A black background with white text

Description automatically generatedA red and black logo

Description automatically generated​

*Information Technology Institute*

*Power BI Developer Track*

**Examination System**

**Presented by:**

Reham Essam

Ahmed Marzouk

Ali Omar

Basma Bassem

Mohammed Ramadan

**Supervised by:**

Eng. Sarah Salah

**ACKNOWLEDGEMENT**

The success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

The special thanks go to our helpful supervisor **Eng. Sarah Salah.**

The supervision and support she gave truly helped in the progression of our graduation project. We also take this opportunity to express our gratitude to the Information Technology Institute (ITI) for their Support and efforts in providing us with all useful resources. We express a special thanks to the Information Technology Institute (ITI) stuff for the valuable information provided by them in their respective fields.

Lastly, we thank God, Our Parents and all the people who helped, supported and encouraged us to successfully finish the graduation project

**ABSTRACT**

This project details the development of an automated Examination System tailored for the Information Technology Institute. The system is engineered to streamline the management of exams, questions, students, instructors, and courses, providing a robust solution for handling various examination-related processes. Key features include automated exam generation, student answer recording, result calculation, and dynamic reporting. The database is constructed with a comprehensive schema that supports these functionalities and integrates seamlessly with SQL Server Reporting Services (SSRS) and Power BI for insightful data visualization. These tools enable the creation of detailed reports and interactive dashboards that provide valuable insights into academic performance, instructor effectiveness, and overall departmental operations, thereby facilitating informed decision-making. The project emphasizes a systematic approach to database creation, data extraction, and report generation, enhancing the efficiency and quality of the examination management process.

**INTRODUCTION**

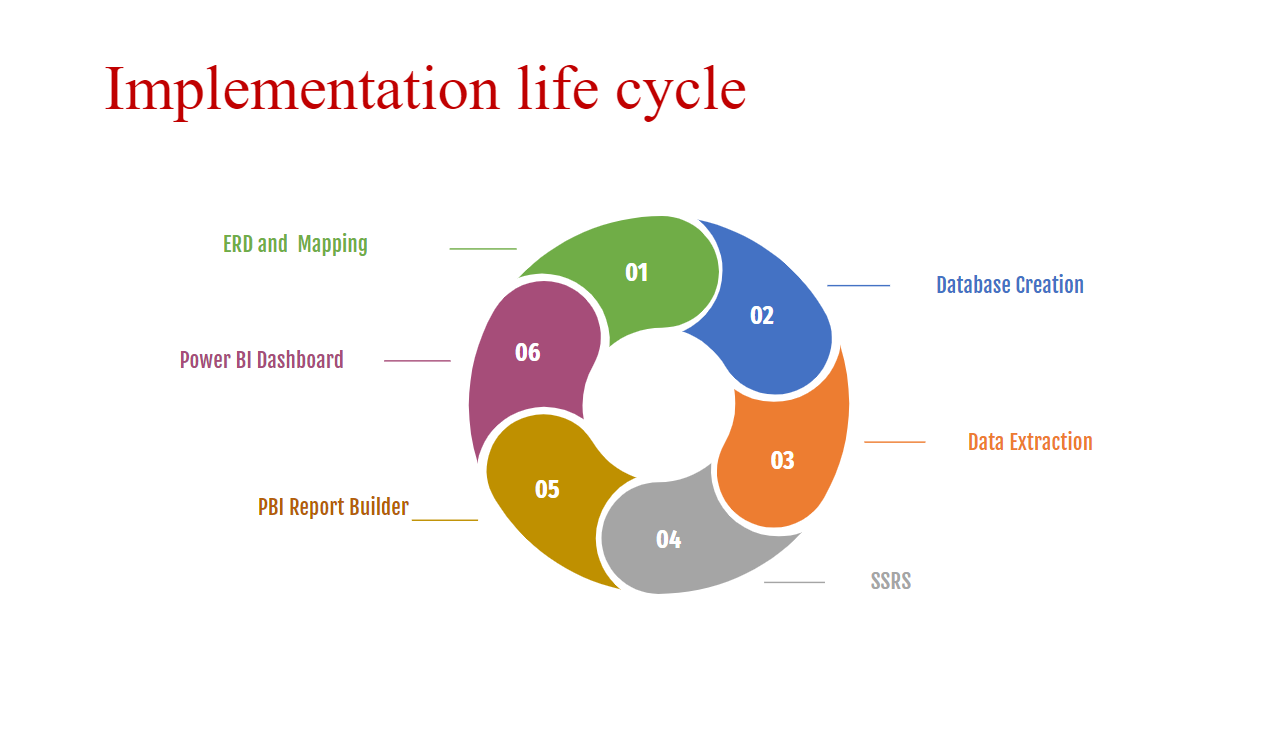
The Examination System Database is developed to streamline the management of examination processes within an educational institution. It addresses the need for an efficient, automated system that handles the entire lifecycle of exams, from creation and management to grading and reporting. The project aims to improve the examination experience for instructors, students, and administrators by providing a structured approach to data management and reporting.

The implementation of the system follows a detailed life cycle, starting from the design of an Entity-Relationship Diagram (ERD) to the development of the database schema and subsequent integration of various system components. The database is designed to support complex functionalities such as random question assignment, automated grading, and student performance tracking. It utilizes Microsoft SQL Server for database management and integrates with Power BI for advanced data visualization and reporting.

This documentation outlines the comprehensive steps taken during the project's development, including database design, stored procedure creation, data extraction processes, and report generation. It also highlights the system's ability to provide interactive dashboards and reports that facilitate the evaluation of academic performance, thereby supporting the institution's goal of enhancing educational quality and operational efficiency.

**IMPLEMENTATION LIFE CYCLE**

The implementation life cycle is an iterative process needed to ensure the system meets the requirements that outline the phases a system goes through, from entity relationship diagrams to reports and interactive dashboards.



**ERD& Mapping:** In this phase, we created detailed design specifications for the entity relationship diagram, and database schema, and developed dashboard perspectives.

**Database creation:** Here, we coded the database schema, integrated the components into a working system, and tested the system to ensure it can manage exams and answers.

**Data extraction:** extracting data from a database for generating reports in Power BI Desktop, SQL Server Report Service, or Power BI Report Builder involves querying the database, saving the data as a CSV file, and importing the data into the reporting tool, or directly connecting to the database from the reporting tool and extracting the data.

**SSRS:** SQL Serverreport server to present the data from stored procedures to SQL Server Report Server to show the targeted report like different courses, topics, and exams.

**Power BI Report builder:** paginated report designed for creating a report definition that specifies what data to retrieve like Report that returns the students’ information according to Department No parameter, and Report that takes the exam number and the student ID and then returns the Questions in this exam with the student answers.

**Power BI Dashboard:** to provide valuable insights into the department, instructor, student performance, and exam results, helping instructors make informed decisions about how to improve the intake quality and support more students learning through online courses.

# **DATABASE DESIGN (ERD)**

# Examination system ERD includes entities such as Intake, Department, Track, Student, Question, Instructor, Exam, Course, Topic, and their relationships to each other.

# 

# DATABASE DESIGN (MAPPING)

# Mapping in the examination system refers to the process of creating a correspondence or association between different entities or attributes in the system needed to implement in the SQL Server Management Studio.

# 

# IMPLEMENTATION

# **1. Creation Database**

# After Mapping and ERD, Database is created including 2 files groups PRIMARY file group, and Secondary file group.

# 

# 

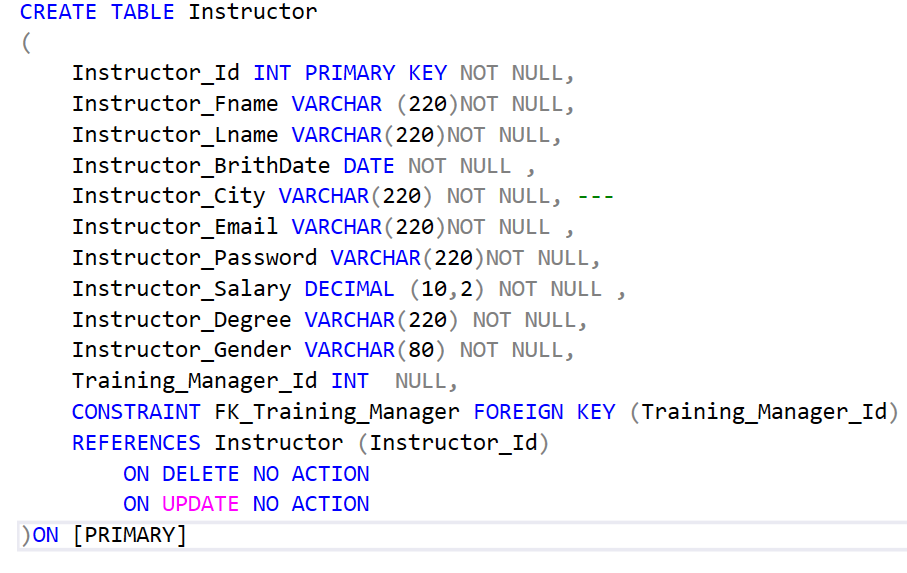
### **2. Tables Creation**

# Department, Instructor, Exam, Question, Intake, Track, Student, Course, Topic, Student\_Course, Course\_Track,

# **2.1. Department Table**

# 

# **2.2. Instructor Table**



# **2.3. Exam Table**

# 

# **2.4. Question Table**

# 

# **2.5. Intake table**

# 

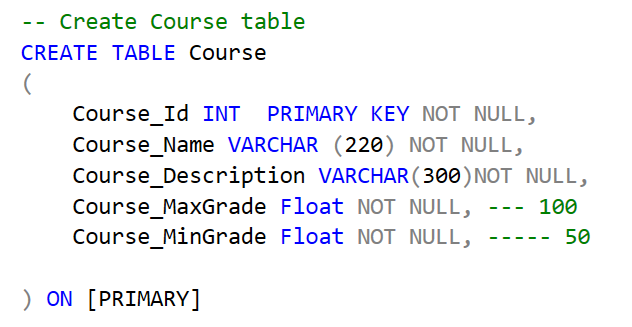
# **2.6. Track Table**

# 

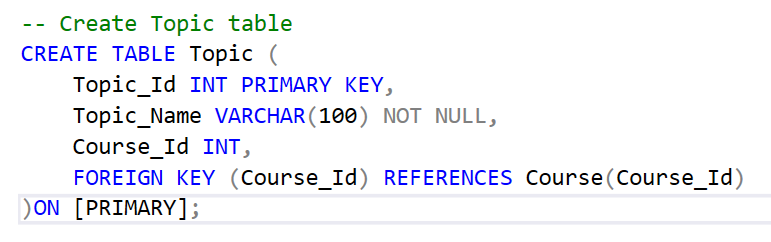
# **2.7. Student Table**

# 

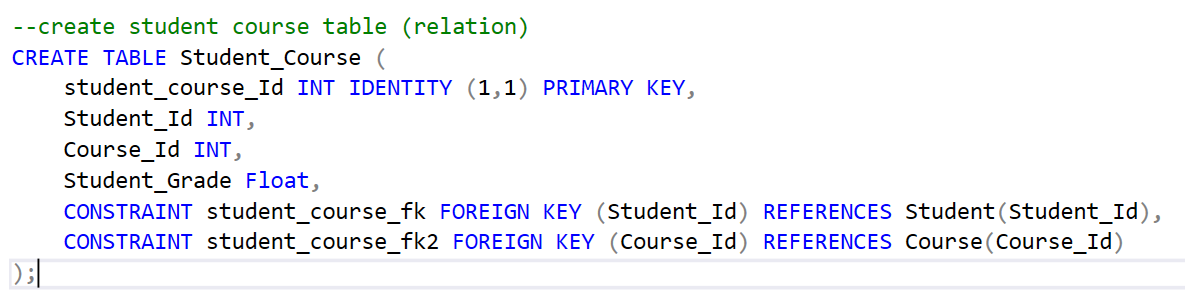
# **2.8. Course Table**



**2.9. Topic Table**



**2.10. Student\_Course Table**

****

# **2.11. Course\_Track Table**

# 

# **2.12. Exam\_Question Table**

# 

# **2.13. Instructor\_Course Table**

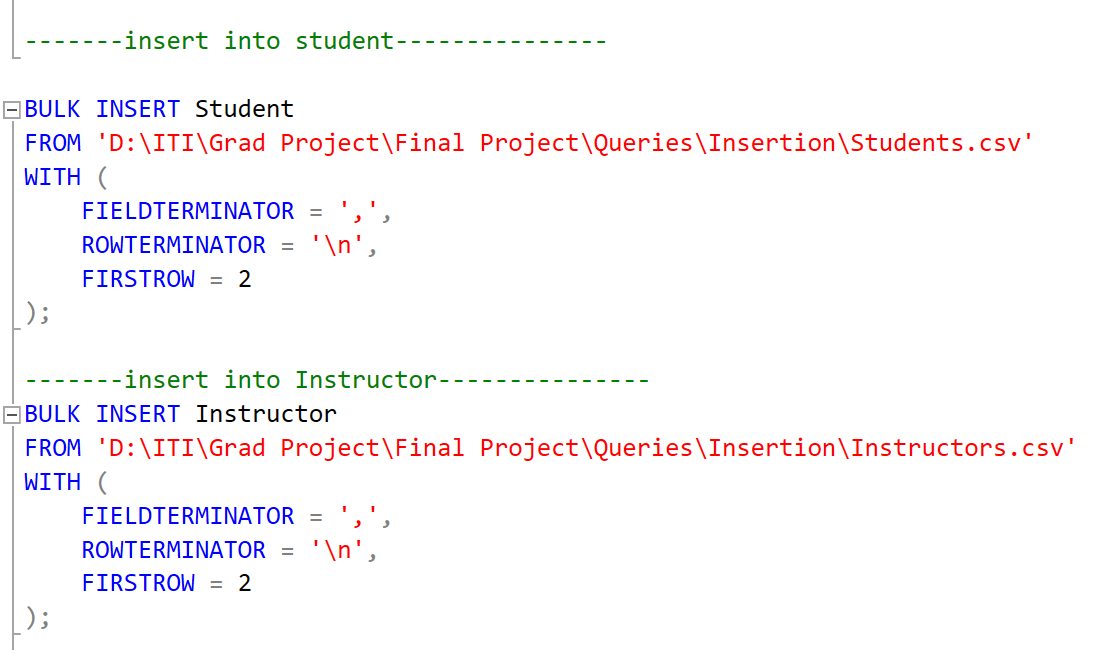
# 

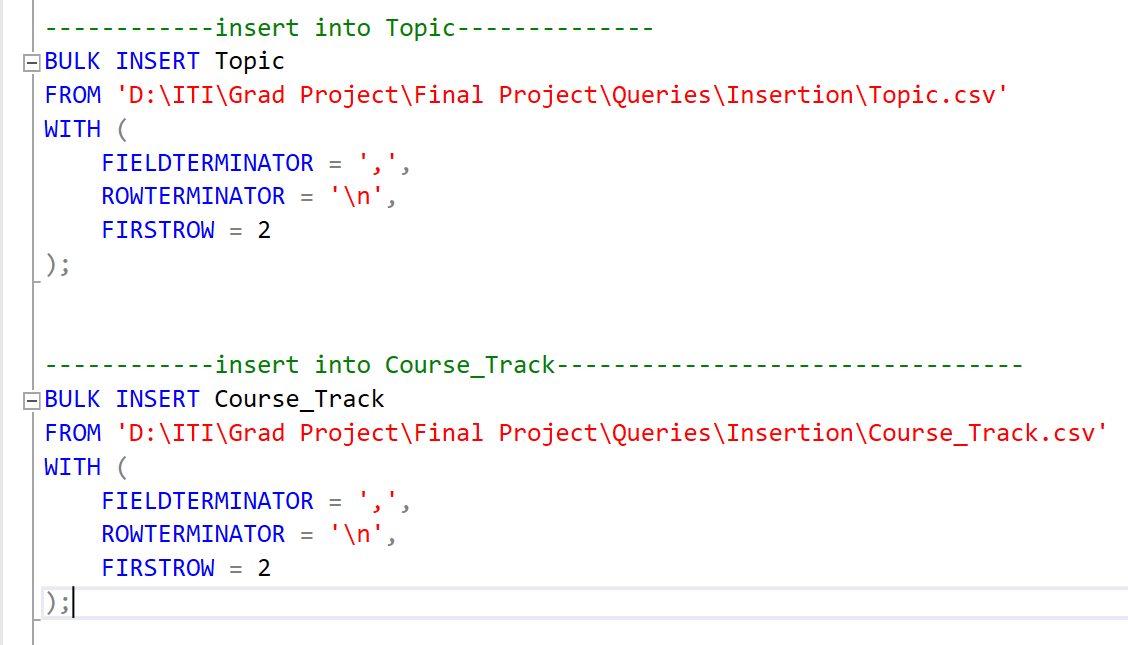
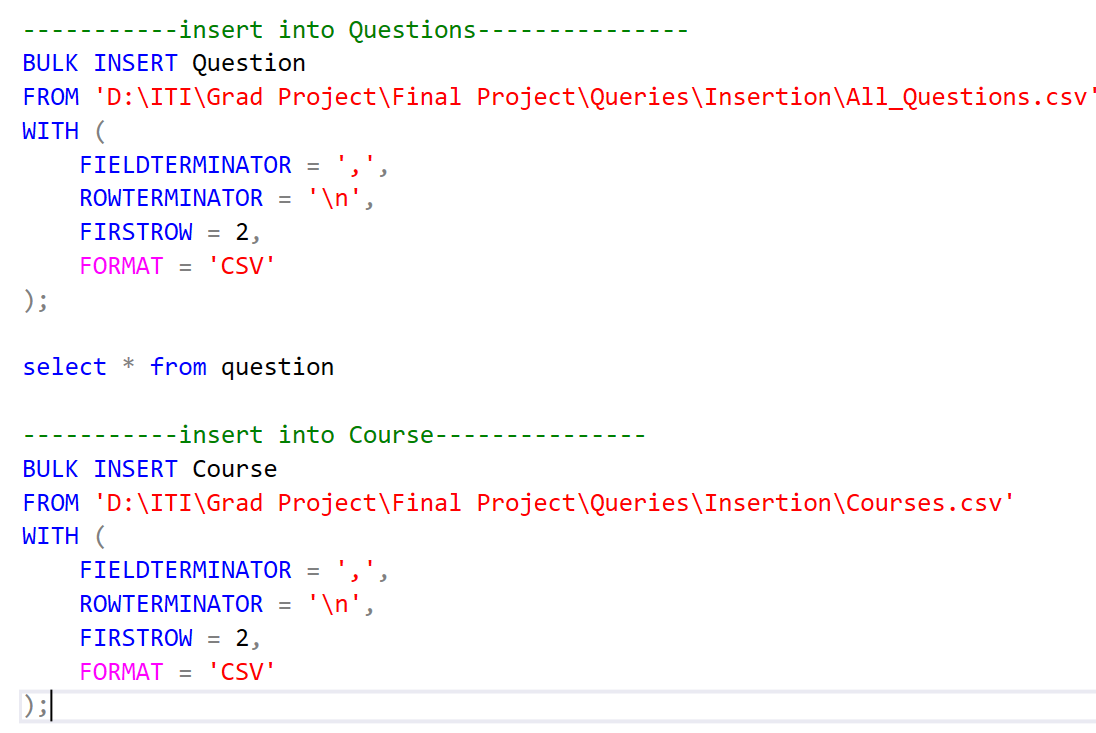
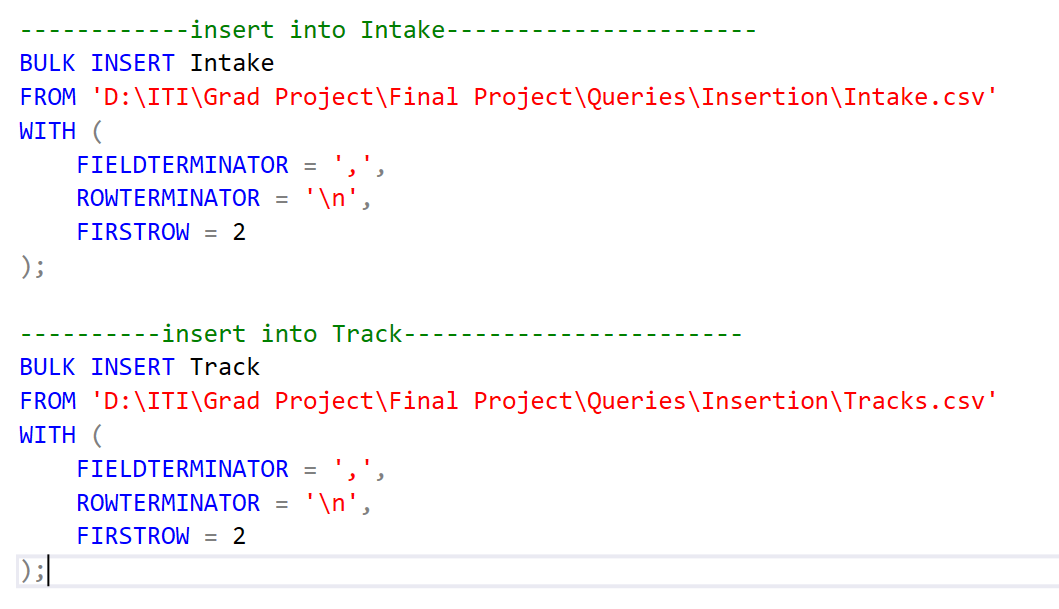
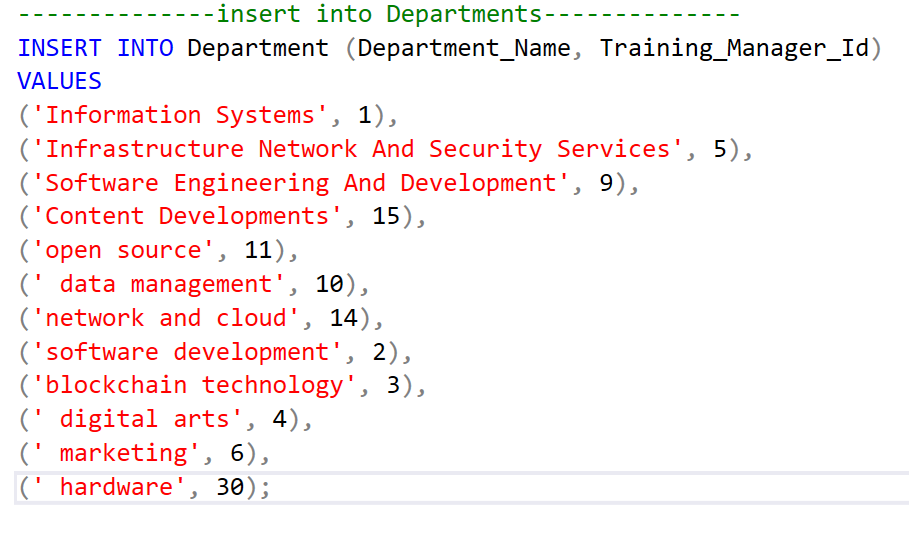
# **2.14. Answer Table**

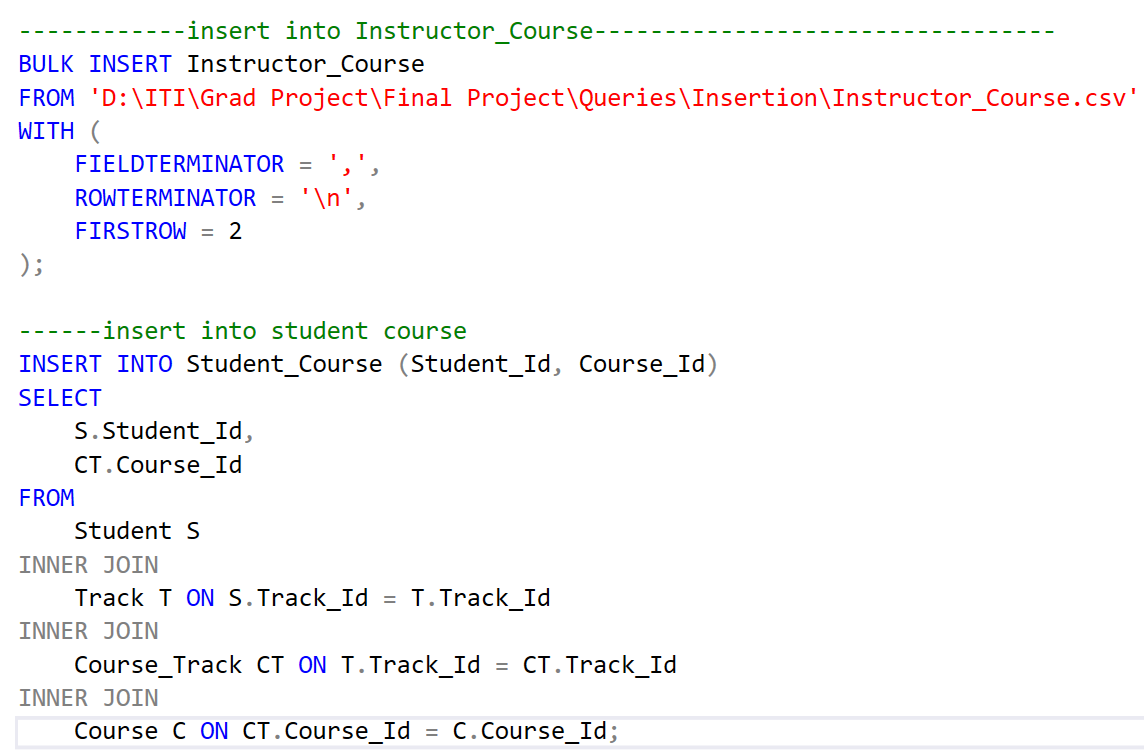
# 

# **3. Insertion**

Once we defined the tables, we populated the tables with data. We used the Bulk Insert command. This command allows for the efficient insertion of large amounts of data into tables from a data file. The data file can be CSV in our system.



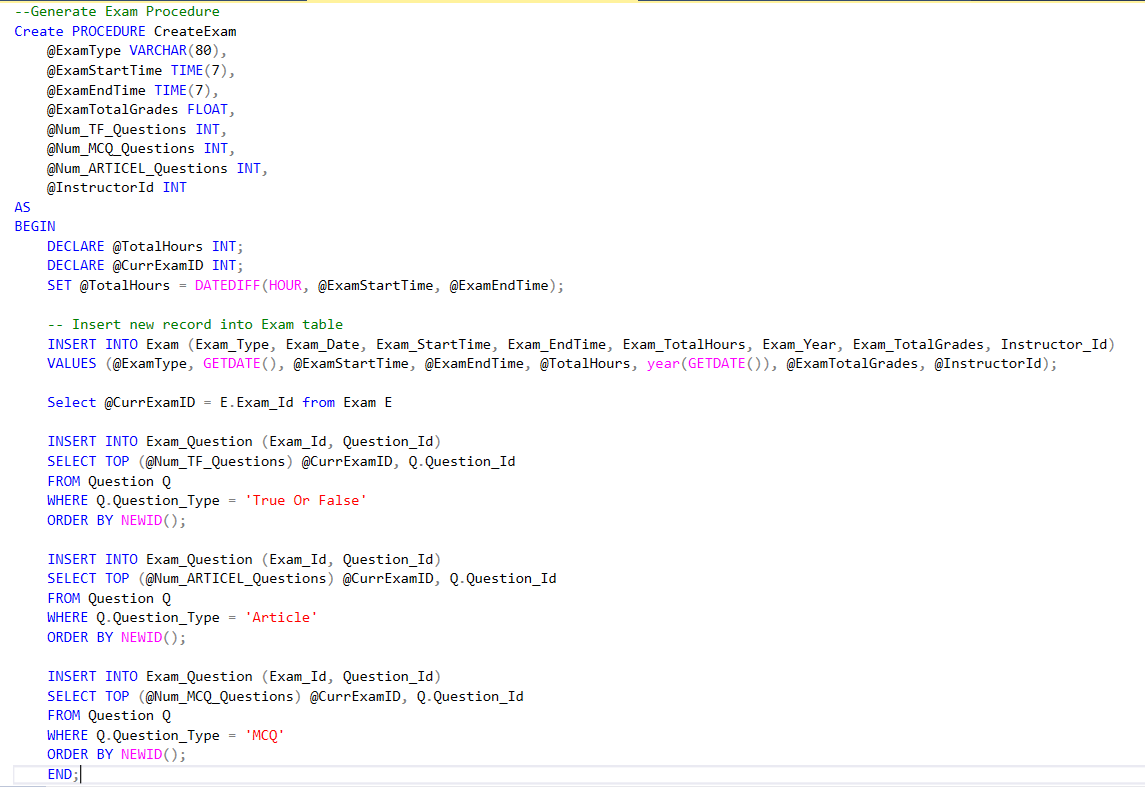




# **4. Procedures**

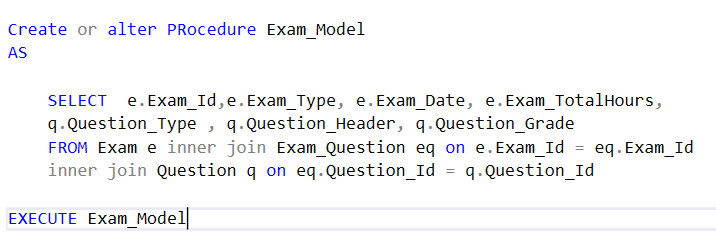
# **4.1. Main Procedure:** Generate Exam

# This procedures is used by Instructor, generates a new exam by inserting a record into the Exam table with specified details such as exam type, start and end times, total grades, and the instructor's ID. It then randomly assigns a specified number of True/False, MCQ, and Article questions from the Question table to the newly created exam.



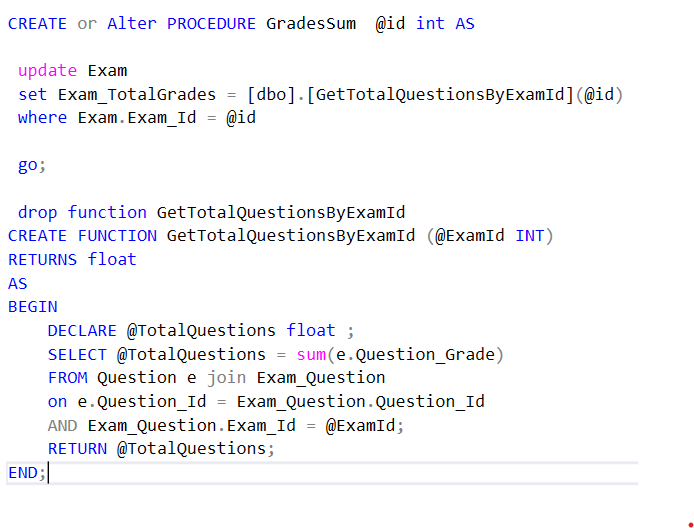
**4.2.** **Main Procedure:** Exam\_Model

This procedure retrieves a detailed list of exams along with associated questions, showing information such as exam type, date, total hours, and question details like type, header, and grade.



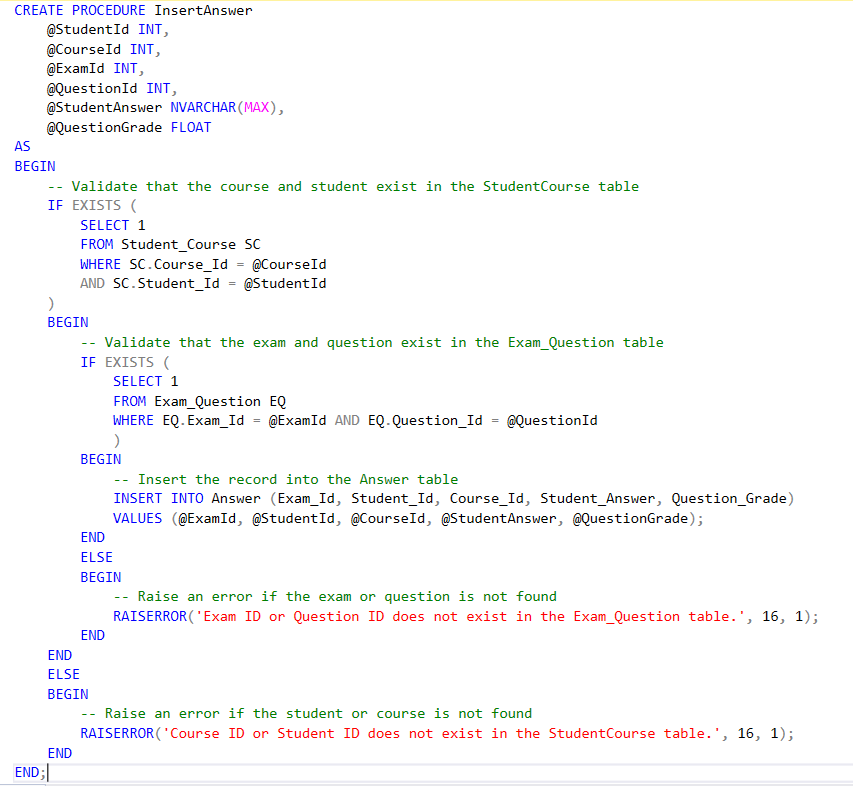
**4.3. Main Procedure:** GradesSum

This procedure calculates the total grades for a specific exam by summing the grades of all its associated questions and updates the Exam table with the calculated total grades using the GetTotalQuestionsByExamId function.



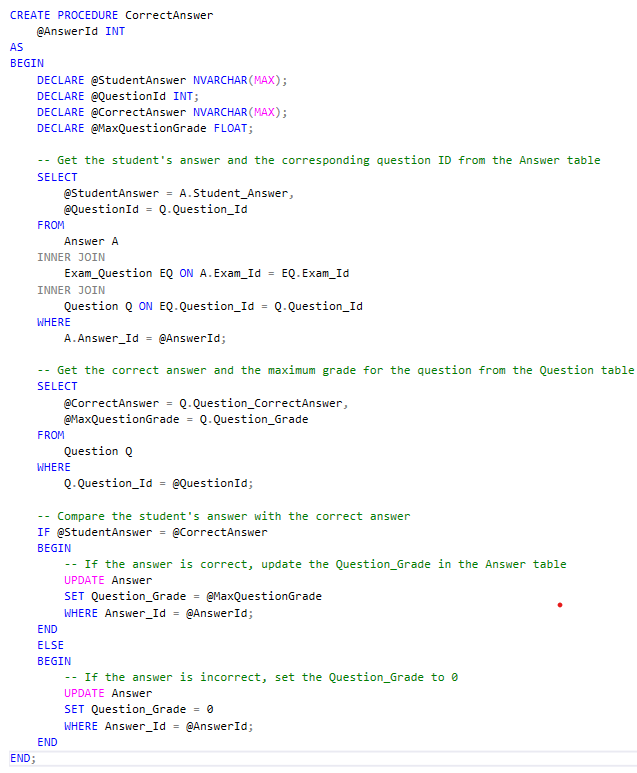
**4.4. Main Procedure:** InsertAnswer

This procedure records a student's answer to a question in an exam. It first validates that the student is enrolled in the course and that the question exists in the exam. If valid, it inserts the answer into the Answer table along with the student ID, course ID, and the assigned grade.



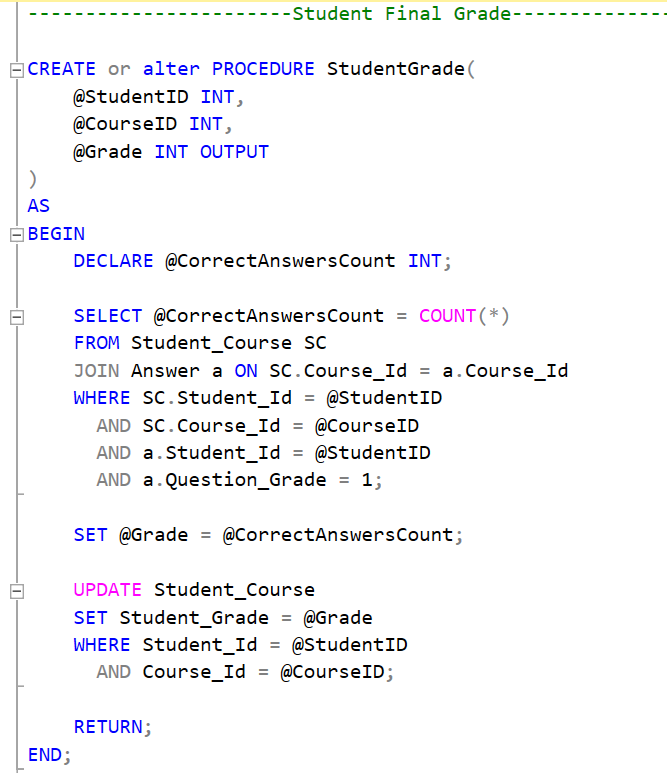
**4.5. Main Procedure:** CorrectAnswer

This procedure corrects a student's answer by comparing it with the correct answer from the Question table. If the student's answer matches, it assigns the full grade to the answer. If not, the grade is set to zero.



**4.6.** **Main Procedure:** StudentGrade

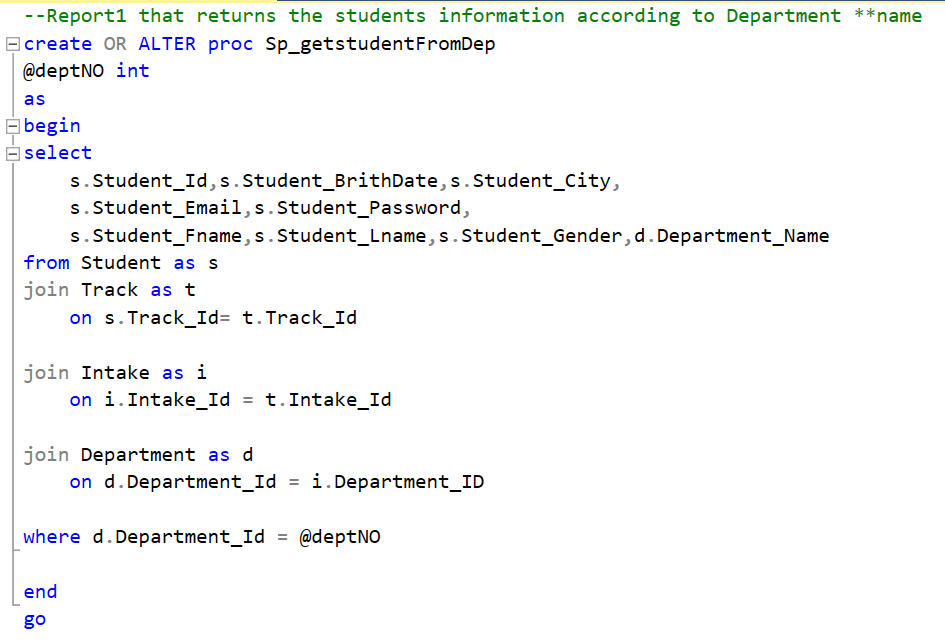
This procedure calculates the final grade for a student in a course by counting the correct answers from the Answer table. It updates the Student\_Course table with the student's grade and returns the grade value.



**4.7. Report Procedure:** Sp\_getstudentFromDep

Report for return the students’ information according to Department Number.

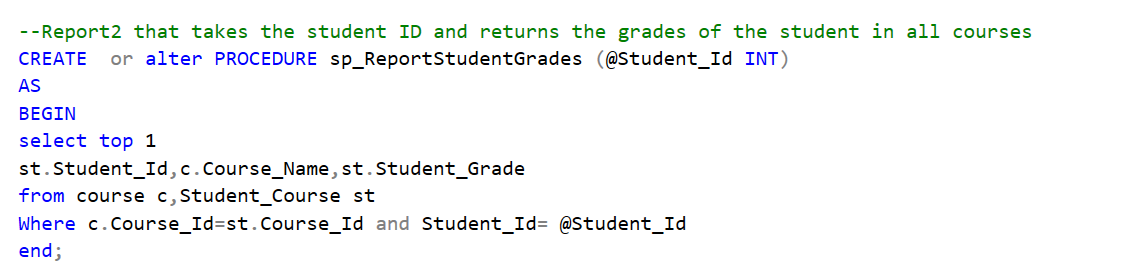
This procedure retrieves student information, including details such as student ID, birthdate, city, email, and department name, based on a given department ID. It joins the Student, Track, Intake, and Department tables to filter students belonging to the specified department.



**4.8. Report Procedure:** sp\_ReportStudentGrades

Report that takes the student ID and returns the grades of the student in all courses.

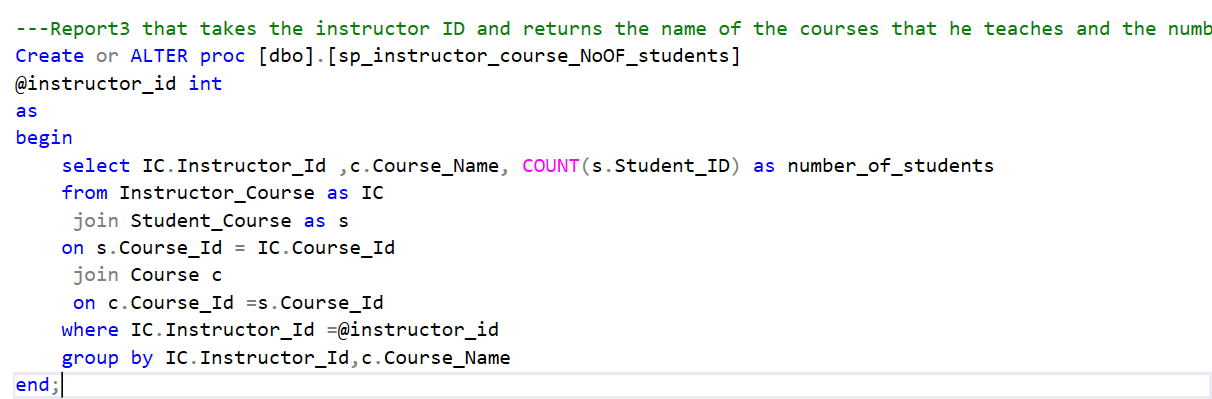
This procedure takes a student ID as input and returns the student's grades for all enrolled courses. It joins the Course and Student\_Course tables to display course names and corresponding grades for the specified student.



**4.9. Report Procedure:** sp\_instructor\_course\_NoOF\_students

Report that takes the instructor ID and returns the name of the courses that he teaches and the number of student per course.

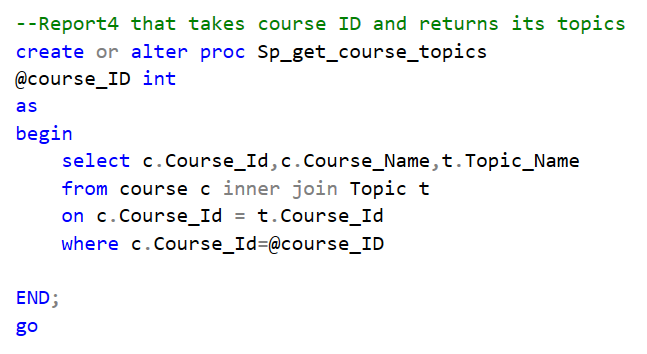
This procedure retrieves the courses taught by a specific instructor, along with the number of students enrolled in each course. It takes an instructor ID as input and joins the Instructor\_Course, Student\_Course, and Course tables to calculate the student count per course taught by the instructor.



**4.10. Report Procedure:** Sp\_get\_course\_topics

Report that takes course ID and returns its topics.

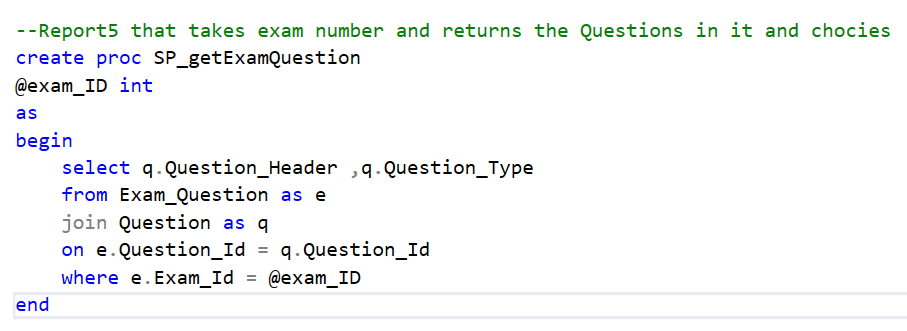
This procedure returns the topics covered in a specific course based on the course ID provided. It joins the Course and Topic tables to list all topics associated with the given course.



**4.11. Report Procedure:** SP\_getExamQuestion

Report that takes exam number and returns the Questions in it and choices.

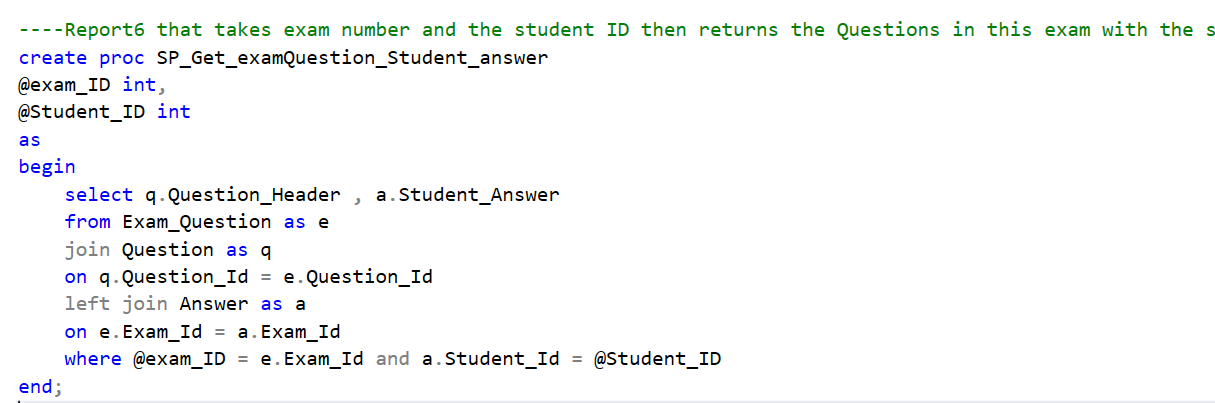
This procedure retrieves the questions associated with a specific exam, including the question header and type. It takes an exam ID as input and joins the Exam\_Question and Question tables to display the questions for the given exam.



**4.12. Report Procedure:** SP\_Get\_examQuestion\_Student\_answer

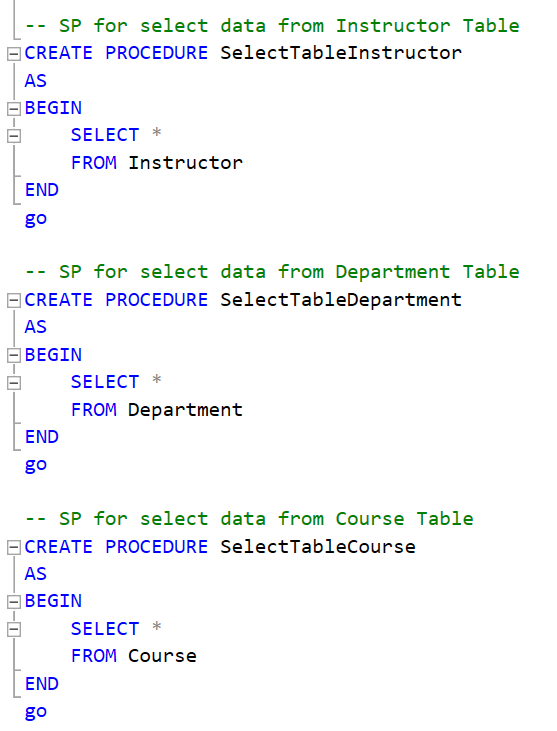
Report that takes exam number and the student ID then returns the Questions in this exam with the student answers.

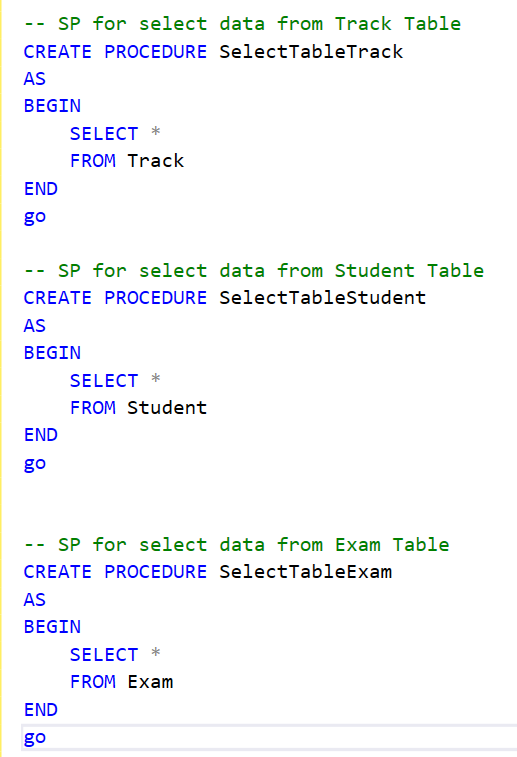
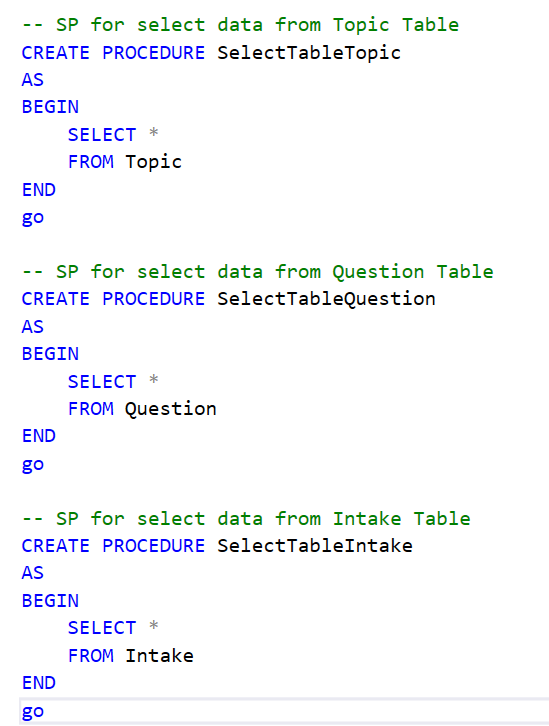
This procedure returns the questions from a specific exam along with the corresponding answers provided by a specific student. It takes both exam ID and student ID as input and joins the Exam\_Question, Question, and Answer tables to display the student's answers for each question in the exam.

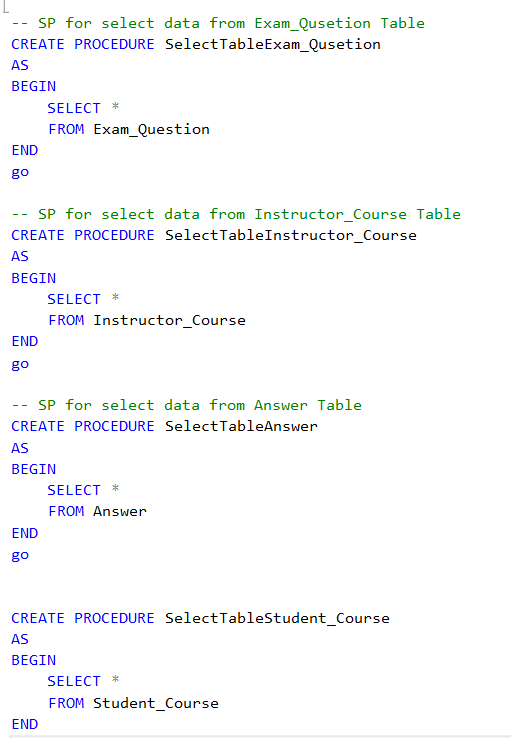


**4.13. Select Procedures:** For All Tables

Retrieves all data from the specified table, returning the complete set of records. It provides a basic mechanism for selecting and viewing all available data within the respective table in the database.

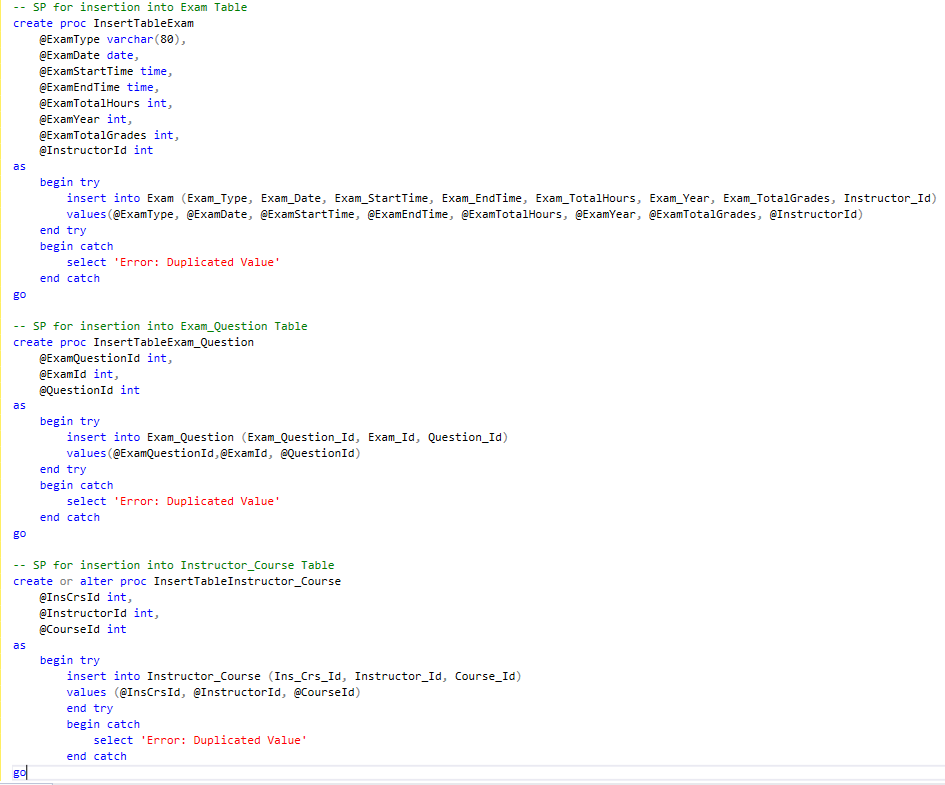
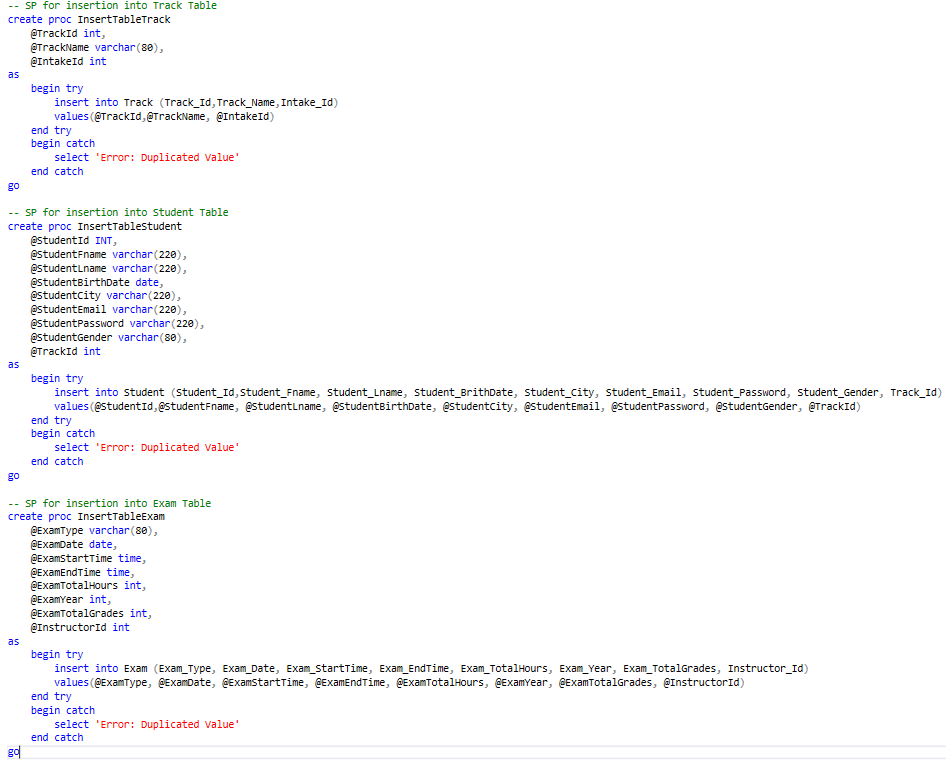
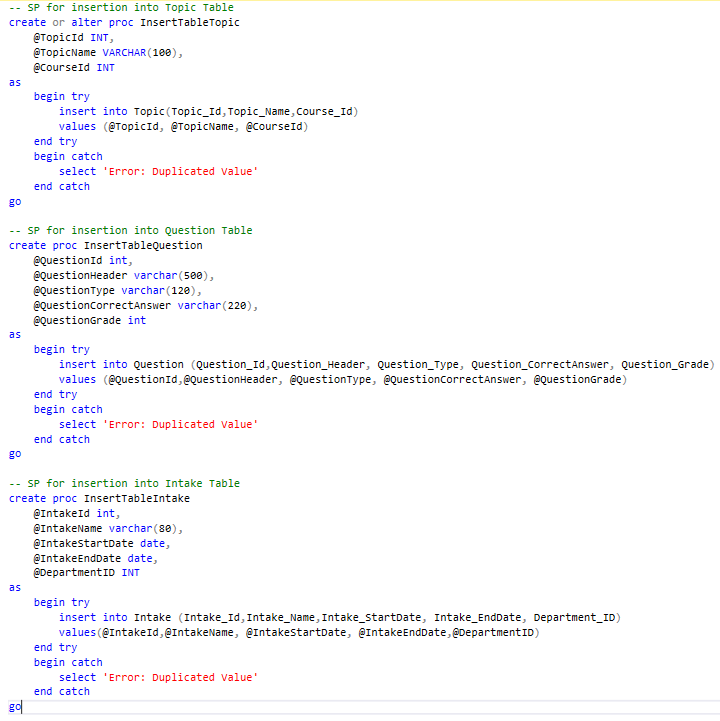


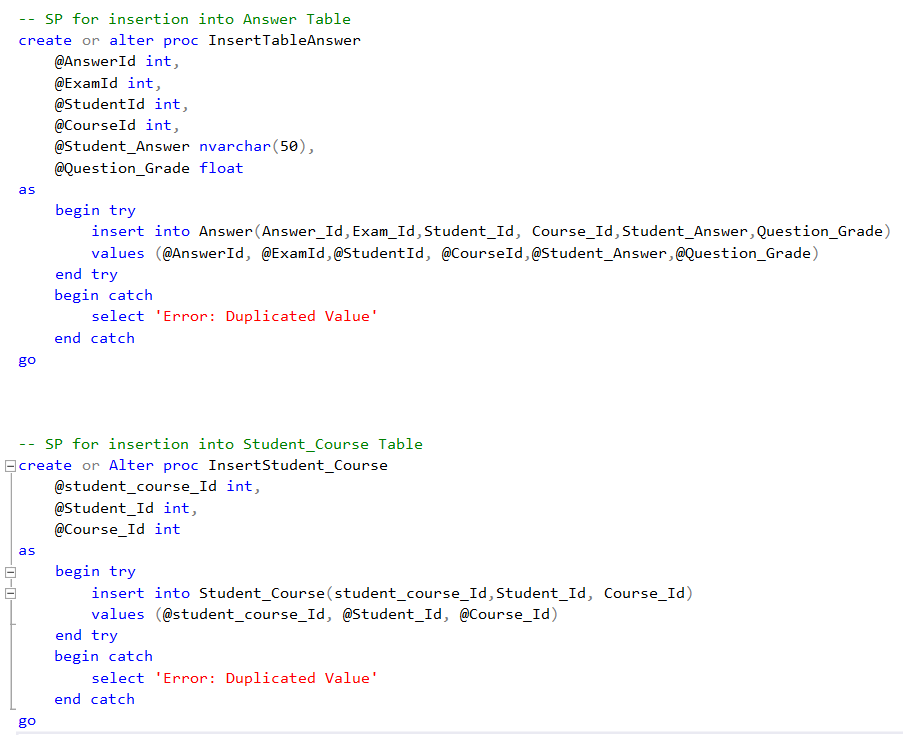




**4.14. Insert Procedures:** For All Tables

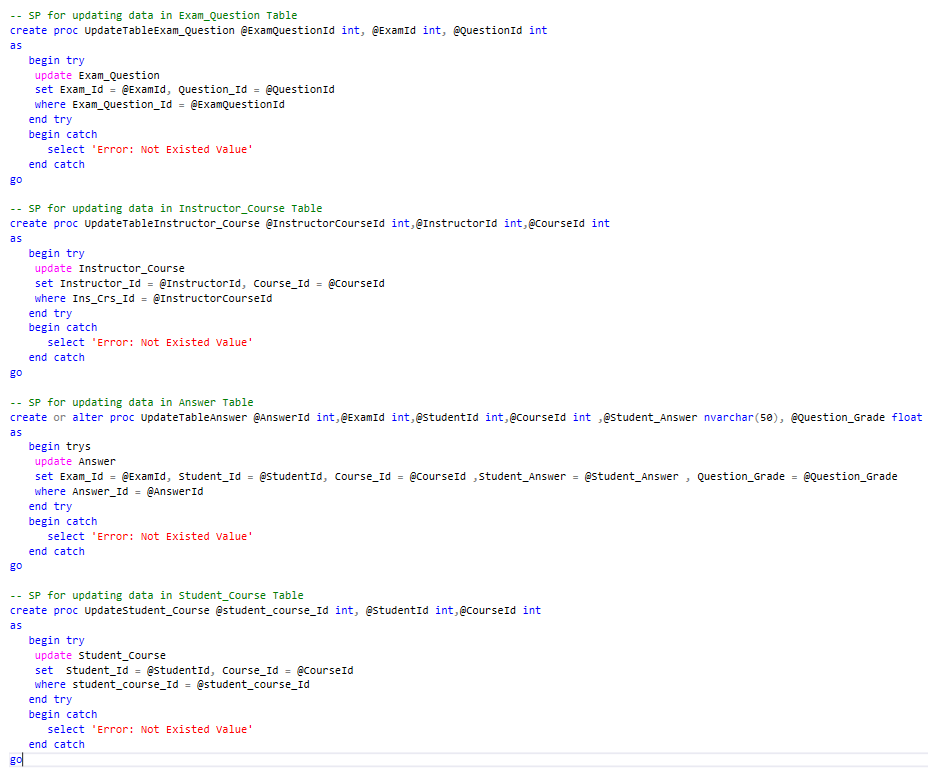
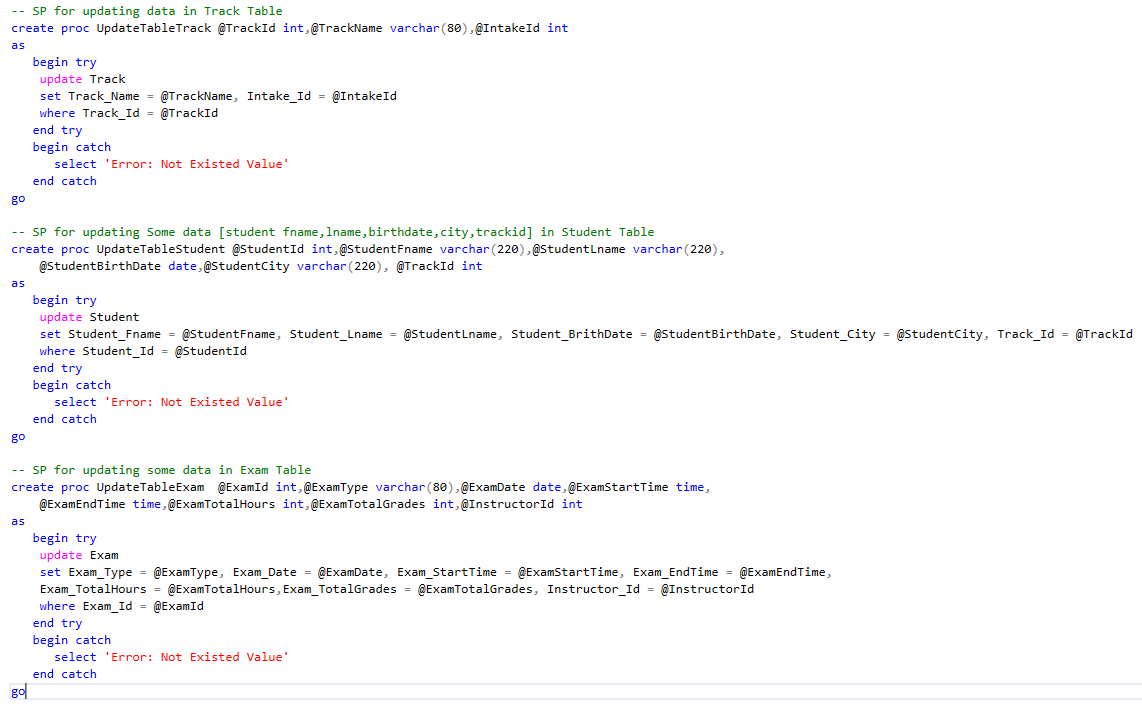
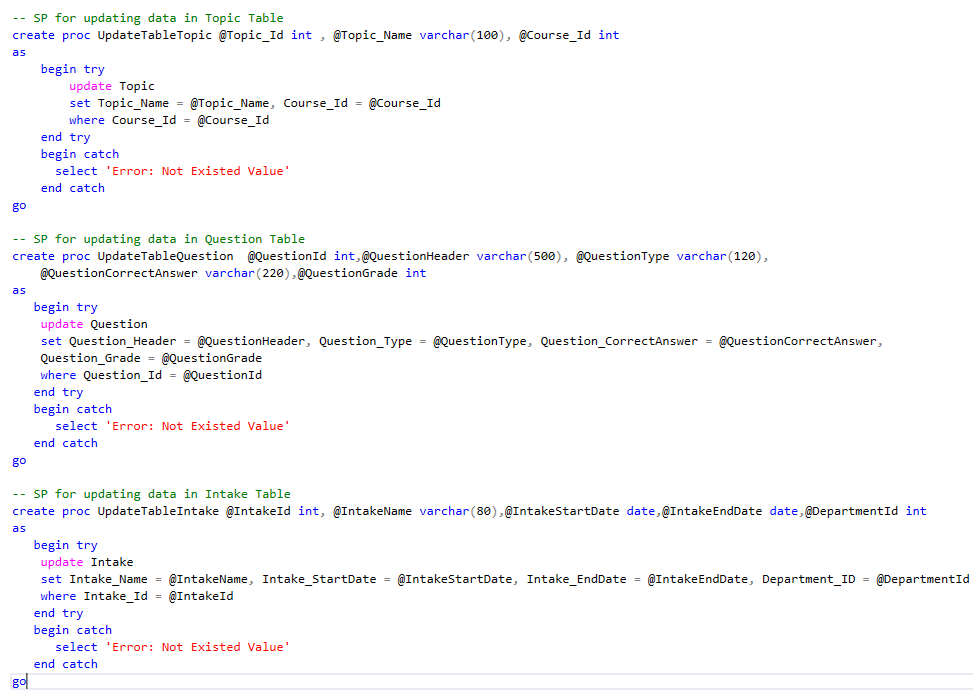
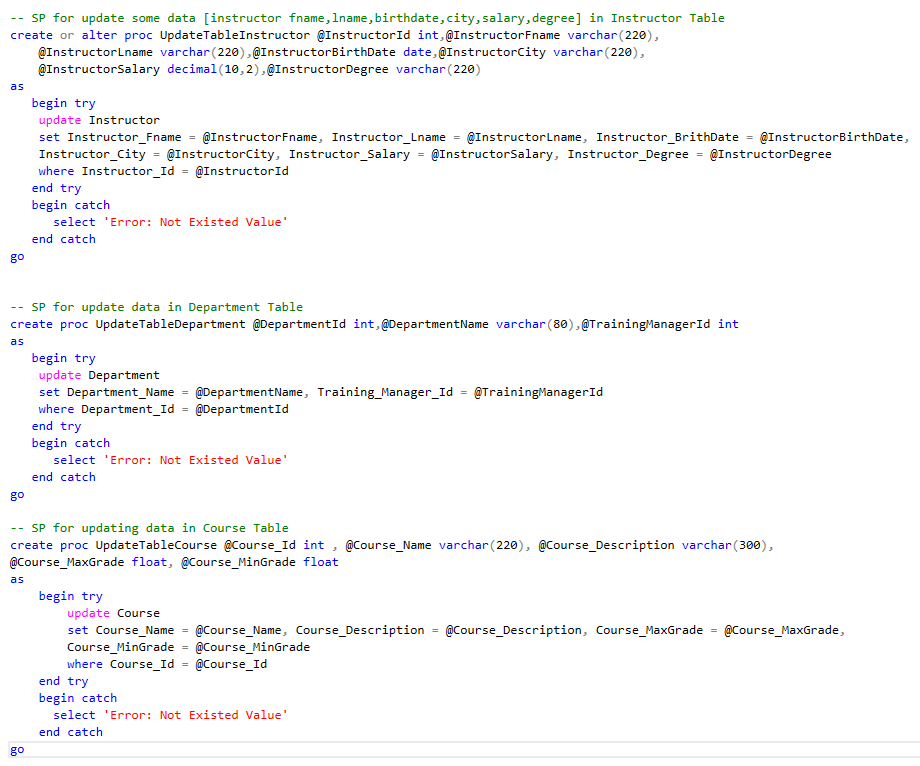
Designed to insert new records into the specified table by accepting various input parameters. It ensures data is added based on the provided values and catches any errors related to duplicate entries.





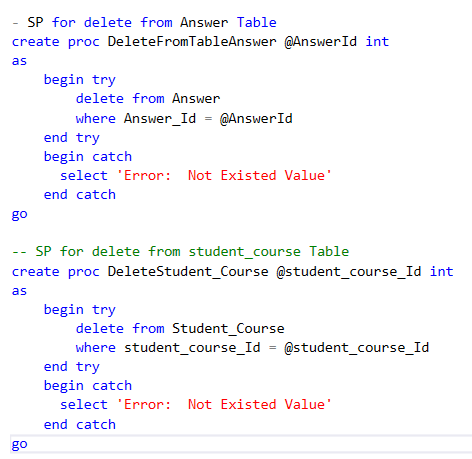
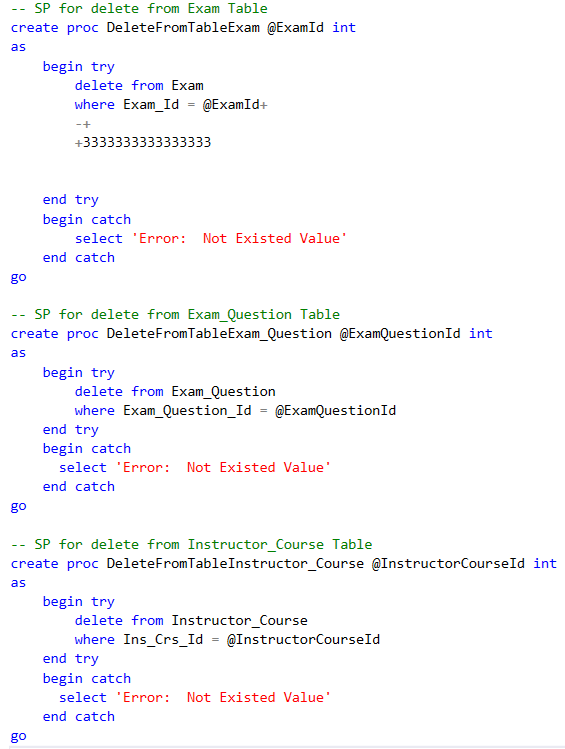
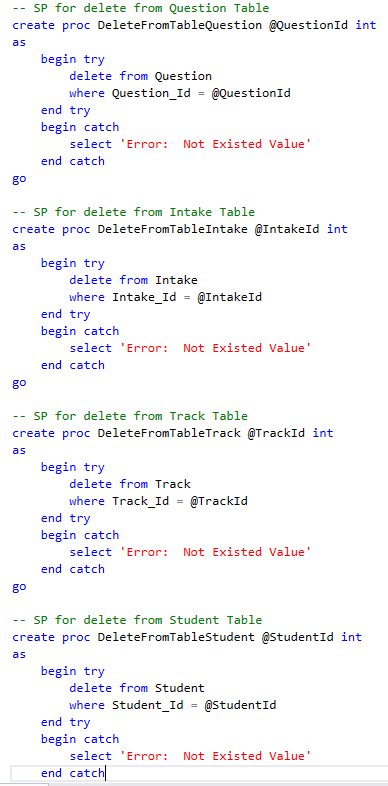
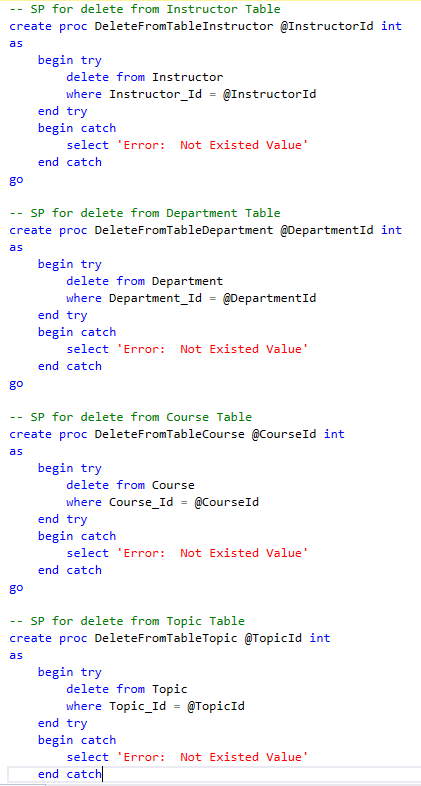
**4.15. Update Procedures:** For All Tables

Updates existing records in the specified table based on the provided parameters. It modifies the specified fields of a record identified by a unique ID and catches errors related to non-existent values.



**4.16. Delete Procedures:** For All Tables

Deletes a record from the specified table based on a unique identifier. It ensures that the targeted record is removed, and catches errors if the specified value does not exist.



**5. Views**

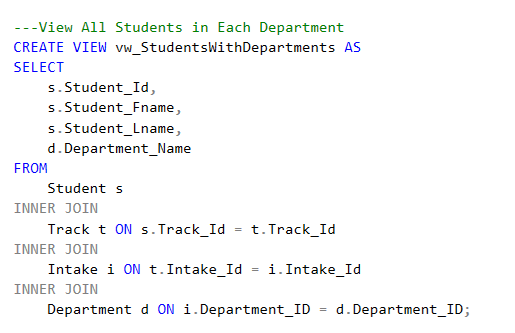
**5.1. View:** vw\_DepartmentsWithManagers

**View All Departments with Their Managers:** Displays all departments with their respective training managers by joining the Department and Instructor tables.

****

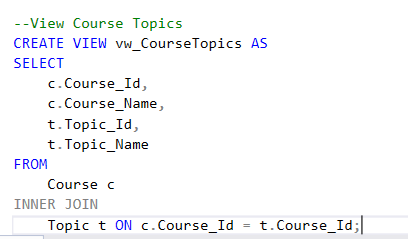
**5.2. View:** vw\_StudentsWithDepartments

**View All Students in Each Department:** Lists students along with the department they belong to by joining the Student, Track, Intake, and Department tables.



**5.3. View:** vw\_CourseTopics

**View Course Topics:** Shows all topics associated with each course by joining the Course and Topic tables.



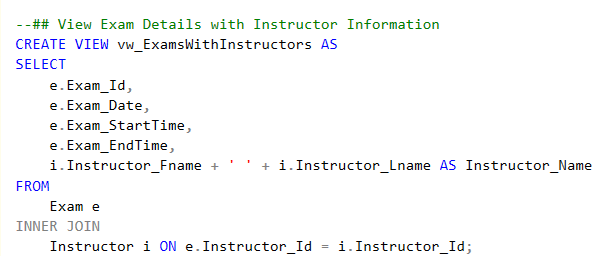
**5.4. View:** vw\_CoursesWithInstructors

**View All Courses with Instructor Names:** Displays courses along with the instructors assigned to them by joining the Course, Instructor\_Course, and Instructor tables.



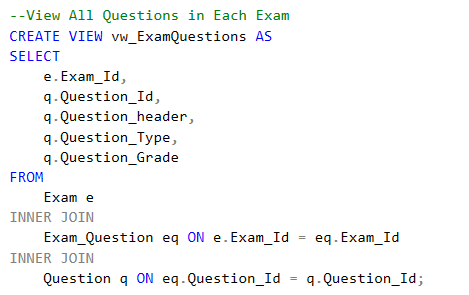
**5.5. View:** vw\_ExamsWithInstructors

**View Exam Details with Instructor Information:** Shows detailed exam information along with the instructor responsible for the exam by joining the Exam and Instructor tables.



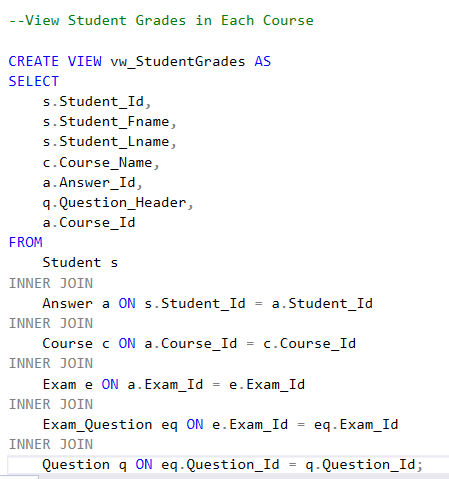
**5.6. View:** vw\_ExamQuestions

**View All Questions in Each Exam:** Lists all the questions included in each exam by joining the Exam, Exam\_Question, and Question tables.



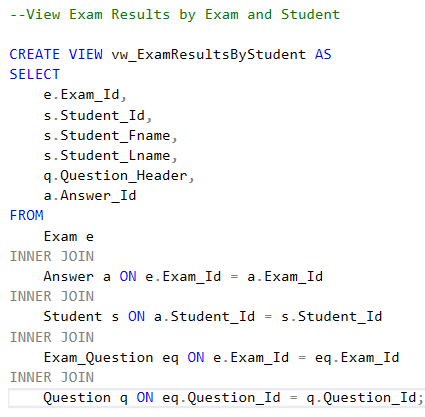
**5.7. View:** vw\_StudentGrades

**View Student Grades in Each Course:** Displays student information, the courses they are enrolled in, and their grades on exam questions by joining the Student, Answer, Course, Exam, Exam\_Question, and Question tables.



**5.8. View:** vw\_ExamResultsByStudent

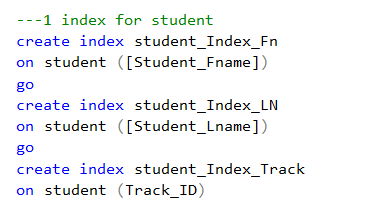
**View Exam Results by Exam and Student:** Lists the exam results of each student by joining the Exam, Answer, Student, Exam\_Question, and Question tables.



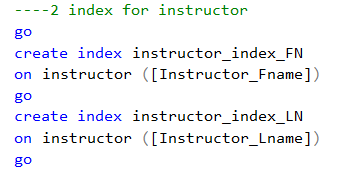
**6. Index**

An index in a database is a data structure that improves the speed of data retrieval operations on a table at the cost of additional storage space. Indexes are typically created on columns that are frequently searched, filtered, or used in JOINs, making queries more efficient by providing faster access to data.

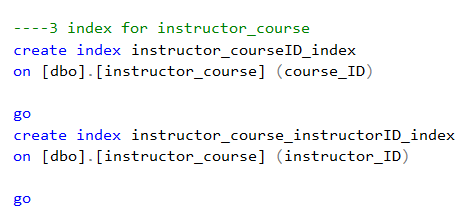
**6.1.** **Index:** For Student Table



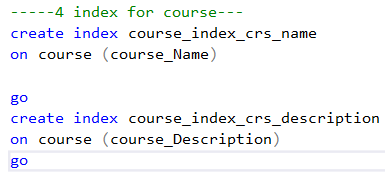
**6.2.** **Index:** For Instructor Table



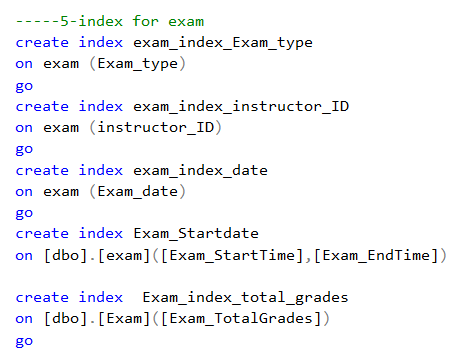
**6.3.** **Index:** For Instructor\_Course Table



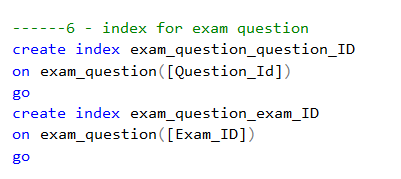
**6.4.** **Index:** For Course Table



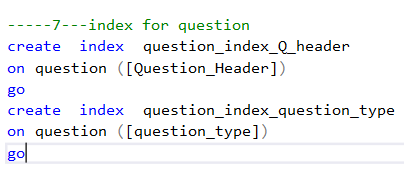
**6.5.** **Index:** For Exam Table



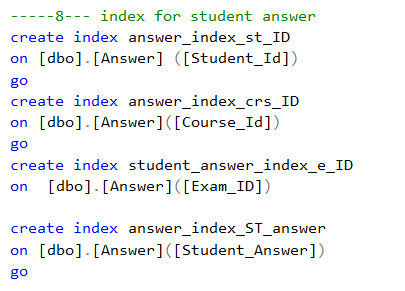
**6.6.** **Index:** For Exam\_Question Table



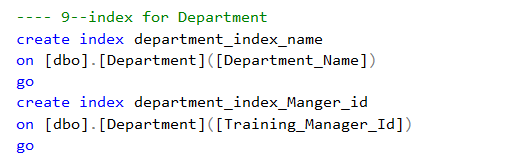
**6.7.** **Index:** For Question Table



**6.8.** **Index:** For Answer Table



**6.9.** **Index:** For Department Table

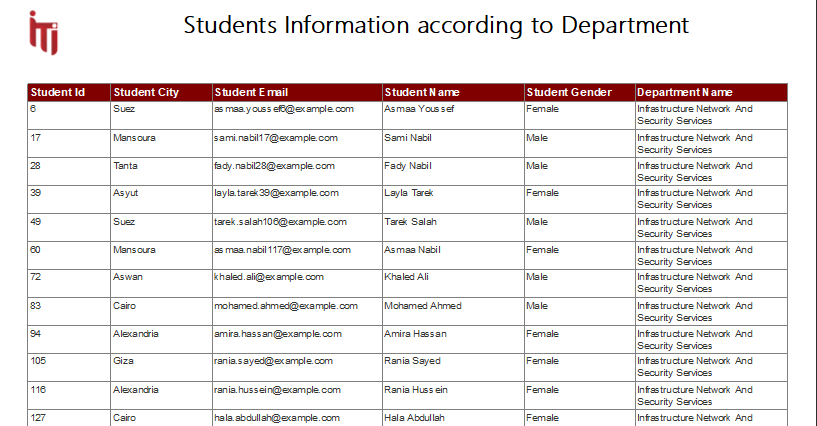


**6.10.** **Index:** For Student\_Course Table

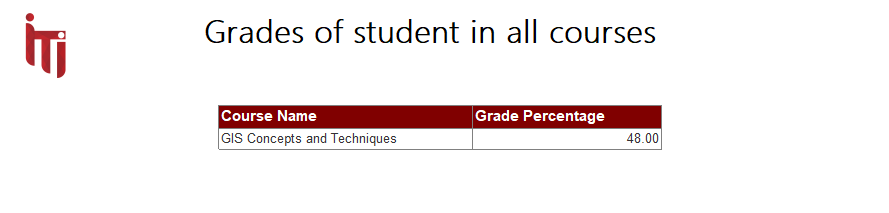


REPORTS

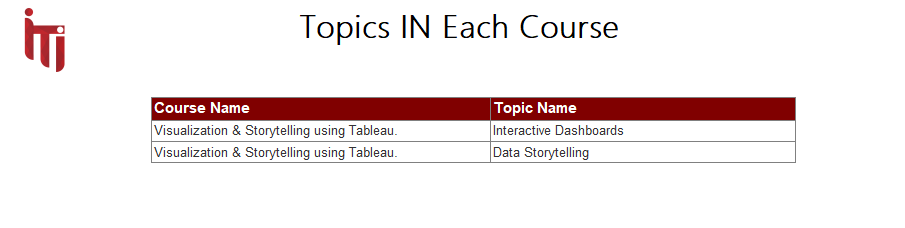
Student information summary report calling the department id by parameter from stored procedure made to take the department id and returns the student id, student city, student email, student Gender and Department Name.



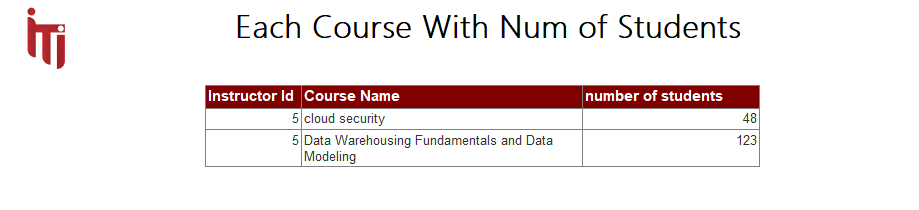
Student grades report calling the student id by parameter from stored procedure made to take the student id and return the names of the courses, and student grades.



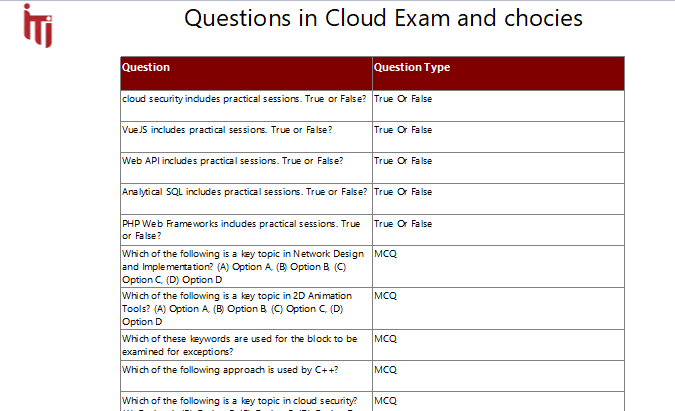
Topic by Course report calling the course id by parameter from stored procedure made to take the course id and returns the course, topic.



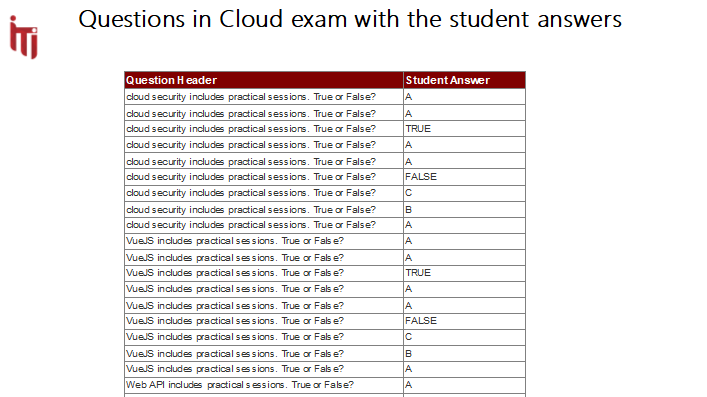
Instructor course summary report calling the instructor id by parameter from stored procedure made to take the instructor id and returns instructor id, course name, and the number of students.



Exam report calling the exam number by parameter from stored procedure made to take the exam number and return the questions, and the Type.



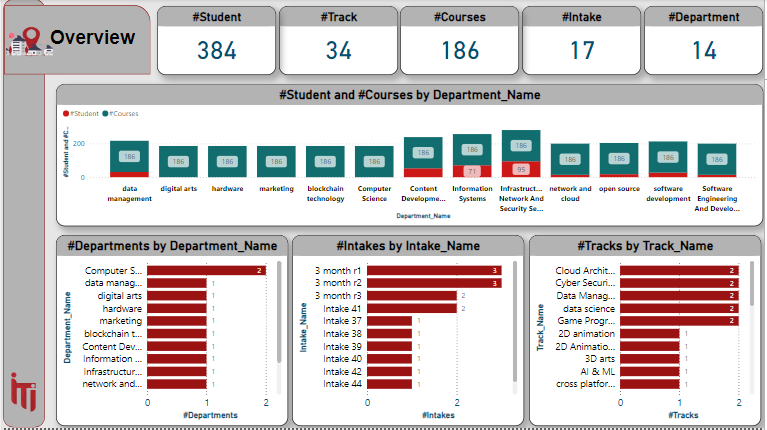
Exam report calling the exam number by parameter from stored procedure made to take the exam number and return the questions, and the answers.



POWER BI DASHBOARDS

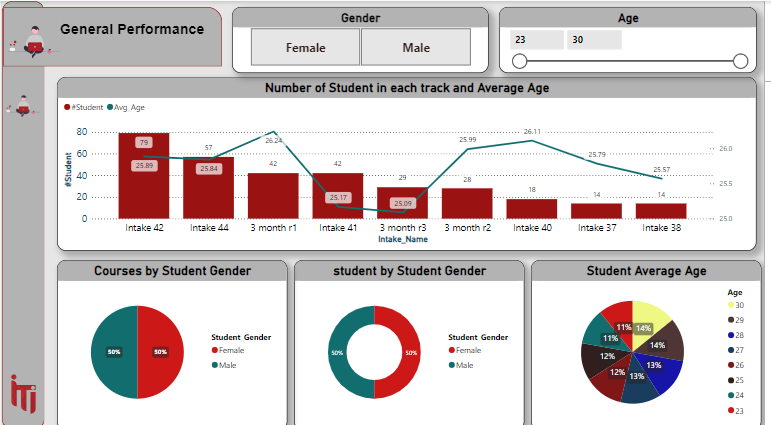
**Overview**

* The dashboard provides a comprehensive view of a training program, including information on instructors, tracks, courses, intakes, and departments.
* Key metrics are summarized in the top section, showing the total number of instructors, tracks, courses, intakes, and departments.



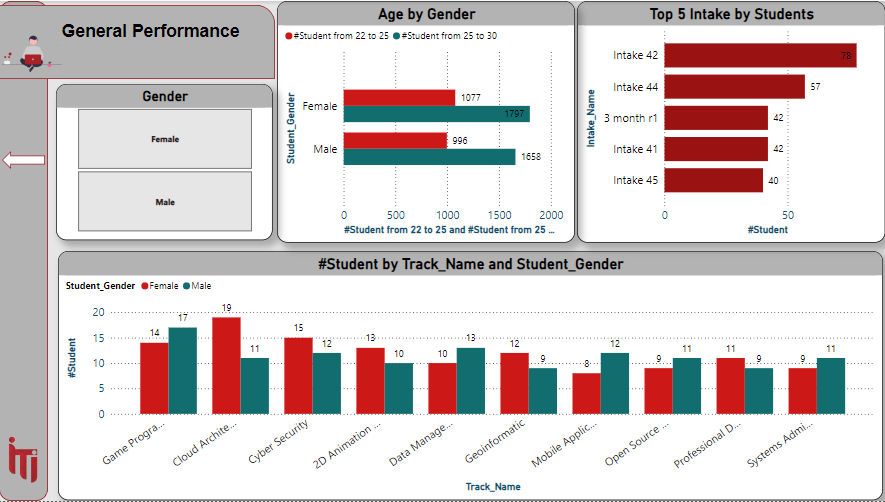
**General Performance**

* The dashboard provides a comprehensive overview of student performance across various dimensions, including gender, age, department, and track.
* Key metrics like average age, average grade, and student count are presented in a clear and concise manner.



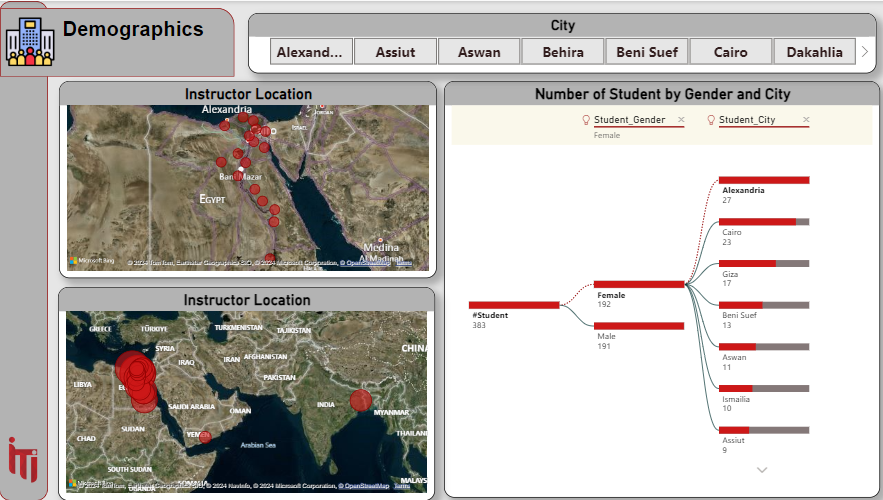
**General Performance**

* The dashboard provides a comprehensive overview of student performance across various dimensions, including gender, age, intake, and track.
* Key metrics like student count and intake popularity are presented in a clear and concise manner.



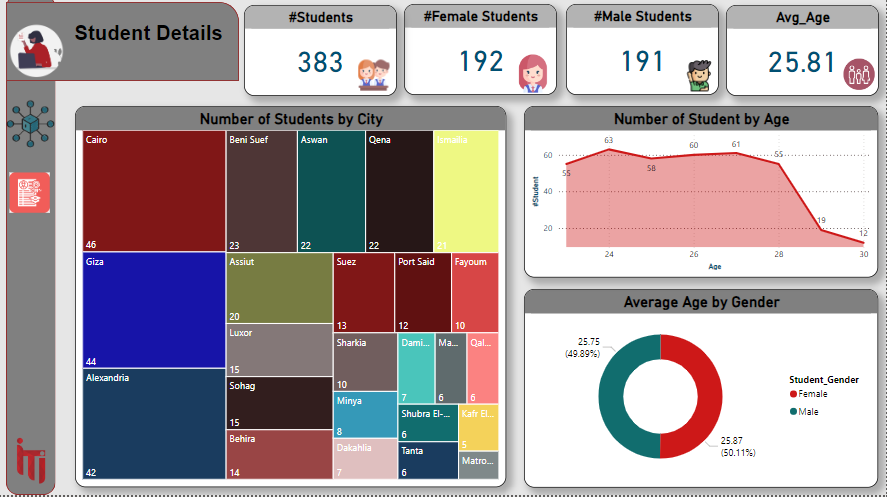
**Demographics**

* The dashboard provides a comprehensive overview of instructor demographics, including location and gender distribution.
* The map visualizations effectively depict the geographical distribution of instructors.



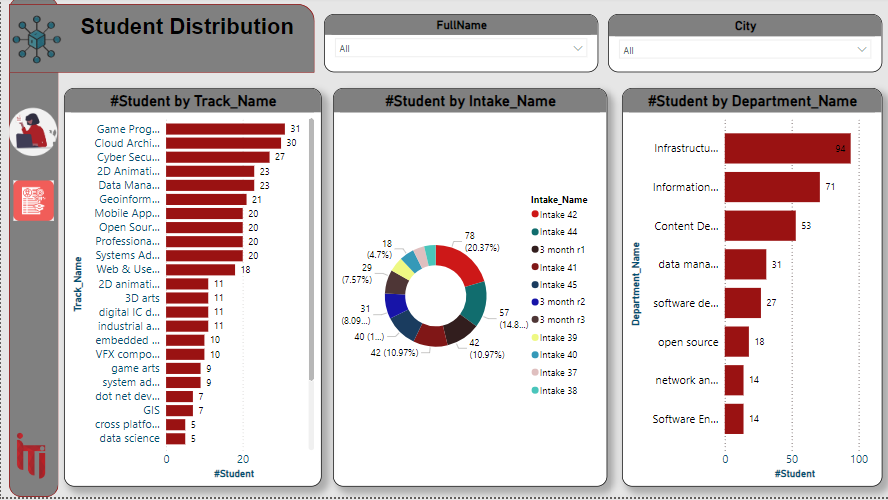
**Student Details**

* The dashboard provides a comprehensive overview of student demographics, including total number of students, gender distribution, and average age.



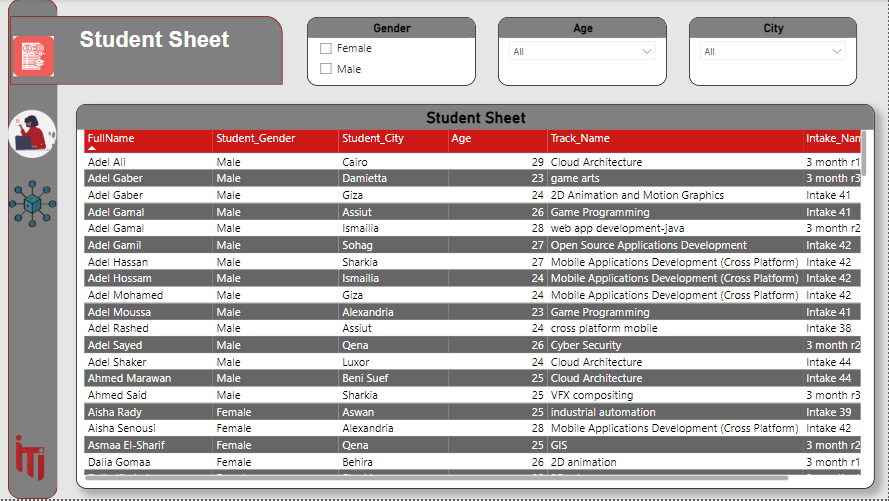
**Student Distribution**

* The dashboard provides a comprehensive overview of student distribution across tracks, intakes, and departments.
* The bar charts effectively visualize the number of students enrolled in each category.



**Student Sheet**

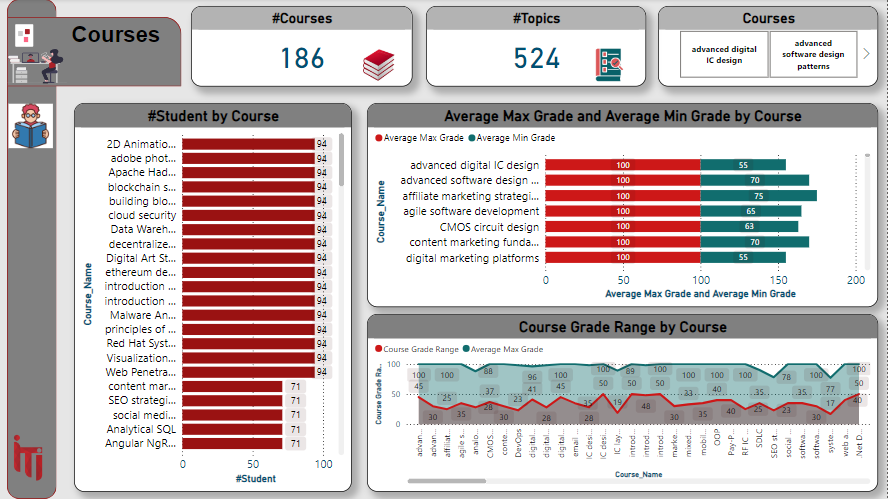
* the student sheet provides a detailed dataset of individual student information



**Courses**

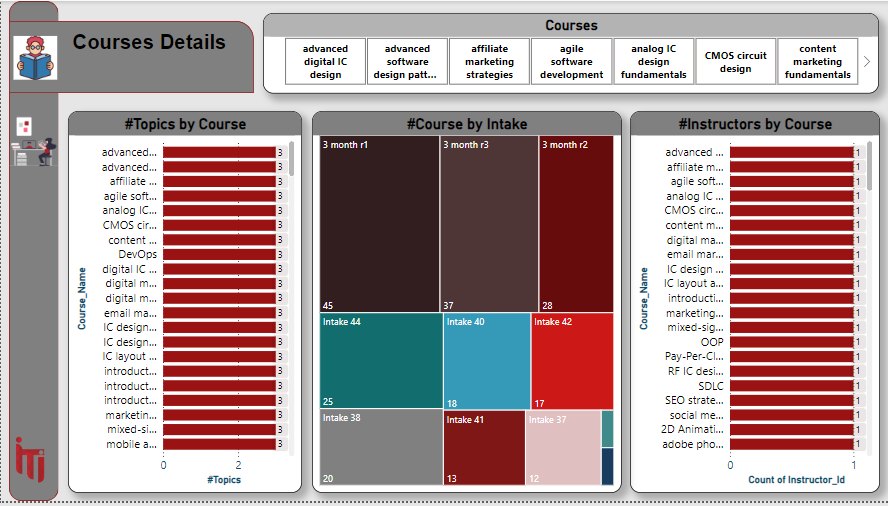
 The dashboard provides a comprehensive overview of course information, including the number of courses, topics, and students enrolled in each course.

 Key metrics like average max and min grades are also presented.



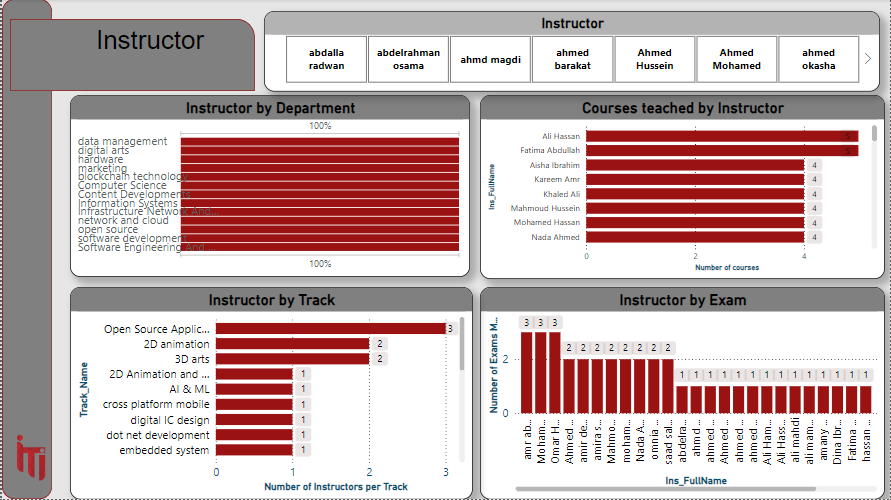
**Courses Details**

* The dashboard provides a comprehensive overview of course information, including the number of topics, intakes, and instructors associated with each course.



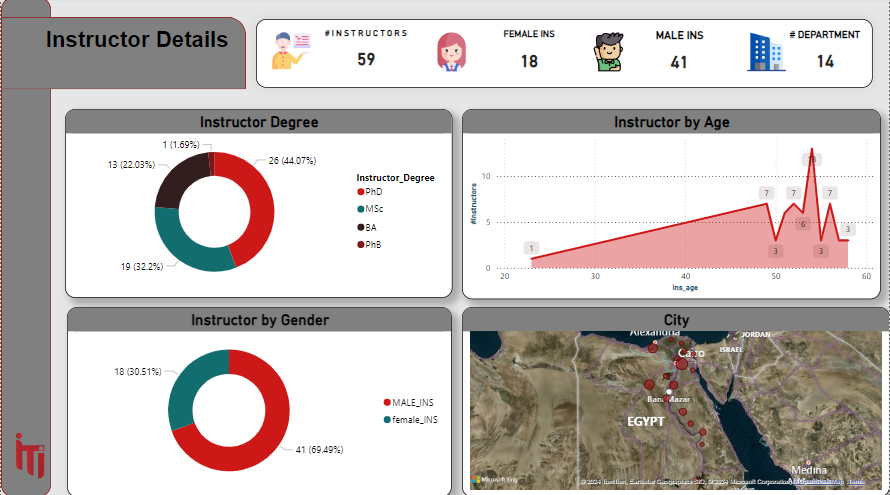
**Instructor**

* The dashboard provides a comprehensive overview of instructor information, including their names, departments, courses taught, and exam involvement.



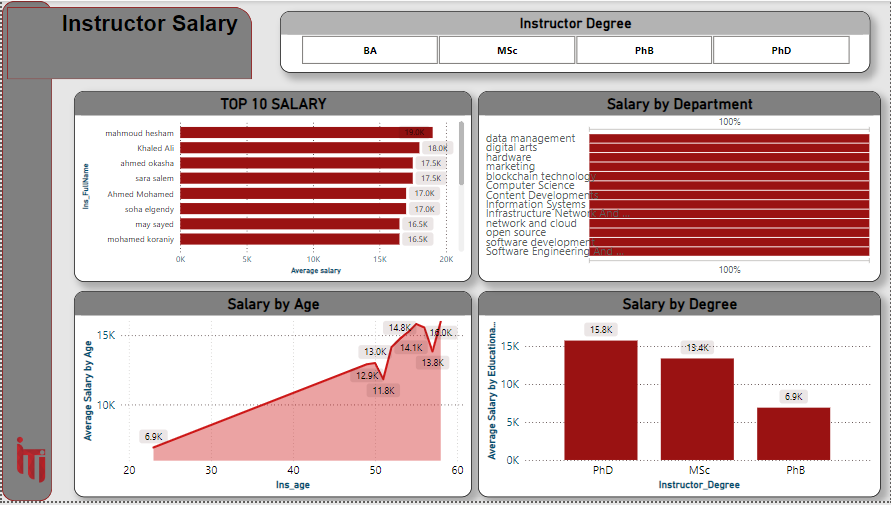
**Instructor Details**

* The dashboard provides a comprehensive overview of instructor demographics, including the total number of instructors, gender distribution, age distribution, and educational qualifications.



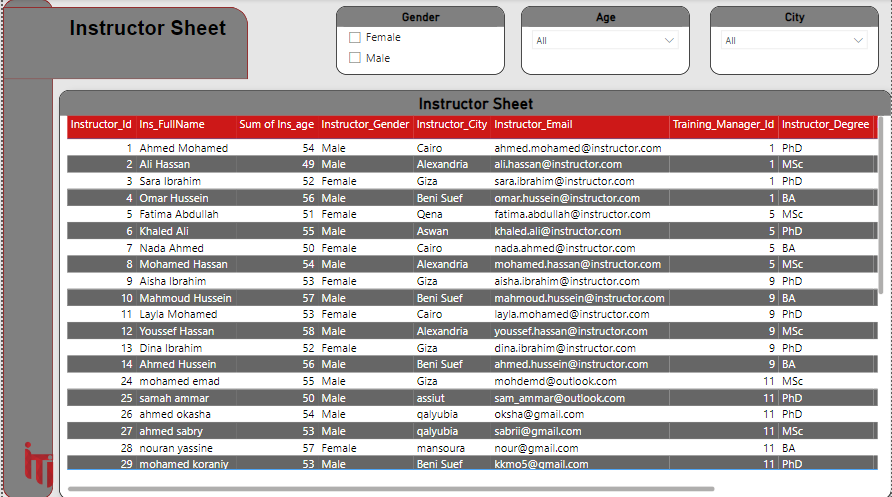
**Instructor Salary**

* The dashboard provides a comprehensive overview of instructor salaries, including top earners, salary by department, age, and degree.



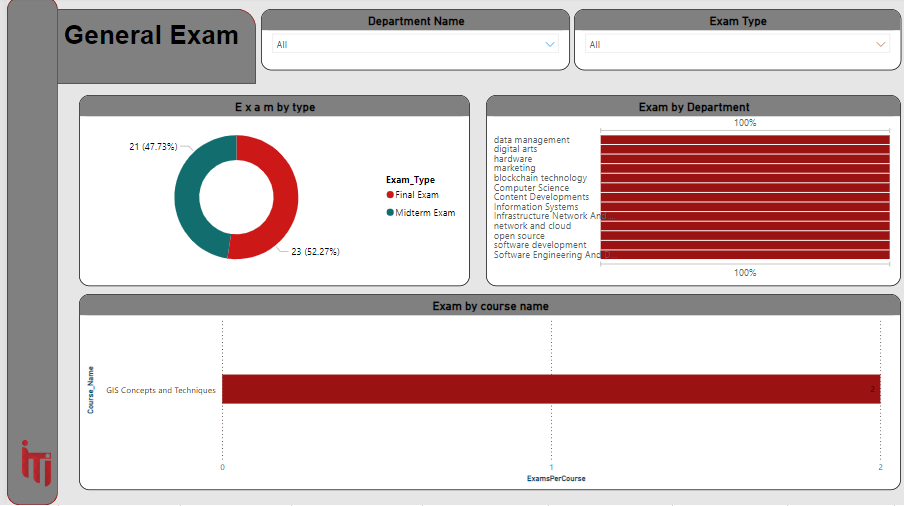
**Instructor Sheet**

* The instructor sheet provides a detailed dataset of individual instructor information



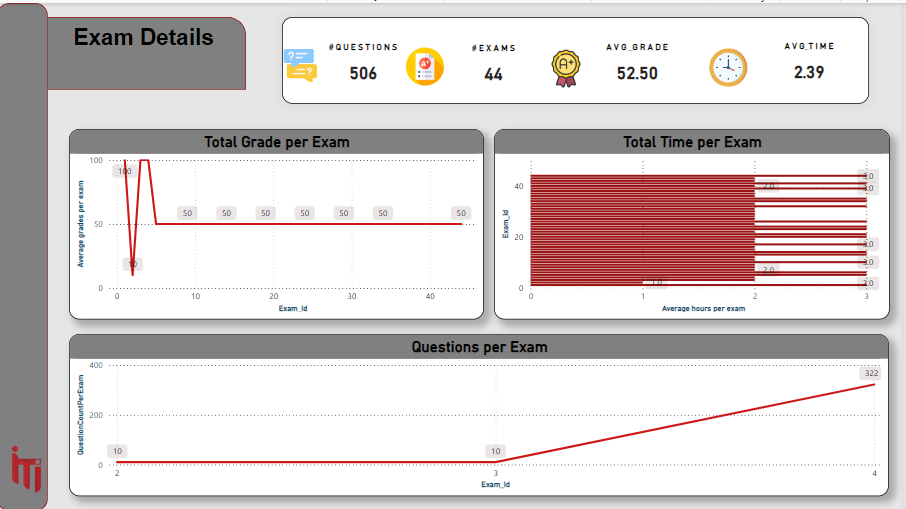
**General Exam**

* The dashboard provides a comprehensive overview of exam data, including the distribution of exam types, departments, and courses.



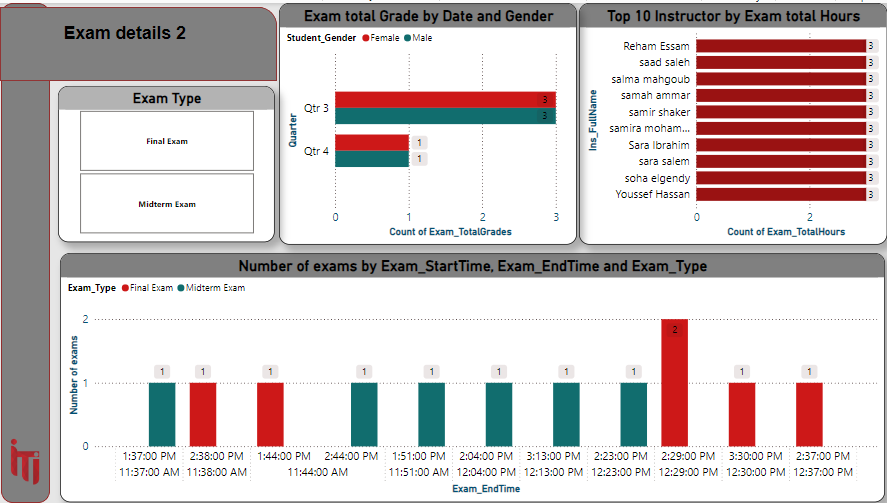
**Exam Details**

* The dashboard provides a comprehensive overview of exam data, including the total number of questions, exams, average grade, and average time taken.



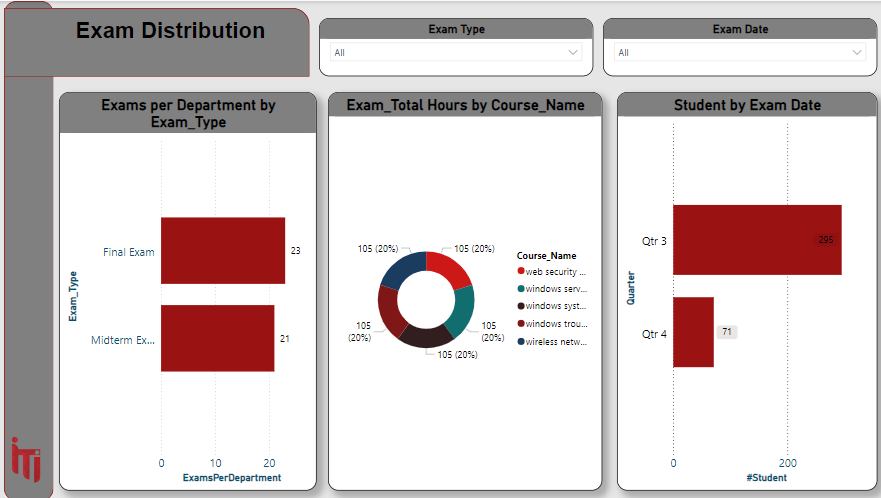
**Exam details 2**

* The dashboard provides a comprehensive overview of exam data, focusing on gender, exam type, and instructor performance.



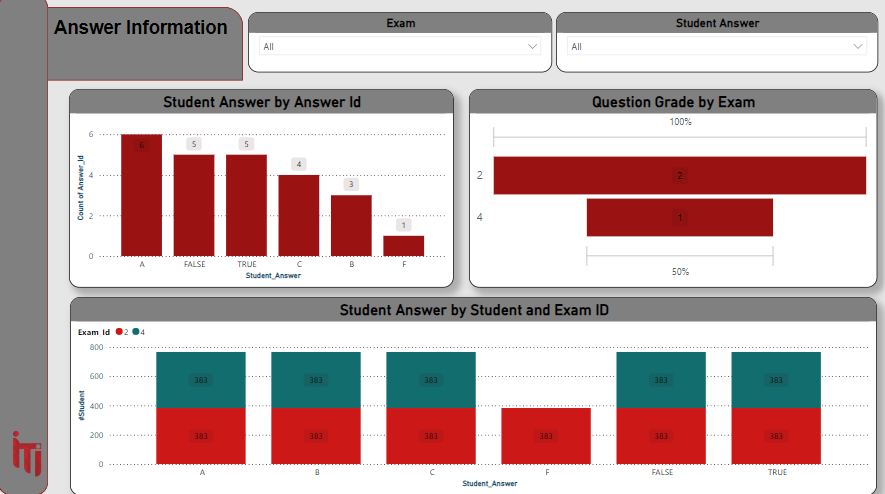
**Exam Distribution**

* The dashboard provides a comprehensive overview of exam data, focusing on exams per department, total hours by course, and student distribution by exam date.



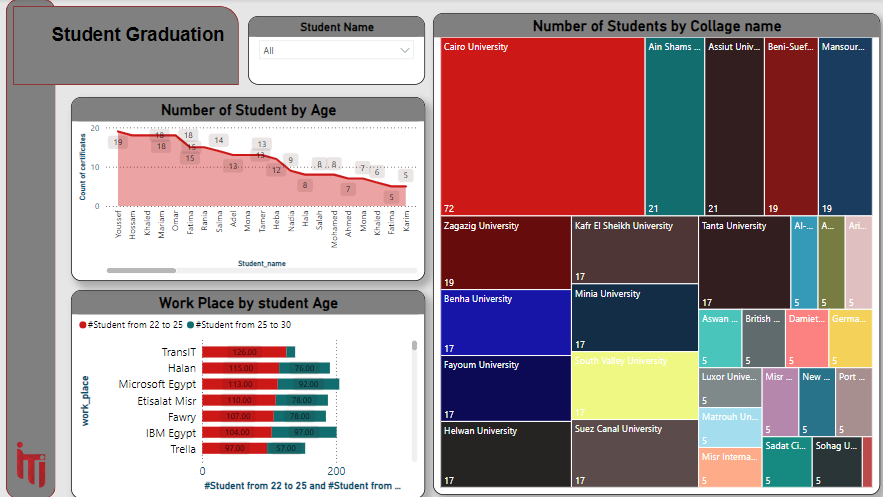
**Answer Information**

* The dashboard provides a comprehensive overview of student answers, including the distribution of answer IDs, question grades, and student responses.



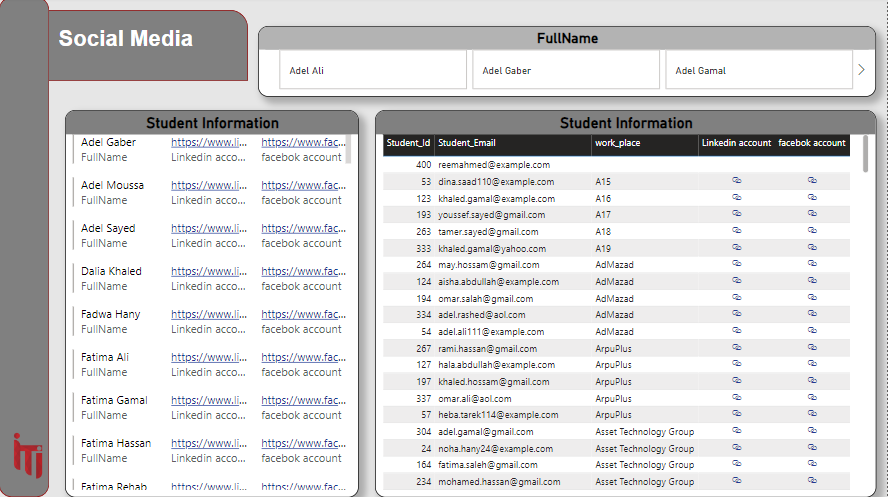
**Student Graduation**

* The dashboard provides a comprehensive overview of student graduation data, focusing on student distribution by college, age, work placement, and certificates earned.



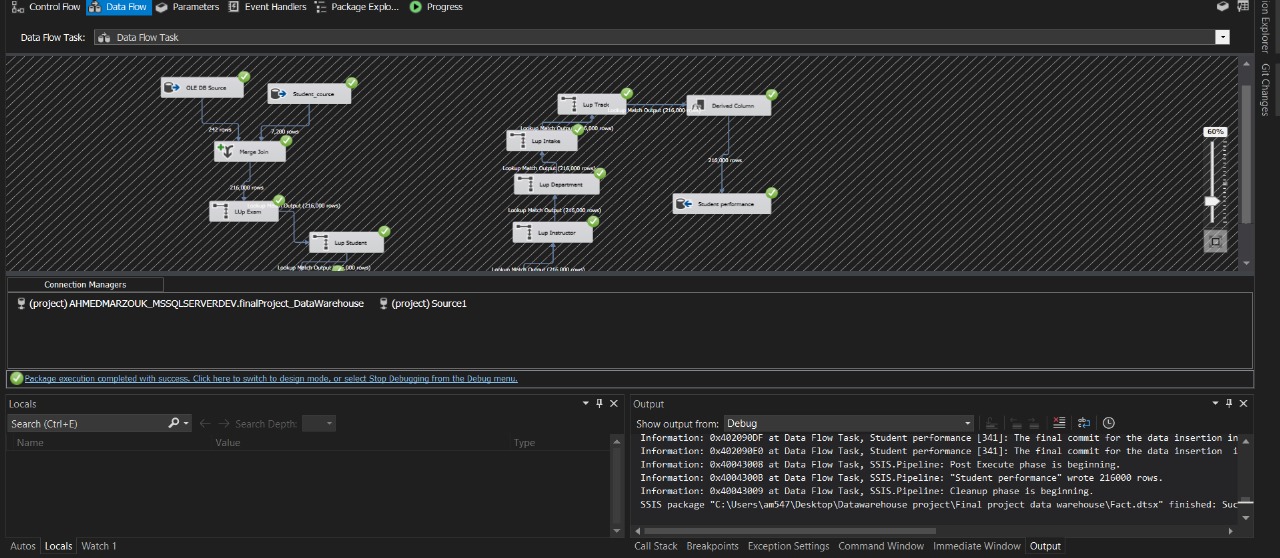
**Social Media**

* The dashboard provides a comprehensive overview of student social media information, including full names, student IDs, emails, workplaces, and links to LinkedIn and Facebook accounts.

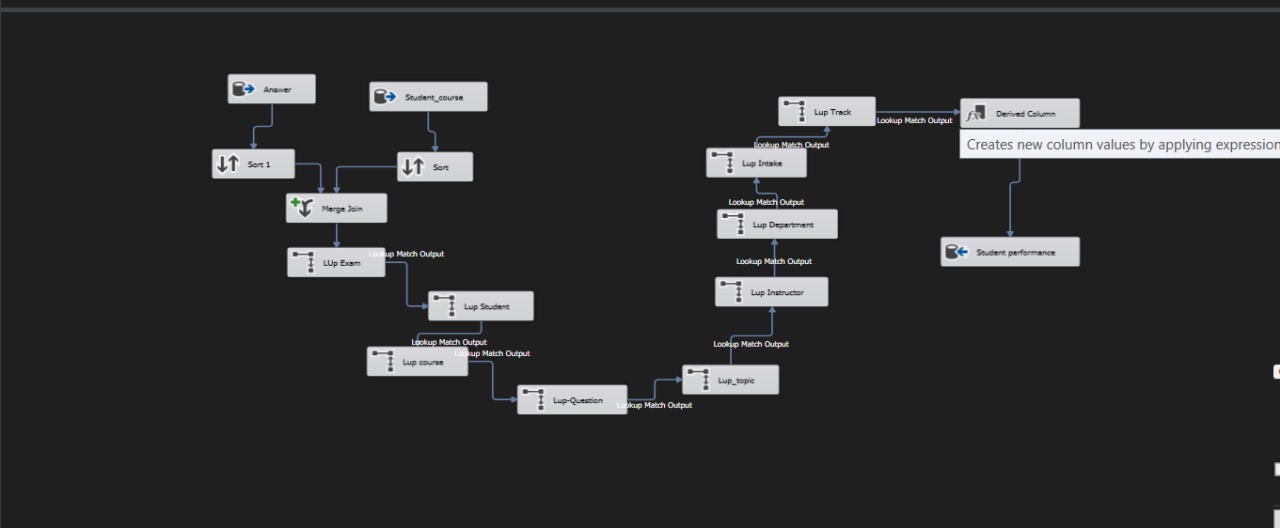


SSIS

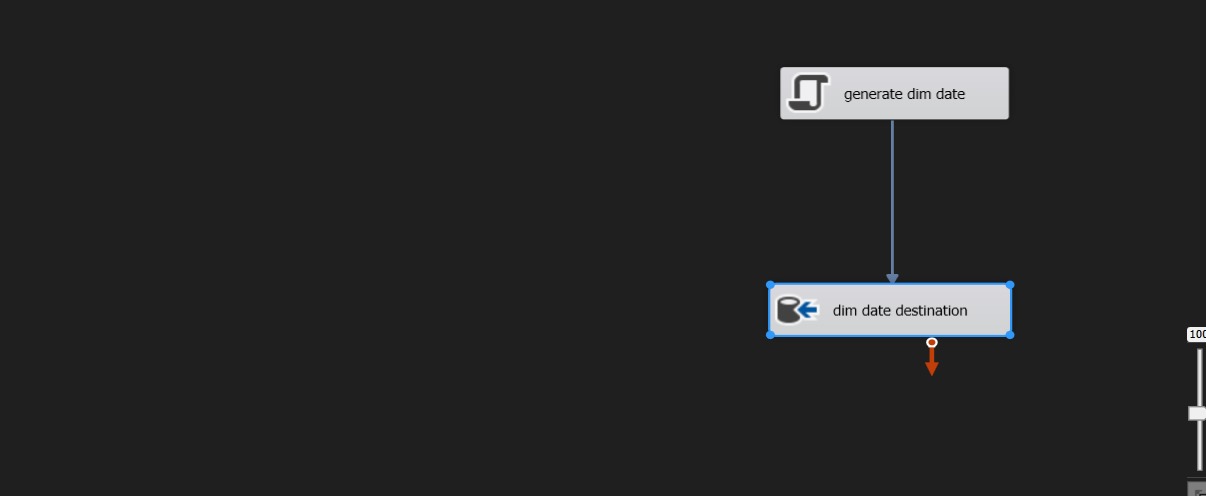
Fact table (Student Performance)



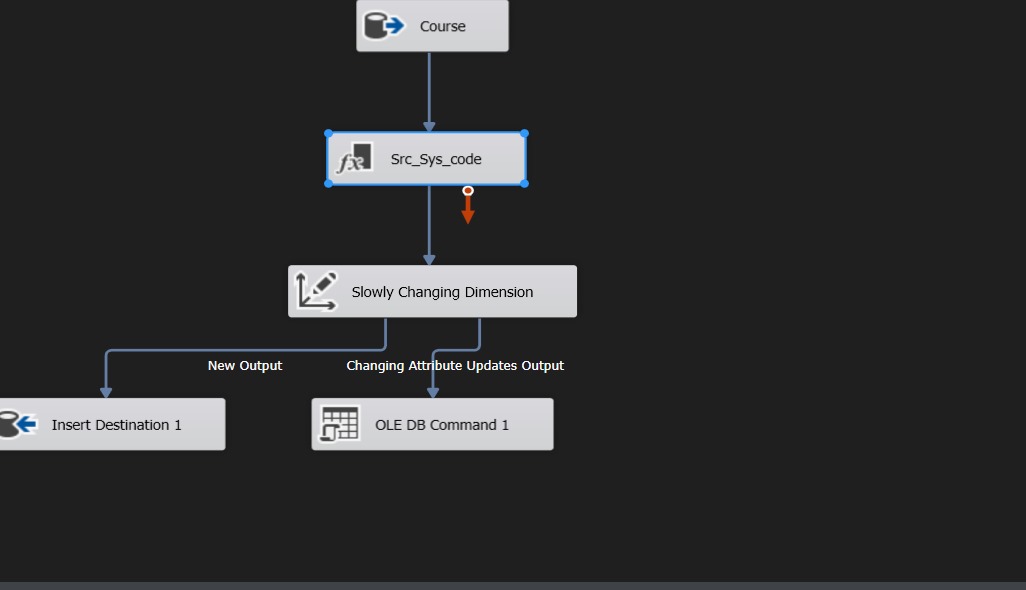
Dim Answer



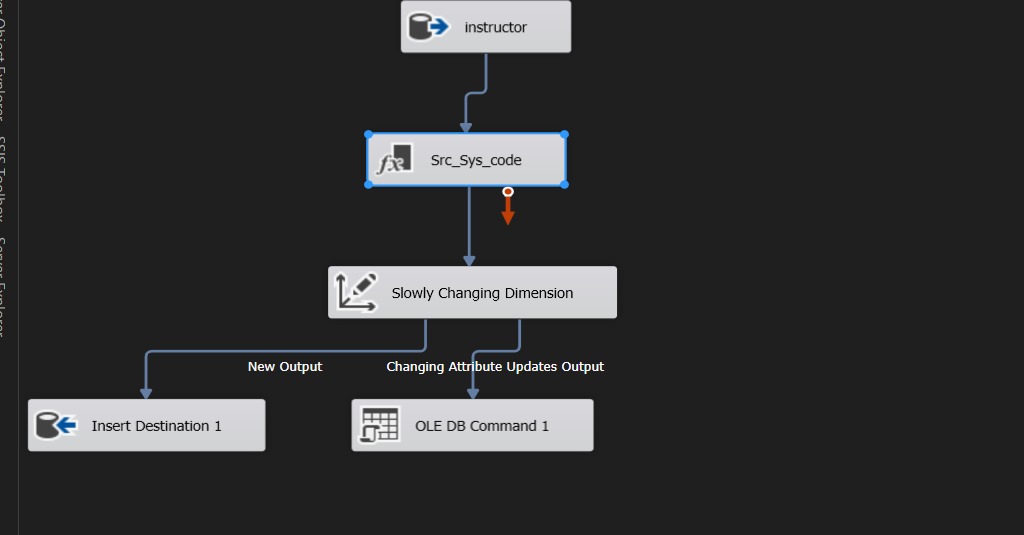
Dim Date



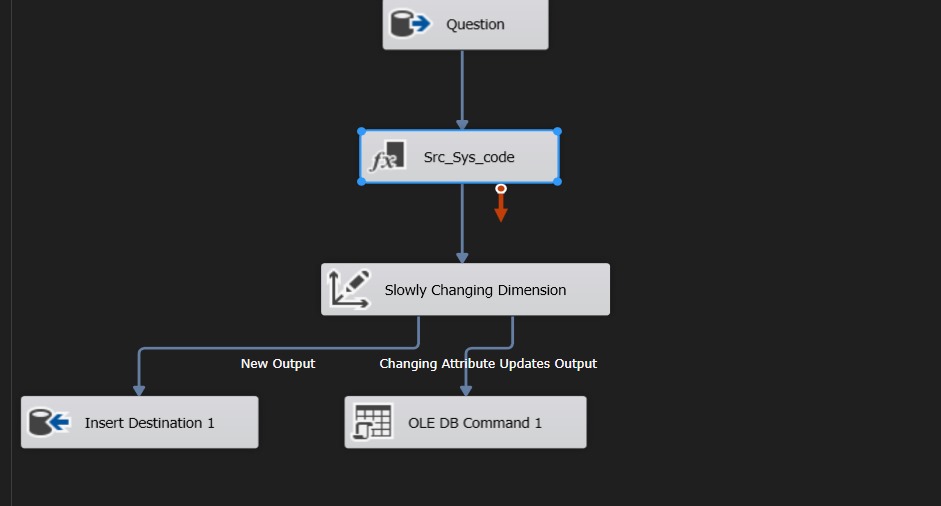
Dim Course



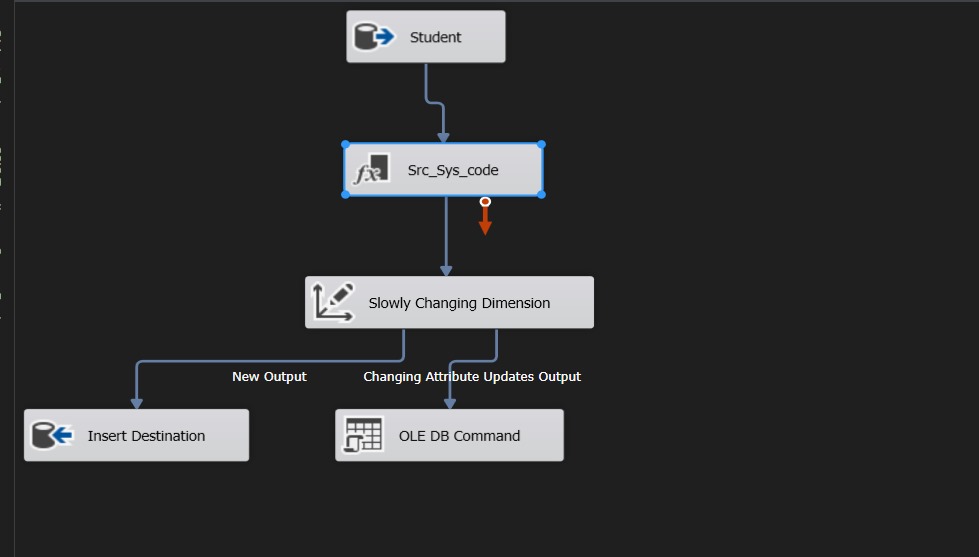
Dim Instructor



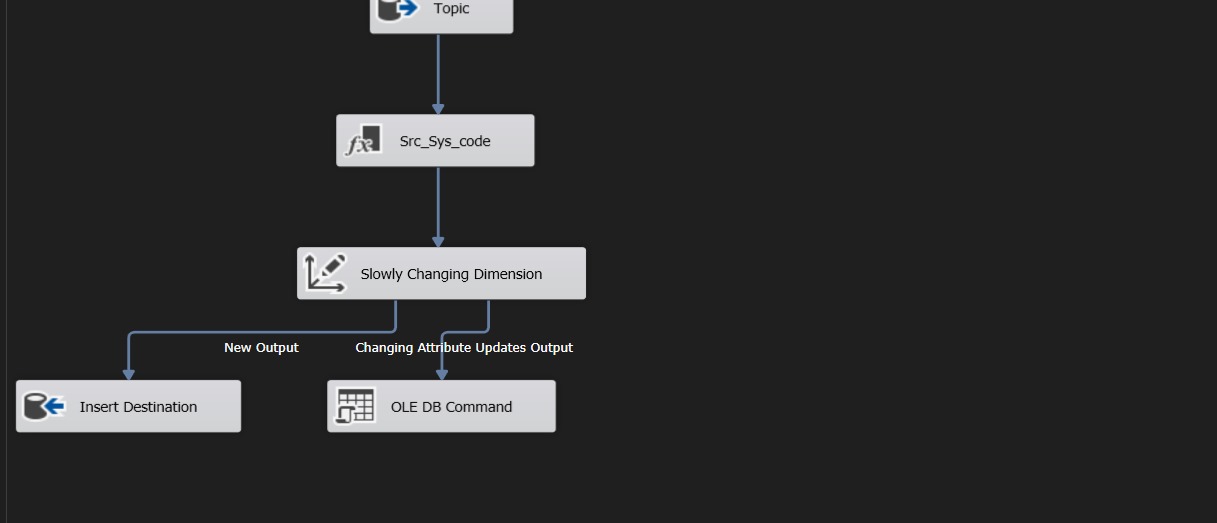
Dim Question



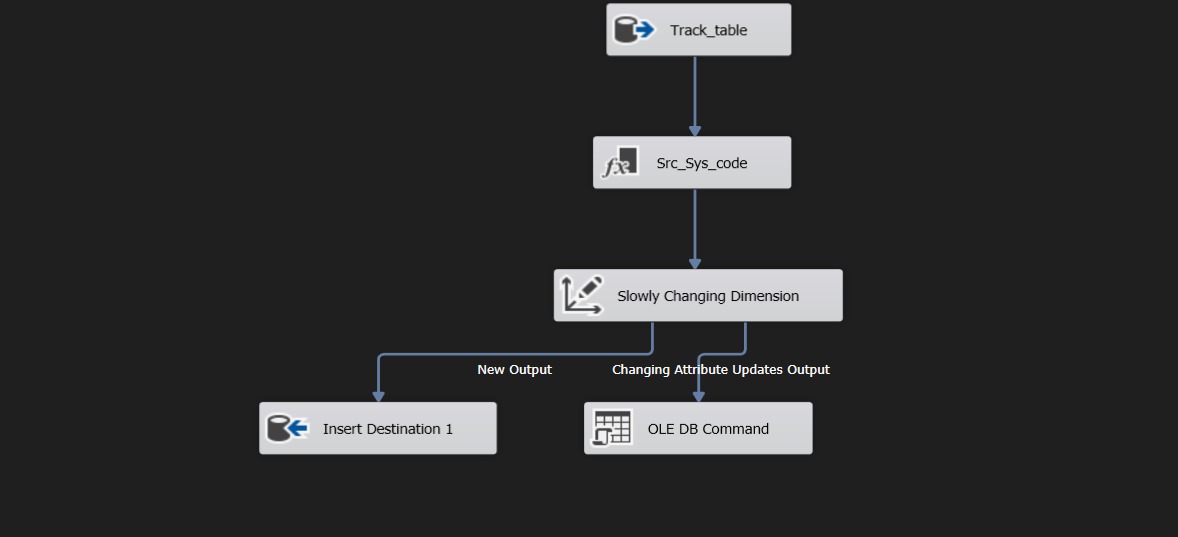
Dim Student



Dim Topic

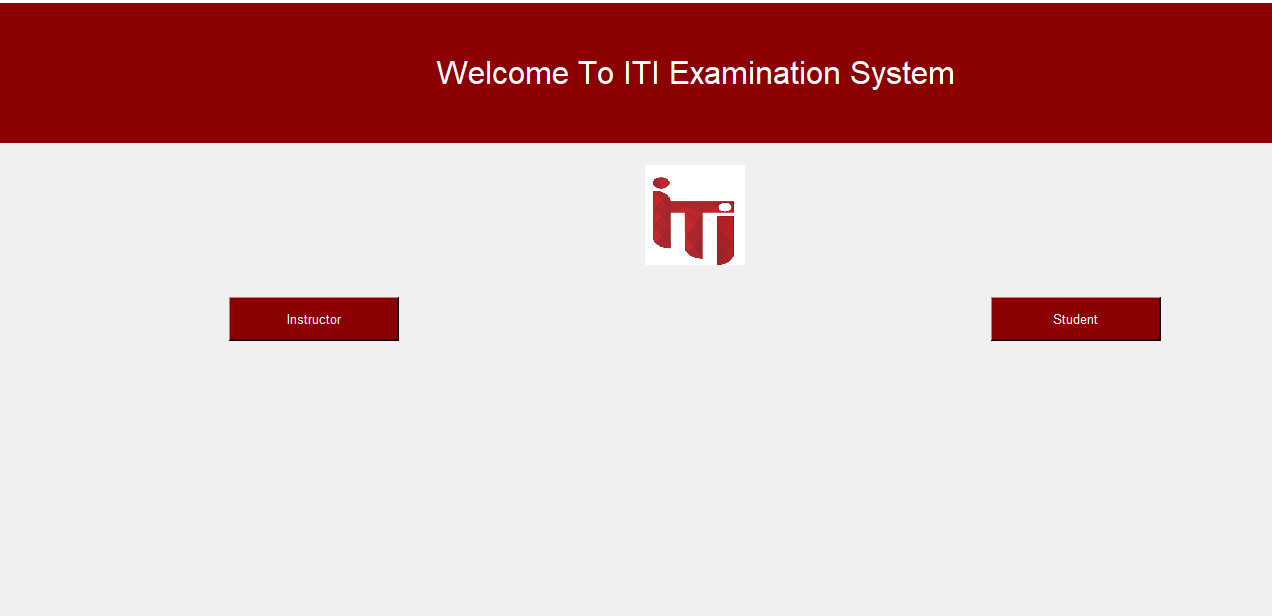


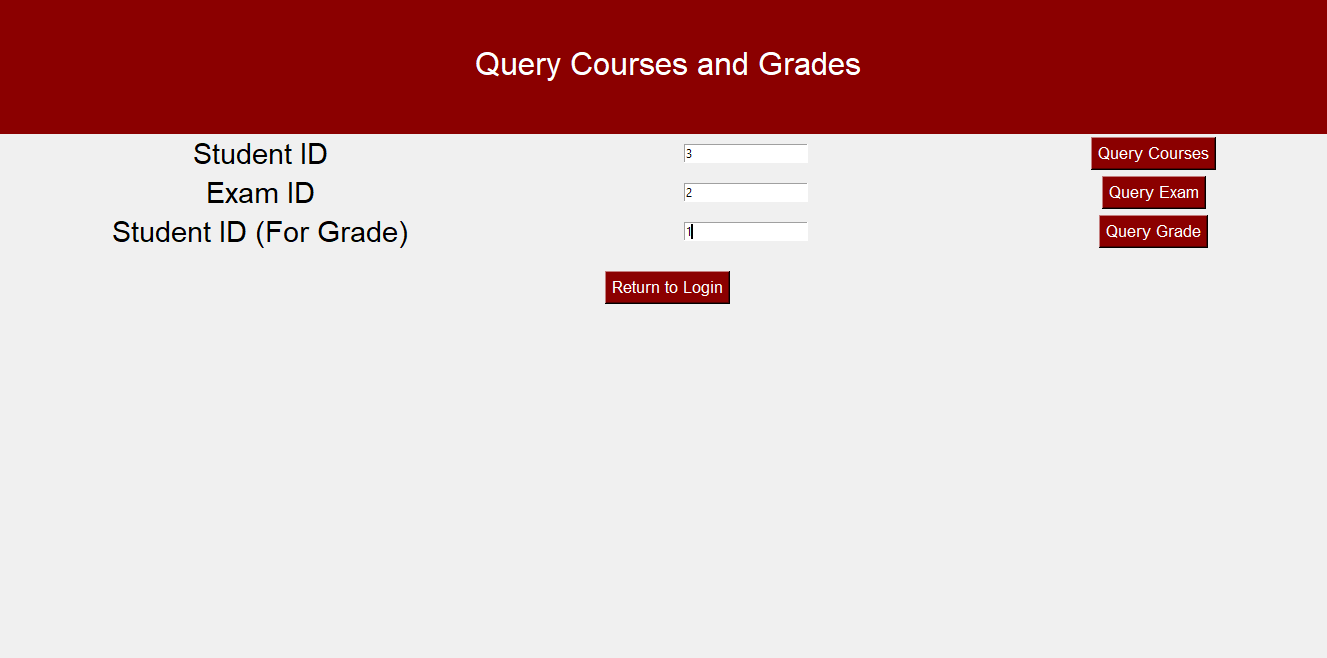
Dim Track

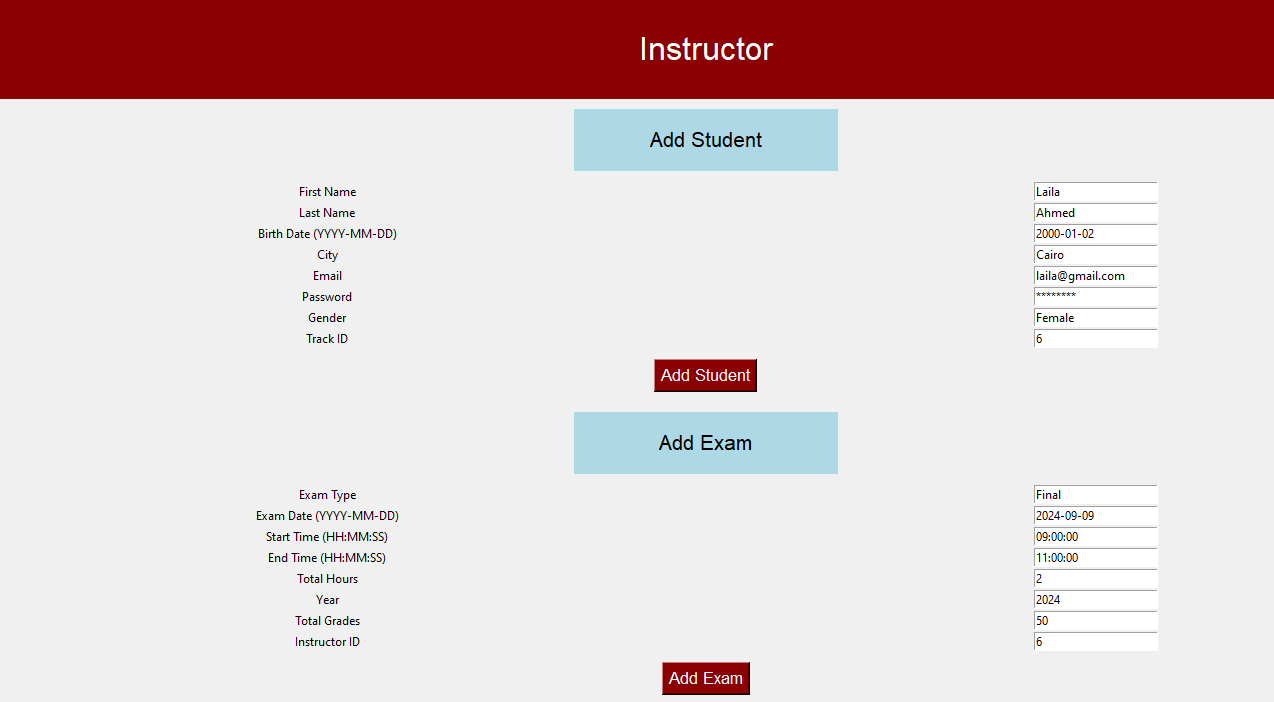


Application

Simple Application help instructor to add Student and add Exam  
help to know Student Courses, Student degree and Exam Details.







CONCLUSION

The Examination System project successfully addresses the critical need for a robust and automated system to manage the complex processes of exam creation, administration, and evaluation within an educational environment. By leveraging a well-structured database design and integrating advanced functionalities such as automated exam generation, dynamic question assignment, and detailed student performance tracking, the system enhances the overall efficiency and reliability of examination management.

The use of Microsoft SQL Server for database management, combined with the powerful reporting capabilities of SQL Server Reporting Services (SSRS) and Power BI, provides stakeholders with comprehensive insights into exam results, instructor performance, and student progress. These tools enable data-driven decision-making, allowing educators to identify areas for improvement and optimize teaching strategies to better support student learning outcomes.

Throughout the development process, careful attention was given to the implementation of best practices in database design, including the creation of Entity-Relationship Diagrams (ERDs), thorough mapping of database entities, and the establishment of effective data retrieval and reporting mechanisms. The project also ensures data integrity, security, and scalability, making it a viable solution for current and future needs.

Overall, the Examination System project represents a significant advancement in the way examinations are managed, offering a scalable and adaptable framework that can be extended and enhanced as educational requirements evolve. It provides a solid foundation for further enhancements, such as the integration of more advanced analytics, additional reporting features, and potential expansions to support new examination formats and educational models.

# REFERENCES

* Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. Pearson Education.
* Microsoft Docs. (n.d.). SQL Server Database Design and Implementation. Retrieved from https://docs.microsoft.com/.
* Microsoft Docs. (n.d.). SQL Server Reporting Services (SSRS). Retrieved from https://docs.microsoft.com/en-us/sql/reporting-services/.
* Bovey, C., & Shinder, M. (2020). Mastering Power BI: Expert techniques for effective data analytics and business intelligence. Packt Publishing.
* Chandra, D., & Wagh, A. (2021). Hands-On Business Intelligence with DAX: Discover how to build rich data models and learn DAX from scratch. Packt Publishing.
* Microsoft Docs. (n.d.). Power BI Documentation. Retrieved from https://docs.microsoft.com/en-us/power-bi/.
* <https://youtu.be/1yRQPZB1mi8?si=_ms3f6SbtBUU3I-o>
* <https://youtu.be/SyghLKIDymo?si=0E94g0F2e7Xs5l4n>
* <https://youtu.be/hi3sjWeSwUg?si=SYLNttworxT0ItA7>