identifying entities is a foundational step in database development. This includes analyzing data requirements based on client needs and industry insights, creating entity-relationship diagrams to define the scope, and enforcing attributes with appropriate data types and constraints.
- **Scalability and Performance:** To handle growing data efficiently, we discussed indexing frequently queried data, while avoiding over-indexing, which could degrade performance during updates. Transaction analysis would play a critical role in determining these key relationships for indexing, especially for large data sets. We also plan to use Explain-Analyse for query optimization and runtime statistics. For large databases, we considered database sharding, as suggested by Intersystems Creative Data Technology, to split the database into smaller, manageable parts (shards) for improved performance and scalability.
- **Maintainability:** We resolved to include well-commented SQL queries and procedures to ensure ease of future updates by other professionals.

#### Legal Issues

Considering **legal issues**, our database system implementation in a real-world setting would involve complying with the **General Data Protection Regulation or Local data protection laws**. To handle sensitive information such as addresses and passport numbers, we would implement encryption and hashing techniques and consent proof from users will be required for storing their information.

**Access Control:** Misuse of access can lead to compliance breaches, as noted by Robert Haeley. We figured to use GRANT and REVOKE commands to limit data modifications to authorized users.

**Data Retention and Deletion:** Organizations should retain data only for specific purposes. For critical records, soft deletion can be implemented, marking data as deleted without permanently removing it. This approach, inspired by Prisma middleware, ensures compliance with retention policies and allows potential recovery while maintaining historical data integrity. A boolean and timestamp can be used to manage and track these data deletions.

#### Ethical Issues

Ethical concerns revolve around transparency, consent, and inclusivity.

**Transparency and Consent:** Users must be fully informed about how their data will be used, and consent must be obtained.

**Inclusivity:** According to Shish Kumar Dubey et al., it is essential to avoid bias by ensuring that diverse data details are represented. For example, excluding operational staff salaries while recording academic staff salaries could lead to unintentional discrimination. Proper entity analysis before development helps mitigate such risks.

Addressing these professional, legal, and ethical considerations ensures a robust, compliant, and fair database system, aligning with client and user expectations while maintaining high standards.

## REFERENCES

*NOTE: These references are for both the optimization session (references A) and the professional, legal, and ethical issues (references B).*

### References A

Colley, D. A. (2021). *Development of a dynamic design framework for relational database performance optimisation* (By M. Asaduzzaman, E. Wilson, & Staffordshire University) [Thesis].  
[https://eprints.staffs.ac.uk/7402/1/Thesis\\_Final\\_with\\_Corrections\\_Submitted.pdf](https://eprints.staffs.ac.uk/7402/1/Thesis_Final_with_Corrections_Submitted.pdf)

CIOLOCA, C., & GEORGESCU, M. (2011). *Increasing Database Performance using Indexes*. In *Academy of Economic Studies, Database Systems Journal: Vol. II* (Issue 2, pp. 13–15) [Journal-article]. Academy of Economic Studies.  
[http://dbjournal.ro/archive/4/2\\_Ciologa\\_Georgescu.pdf](http://dbjournal.ro/archive/4/2_Ciologa_Georgescu.pdf)

*Practical Applications of Triggers and Constraints: Successes and Lingering Issues*. (2000). [Journal-article].

### References B

Watt, A. (2014, October 24). *Chapter 13 Database Development Process*. Pressbooks.  
<https://opentextbc.ca/dbdesign01/chapter/chapter-13-database-development-process/>

Open Education. (2024, September 9). Hewlett Foundation.  
<https://hewlett.org/strategy/open-educational-resources/>

Mancas, C., & Dorobantu, V. (2017). *On enforcing relational constraints in MatBase*. London Journal of Research in Computer Science and Technology, 17(1), 1–39.  
[https://journalspress.com/LJRCST\\_Volume17/16\\_On-enforcing-relational-constraints-in-MatBase.pdf](https://journalspress.com/LJRCST_Volume17/16_On-enforcing-relational-constraints-in-MatBase.pdf)

Mullins, C. S. (2023, December 18). *Scalability and Performance: Different but Crucial Database Management Capabilities*. Database Trends and Applications.  
<https://www.dbta.com/Editorial/Think-About-It/Scalability-and-Performance-Different-but-Crucial-Database-Management-Capabilities-161866.aspx>

*Mastering Database Sharding: Strategies and Best Practices*. (2024, September 26). United Kingdom.  
<https://www.intersystems.com/uk/resources/mastering-database-sharding-strategies-and-best-practices/>

Desktop, N. (2024, June 4). *Understanding SQL Comments: How to Document Your Code*. Noble Desktop.  
<https://www.nobledesktop.com/learn/sql-server/understanding-sql-comments-how-to-document-your-code>

*Middleware sample: soft delete (Reference)*. (n.d.). Prisma Documentation.  
<https://www.prisma.io/docs/orm/prisma-client/client-extensions/middleware/soft-delete-middleware>

ICO. (n.d.). *Retention and destruction of information.*

[https://ico.org.uk/for-organisations/foi/freedom-of-information-and-environmental-information-regulations/\\_retention-and-destruction-of-information/](https://ico.org.uk/for-organisations/foi/freedom-of-information-and-environmental-information-regulations/_retention-and-destruction-of-information/)

Flyriver: Soft Deletion: A Comprehensive Overview. (n.d.). <https://www.flyriver.com/g/soft-deletion>

Centraleyes. (2024, May 29). *What is Access Control Policy ?*

<https://www.centraleyes.com/glossary/access-control-policy/>

Healey, R. (2024, November 11). *GDPR Access Controls: Choosing the Right Model for Your Organisation.*

Formiti. <https://formiti.com/gdpr-access-controls-which-one-is-right-for-you/>

ICO. (n.d.-b). *What is the right of access?*

<https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/individual-rights/right-of-access/what-is-the-right-of-access/>

Dubey, S.K., Gupta, S., Pandey, S., & Mittal, S. (2024). *Ethical Considerations in Data Mining and Database Research.* 2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 1-4.

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10522200>

# Database Systems Development

## M21269

Group Number: [24]

### Database Development Tracker

Date	Task Description	Member ID	Task Details	Time	Signature
11/10/2024	<b>Initial meeting</b>	UP2158902 UP2209423 UP2162181	We initially met to discuss the coursework details and plan future meetings. Most of our meetings were held in person, often in the library, although we occasionally met at the Richmond Building. Each member also agreed to work out of agreed meeting times, which we did, leading to the submission of the coursework.	1H	Augustine Oliver Abdelrhman
14/10/2024	<b>Case Study</b>	UP2158902 UP2209423 UP2162181	We identified obvious entities (direct and inferred), placing their interactions with each other in an initial entity diagram. We analyzed the case study to identify more <b>entities</b> and their relationships, then discussed the elements of our <b>ERD</b> with <b>tables</b> , <b>attributes</b> , and <b>keys</b> . An initial framework of the Entity Relational diagram was designed with a brief explanation of the key <b>assumptions</b> which we documented.	2H	Augustine Oliver Abdelrhman
17/10/2024	<b>ERD Design</b>	UP2158902 UP2209423 UP2162181	We started creating the <b>ERD</b> based on what we analyzed from the case study and our findings from all our analysis.	2H	Augustine Oliver Abdelrhman
18/10/2024	<b>ERD Design</b>	UP2158902 UP2209423 UP2162181	Still working on the <b>ERD</b> design.	2H	Augustine Oliver Abdelrhman
21/10/2024	<b>Assumptions</b>	UP2158902 UP2209423 UP2162181	We documented our assumptions, although this was a continuous process in the build-up of the coursework.	3H	Augustine Oliver Abdelrhman

24/10/2024	<b>Data Dictionary</b>	UP2158902 UP2209423 UP2162181	After completing the <b>ERD</b> , we created the <b>Data Dictionary</b> , documenting each entity's attributes, data types, constraints, and relationships. This ensures consistency and clarity in the database design and implementation.	2H	Augustine Oliver Abdelrhman
25/10/2024	<b>SQL Code</b>	UP2158902 UP2209423 UP2162181	We created the SQL code, organizing it by parent and child tables, and then tested it to ensure everything functions correctly.	2H	Augustine Oliver Abdelrhman
28/10/2024	<b>SQL Code</b>	UP2158902 UP2209423 UP2162181	Still working on creating the SQL code and made sure to continuously test the codes from the previous session to avoid inconsistencies.	2H	Augustine Oliver Abdelrhman
31/10/2024	<b>INSERTS</b>	UP2158902 UP2209423 UP2162181	After creating the Code, we started creating the <b>INSERT</b> statements to populate the tables, ensuring data consistency and referential integrity.	2H	Augustine Oliver Abdelrhman
1/11/2024	<b>INSERTS</b>	UP2158902 UP2209423 UP2162181	Still working on the <b>INSERT</b> , as we needed to add more values to each attribute.	2H	Augustine Oliver Abdelrhman
07/11/2024	<b>Testing</b>	UP2158902 UP2209423 UP2162181	After completing the code and <b>INSERT</b> statements, we began testing to evaluate the performance and identify any errors. This allowed us to fix issues and ensure everything was functioning as expected.	2H	Augustine Oliver Abdelrhman
21/11/2024	<b>Create Queries</b>	UP2158902 UP2209423 UP2162181	To evaluate the database's usability, we created five queries that we believe will be useful for the business. We used <b>VIEWS</b> where necessary and based other queries on them. Each query was tested and the results captured as screenshots, which are included in the main report along with a description of their purpose. A copy of all queries is also provided at the end of the SQL file. We created a transaction analysis matrix.	2H	Augustine Oliver Abdelrhman
30/11/2024	<b>Security</b>	UP2158902 UP2209423 UP2162181	Each team member resumed their tasks from the previous week, focusing on the security aspects of the database. We looked at designing and implementing two roles, <b>ses_manager</b> and <b>ses_tutor</b> , ensuring each role was assigned appropriate privileges based on their responsibilities. Additionally, we tested access controls to verify that	1H	Augustine Oliver Abdelrhman

			privileges were correctly configured, preventing unauthorized access to sensitive data..		
07/12/2024	<b>Optimisation</b>	UP2158902 UP2209423 UP2162181	We discussed three most performance-sensitive transactions related to the business setting, also using the transaction analysis matrix and started with discussion and research pertaining to the optimisation; also recording our findings in our report.	2H	Augustine Oliver Abdelrhman
11/12/2024	<b>Optimisation</b>	UP2158902 UP2209423 UP2162181	Still working on the previous aspect of our work and concluded on it.	2H	Augustine Oliver Abdelrhman
14/12/2024	<b>Professional,legal and Ethical issues</b>	UP2158902 UP2209423 UP2162181	We discussed professional, ethical and legal issues that could influence our database if we were working for a real client, we researched on these and documented our findings on our report. We also noted the references as well.	3H	Augustine Oliver Abdelrhman
20/12/2024	<b>Submission</b>	UP2158902 UP2209423 UP2162181	We thoroughly rechecked everything to ensure everything was functioning properly and met all requirements before confidently submitting it.	1H	Augustine Oliver Abdelrhman