**Power Bi Track (Cairo UNI. Branch)**

**Graduation Project**

**(Examination System for ITI)**

**Documentation**



**Performed By:**

**Ahmed Saber**

**Amr Abdel-Gawad**

**Asmaa Abdel-Mohsen**

**Abanoub Ibrahim**

**Heba Amgad**

Abstract

**The purpose of this project is to design and simulate an examination system for the Information Technology Institute (ITI). This system is being developed as a graduation project and aims to automate the entire examination process for the ITI.**

**The examination system will store all relevant information about the students, tracks, branches, and instructors of the ITI. It will also have the capability to generate exams for each course by selecting questions from a question bank in a random manner. This will help ensure the validity and reliability of the exams and prevent cheating.**

**The system will record the examination results and store them for future reference. This will provide a comprehensive record of the students' performance and help in the evaluation process. The examination system will streamline the entire examination process and make it more efficient and organized.**

**It will also provide a centralized database of all examination-related information, making it easy to access and manage. In this document, we will outline the procedures and steps involved in the development of the examination system, and identify all the system requirements, processes, inputs, and outputs. Further sections will develop into each aspect in greater detail**

Introduction

The Information Technology Institute (ITI) examination system project is a graduation project aimed at developing an efficient and comprehensive examination system. The system is designed to cater to the needs of the ITI and is intended to record all necessary data related to students, tracks, branches, and instructors.

The project lifecycle is comprised of several critical stages, starting with the development and enhancement of the system scenario. This stage involves a thorough analysis of the examination requirements and the identification of potential solutions to meet those requirements. The next stage is the creation of an Entity Relationship Diagram (ERD) that helps to visualize the relationships between the entities in the system.

Once the ERD is completed, the next stage is to create the database from scratch. The data is then imported into the database, which is further used for various purposes. To ensure that the system is functional, stored procedures are developed to perform specific tasks within the system.

The data is then visualized through Power BI, which provides valuable insights into various aspects of the data. This helps to monitor the performance of the students, tracks, branches, and instructors, and to make data-driven decisions to improve the examination system.

In conclusion, the ITI examination system project is a comprehensive and well-structured project aimed at improving the examination process at ITI. The project is designed to meet the needs of the ITI and provides an efficient and effective solution for conducting exams, recording data, and analyzing results

System Scenario

The Examination System for the Information Technology Institute (ITI) is an comprehensive solution designed to manage and streamline the examination process for the institute. The system is specifically developed to cater to the needs of the ITI, which consists of multiple branches, each with a unique ID, name, and address.

The ITI also consists of various tracks, which are postgraduate specializations where different courses are taught. Each track has a unique ID, name, type, and description. It is important to note that a track must exist in one or more branches, and a branch must contain one or more tracks. The students enrolled in these tracks have unique identification information such as ID, name, gender, status, mobile number, and address, which is comprised of government, city, and street.

In addition, students' educational information such as university, faculty, and graduation year is also recorded in the system. It is important to note that each student must be enrolled in only one track, and a track can contain many students.

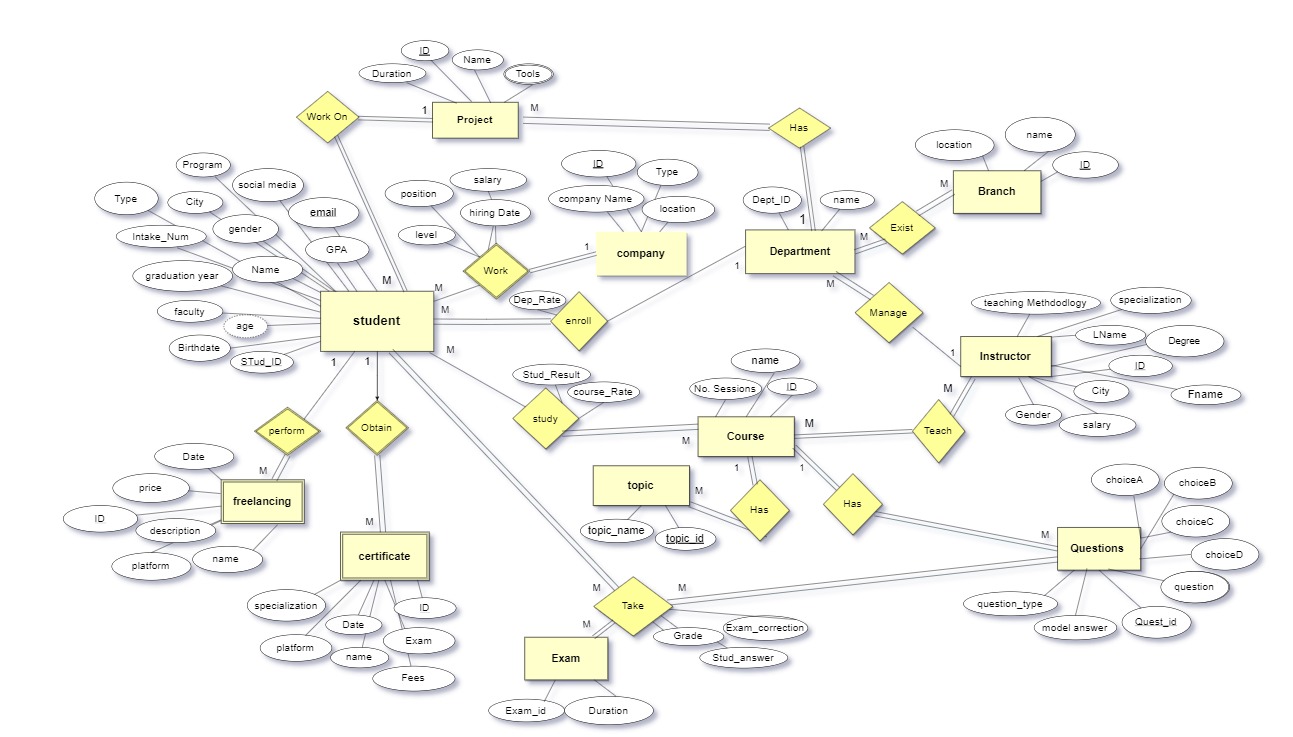
Each track in turn consists of a set of courses, each with a unique ID, name, number of sessions, student results and a rating given by students. It is important to note that a course must exist in one or more tracks, and a track must contain one or more topics. Only topics existing in a student's track will be studied by the student. Instructors are responsible for teaching these courses each have unique identification, specialization, teaching Methodologies and other information as salary ,degree, name and city. It is important to note that each instructor must teach one or more courses, and each course must be taught by one or more instructor.

Each track is managed by only one instructor, and an instructor may manage one track. There are also topics, each with a unique ID and name. A course contains several topics.

In order to graduate, students must accomplish a certain number of certificates, each with a unique ID, name, platform, date, fees, exam and specialization. It is important to note that a student must take more than one certificate, and each certificate can only be accomplished by one student. Furthermore, students must also perform freelancing tasks, consisting of unique ID, platform, type, description, price, date.

At the end of a course, enrolled students will take an exam. The exam consists of a unique ID, date, number of multiple-choice questions (MCQ), and number of true/false questions. The exam is generated by selecting random questions from the question bank, which consists of question ID, question, type, group of answers (A, B, C, D), and the correct answer. It is important to note that for each course, there is a related question bank that contains many questions, and each question belongs to only one course. Each student's answers and grades must be recorded after taking the generated exam.

Entity Relationship Diagram



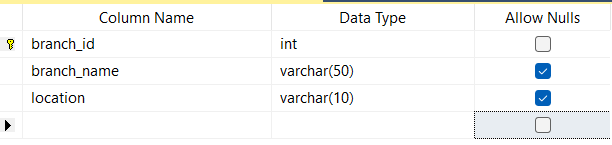
**Database**

**Tables**

Branch table

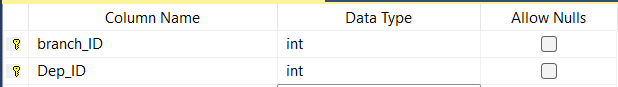
Primary Key - PK: branch\_id.

Foreign key - FK: None in this table.



Branch department table

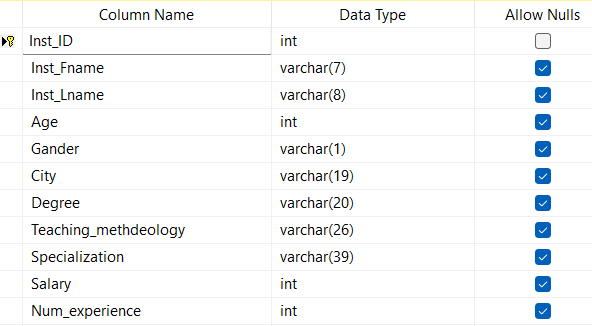
Composite Primary Key - CPK: branch\_Id from (branch table), dep\_id from (department table)



Instructor Table

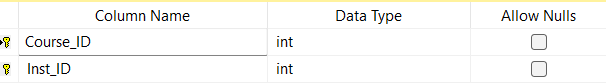
Primary Key - PK: Inst\_id

Foreign key - FK: None in this table.



Instructor course table

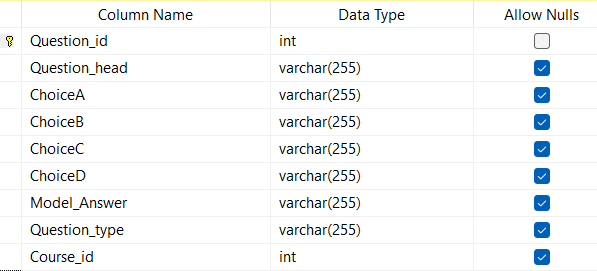
Composite Primary Key - CPK: course\_id from (course table), Inst\_id from (instructor table).



Questions

Primary Key - PK: Question\_id

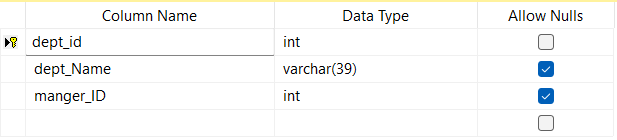
Foreign key - FK: Course\_id from (course table)



Department table

Primary Key - PK: dept\_id

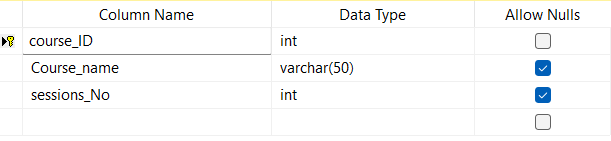
Foreign key - FK: manger\_id from (instructor table)



Courses table

Primary Key - PK: course\_id

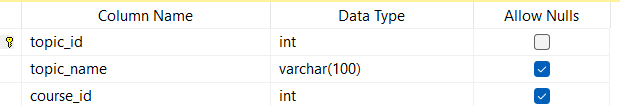
Foreign key - FK: None in this table.



Topic table

Primary Key - PK: topic\_id

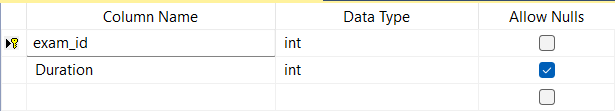
Foreign key - FK: course\_id from (course table)



Exam table

Primary Key - PK: Exam\_id

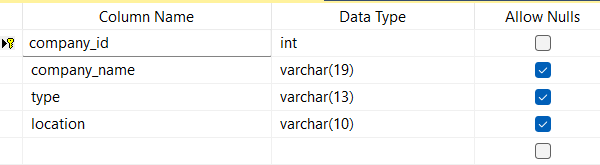
Foreign key - FK: None in this table.



Company table

Primary Key - PK: company\_id

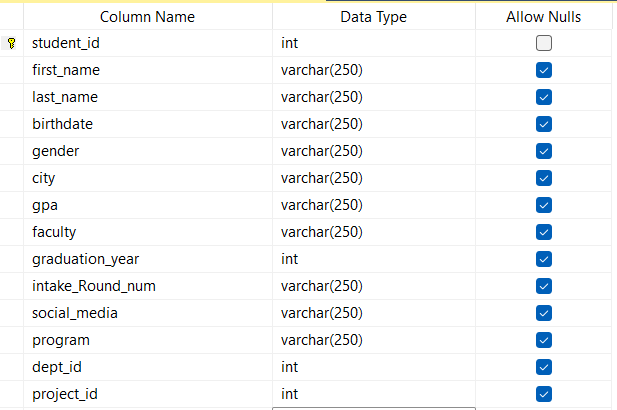
Foreign key - FK: None in this table.



Student table

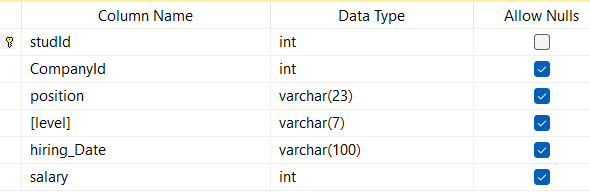
Primary Key - PK: student\_id

Foreign key - FK: dept\_id from(department table), project\_id from(project table)



Student company table

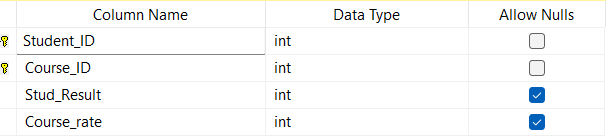
Composite Primary Key - CPK:



Student course table

Composite Primary Key - CPK: student\_id from (student table), course\_id from (course table)

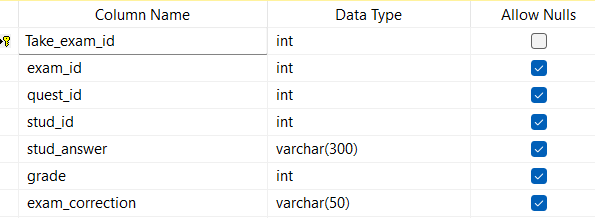
Foreign key - FK: None in this table.



Take exam table

Primary Key - PK: take\_exam\_id

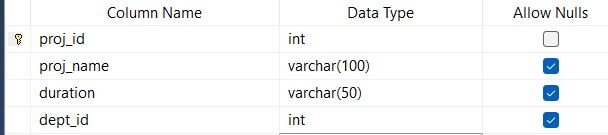
Foreign key - FK: quest\_id from (question table), exam\_id from (exam table), stud\_id from (student table)



Project table

Primary Key - PK: project\_id

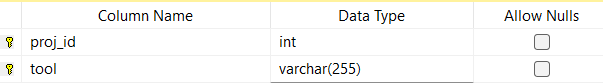
Foreign key - FK: dept\_id from (department table)



Project tools table

Composite Primary Key - CPK:proj\_id from (project table ) and (tool) from (project tool table )

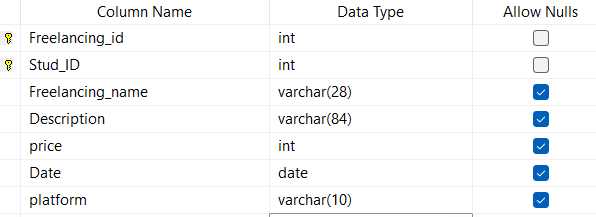
Foreign key - FK: None in this table.



Freelancing table

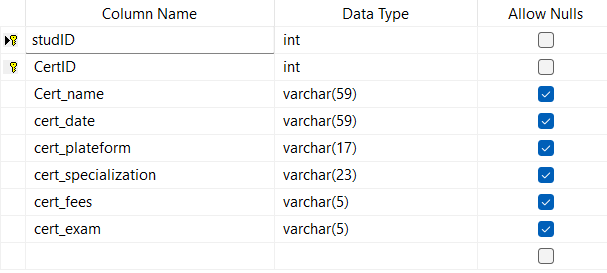
Composite Primary Key - CPK: freelancing\_id from (freelancing table) and (stud\_id) from(student table)

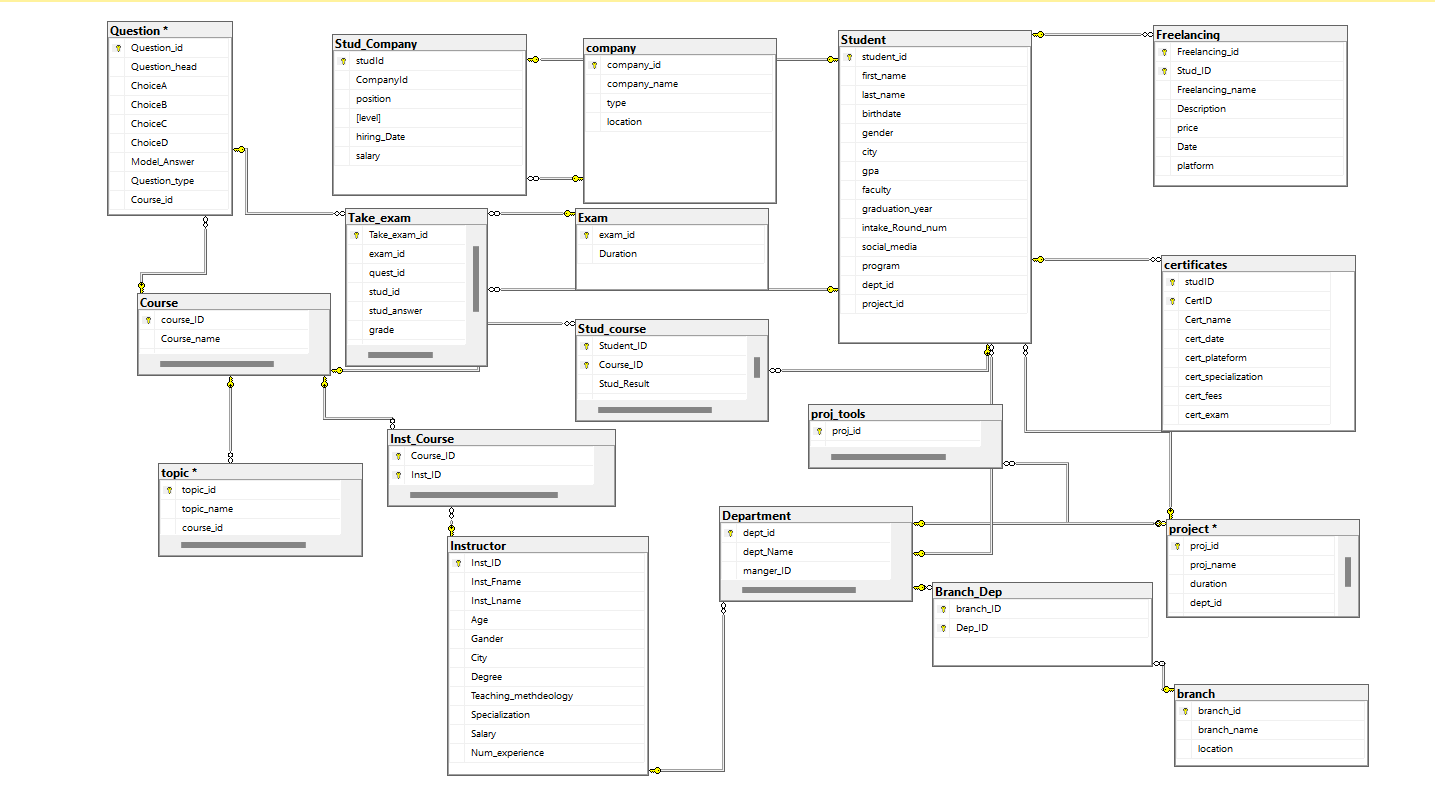
Foreign key - FK: None in this table.



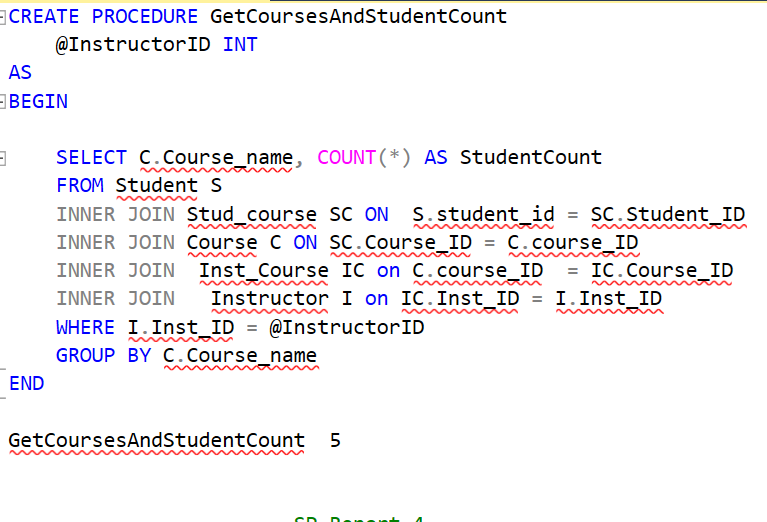
Certificate table

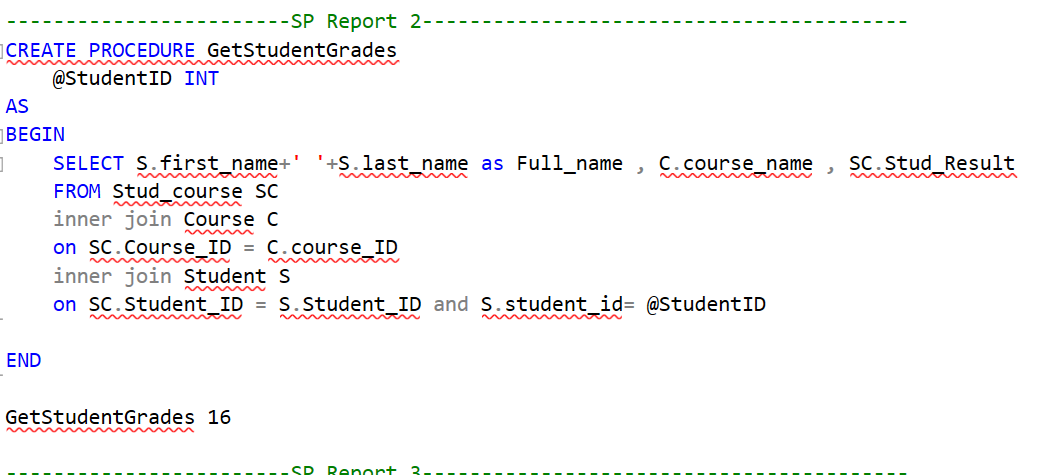
Composite Primary Key - CPK: cert\_Id from (certificate table) and (stud\_id) from(student table)

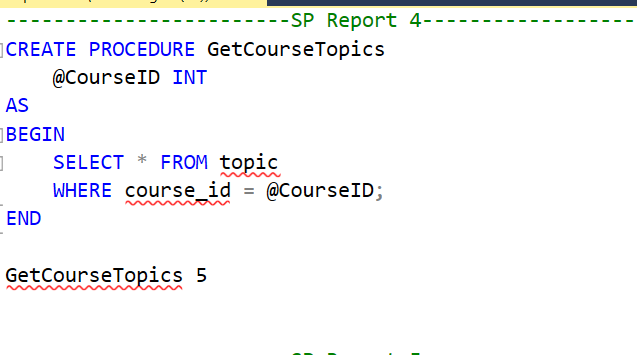
Foreign key - FK: None in this table.

SQL Server Mapping

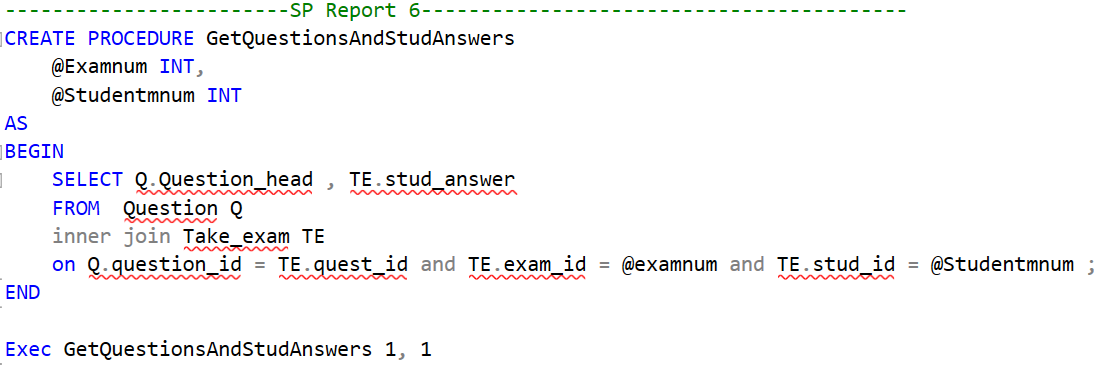
Stored Procedures Reports scripts

* Get courses and student count
* Get student and grades

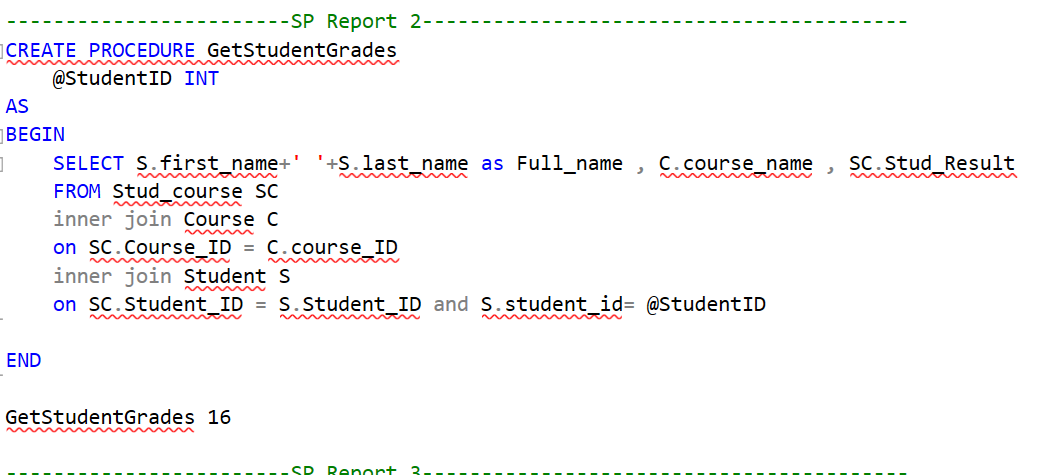




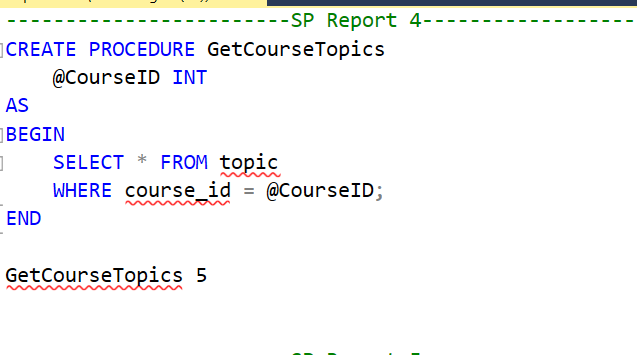
* Get questions and student answers

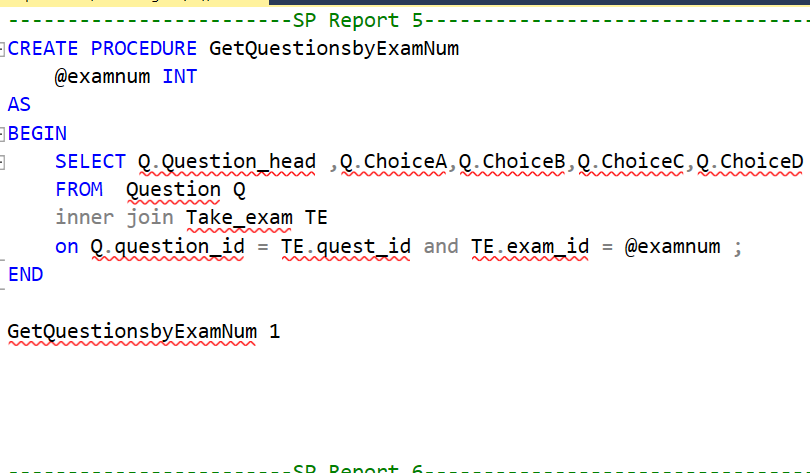


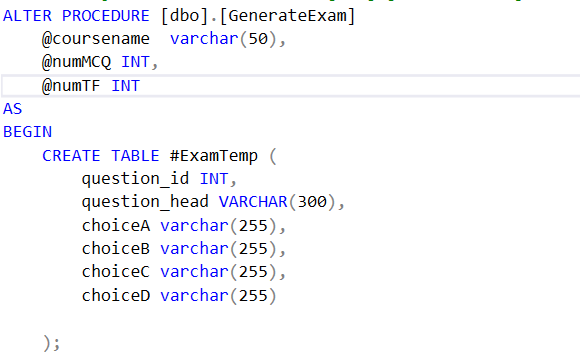
* Get student grades

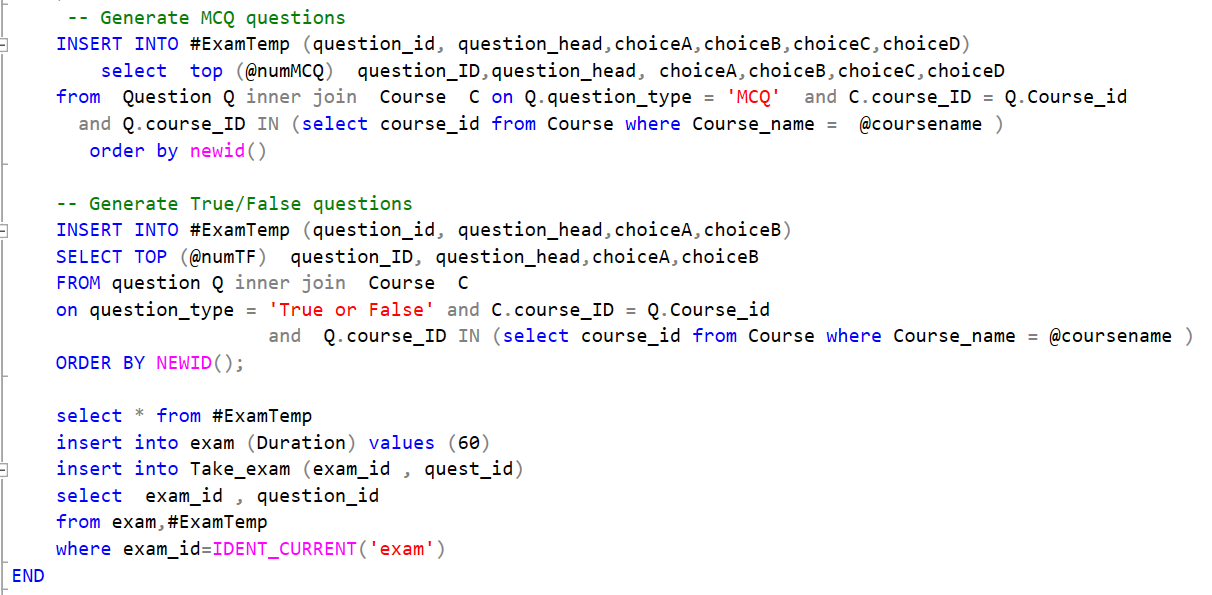


* Get course topics

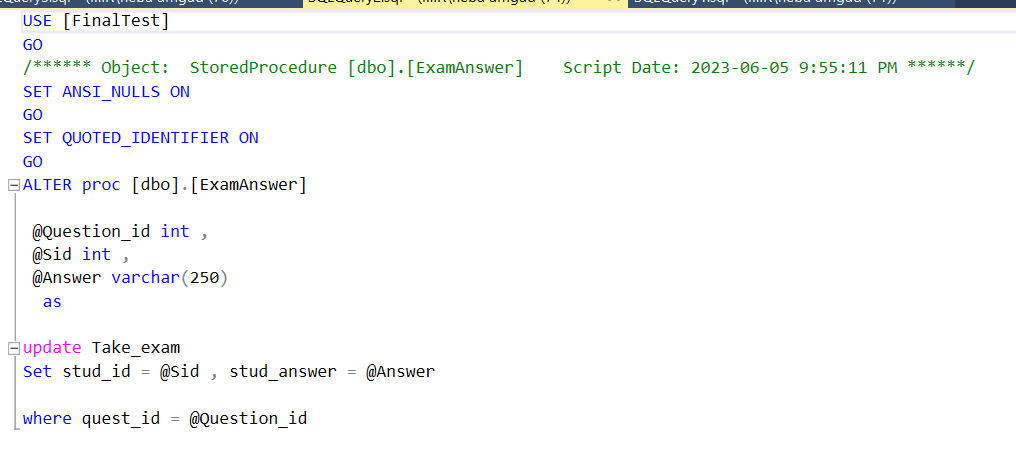


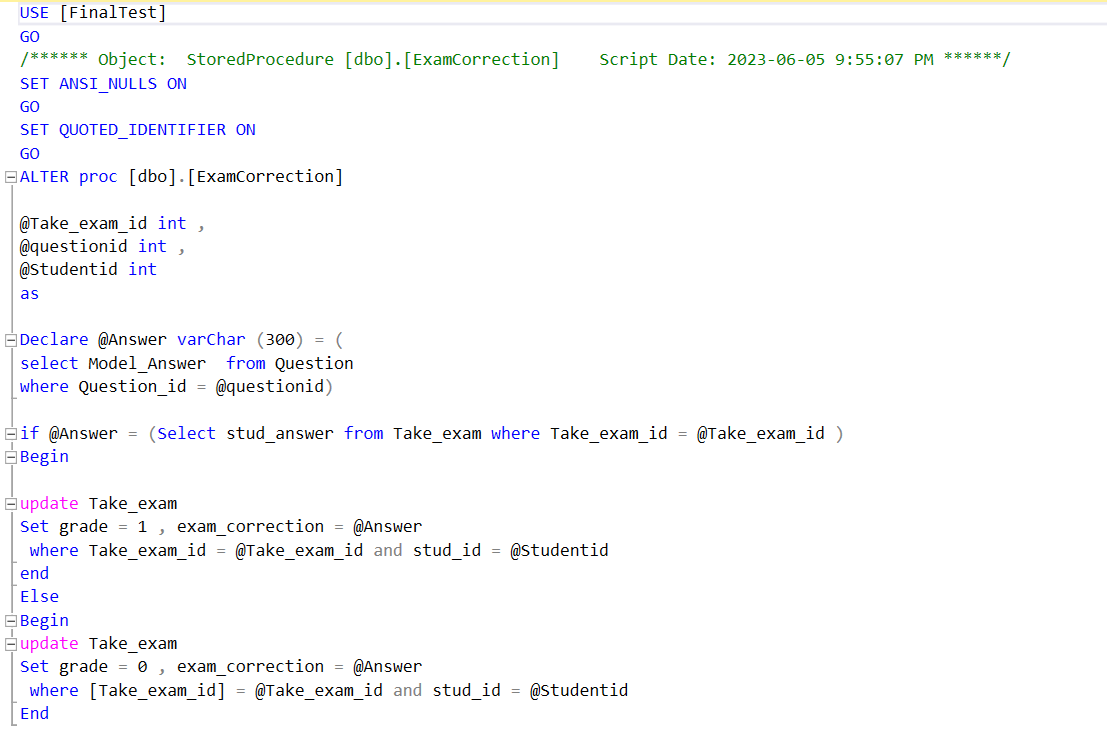
* get question by exam
* Exam Generation



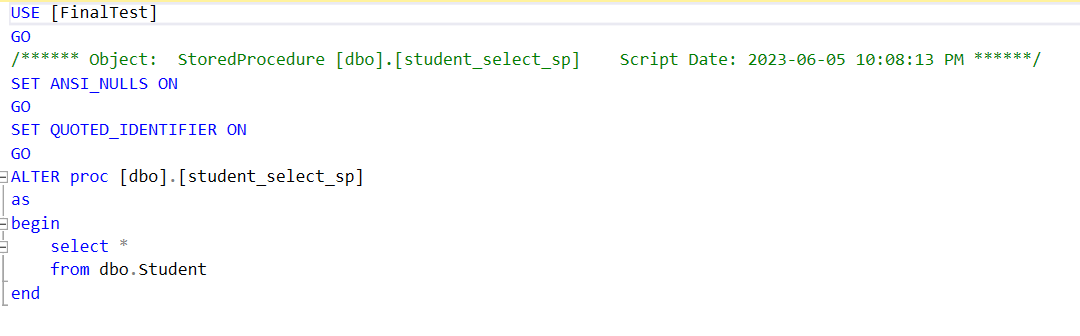


Student Answers

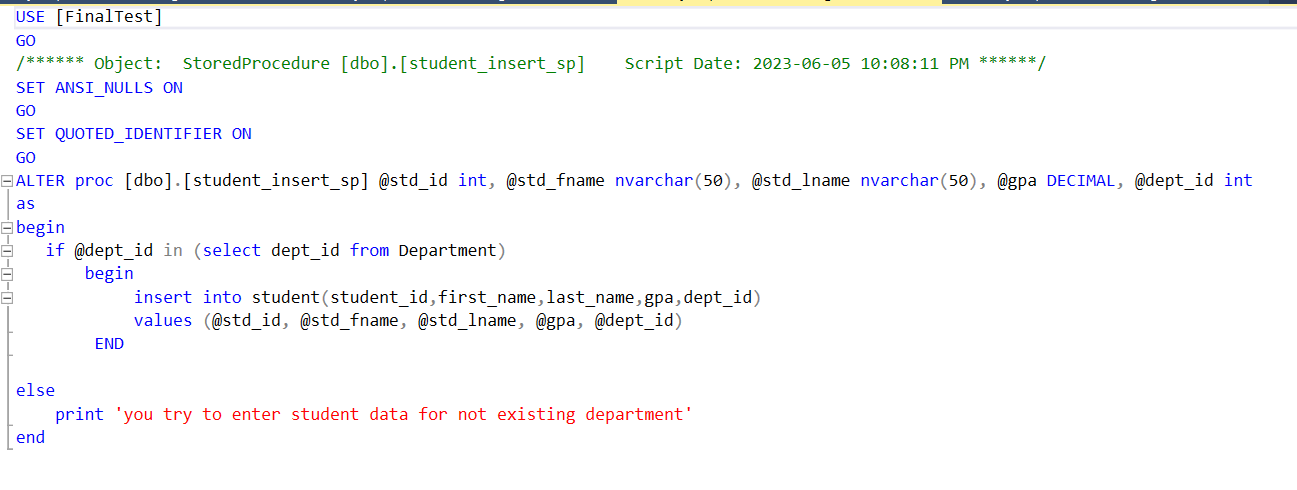
* Student Answers
* Exam correction



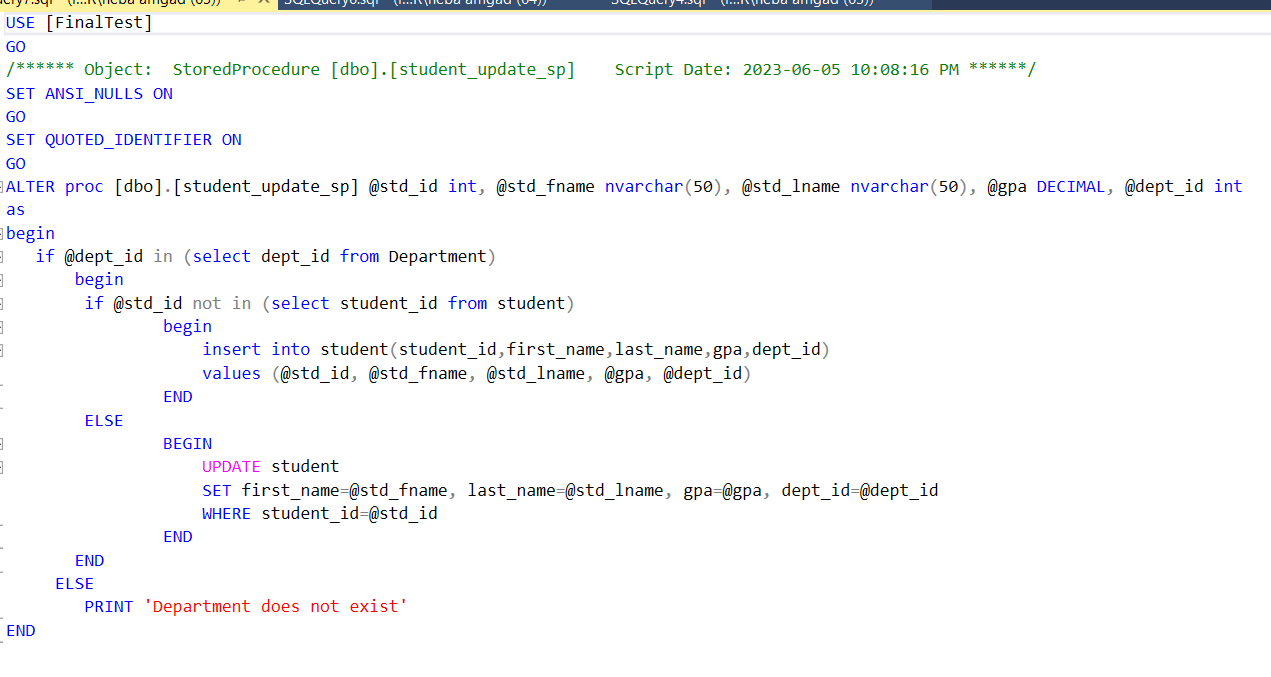
* Student select procedure



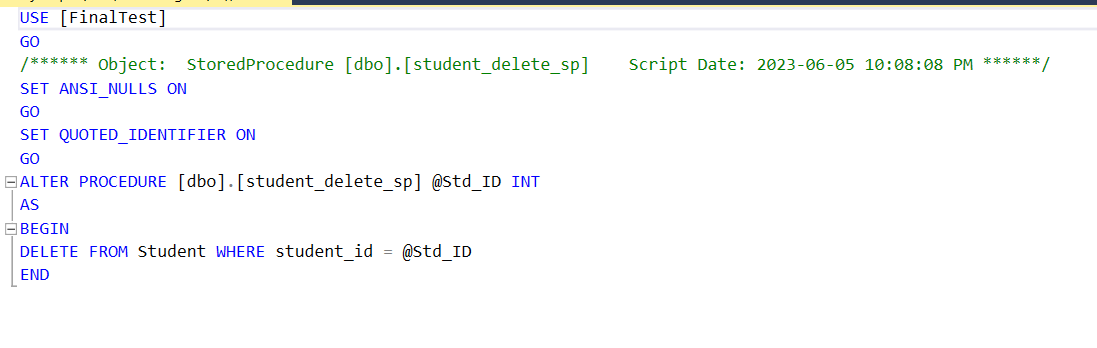
* Student insert procedure



* Student update procedure



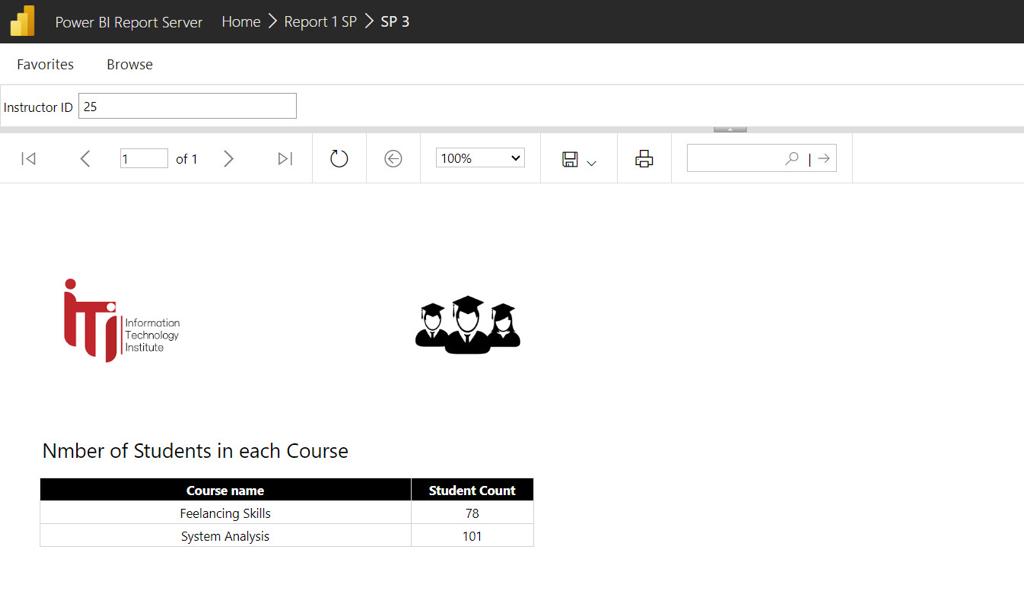
* Student delete procedure



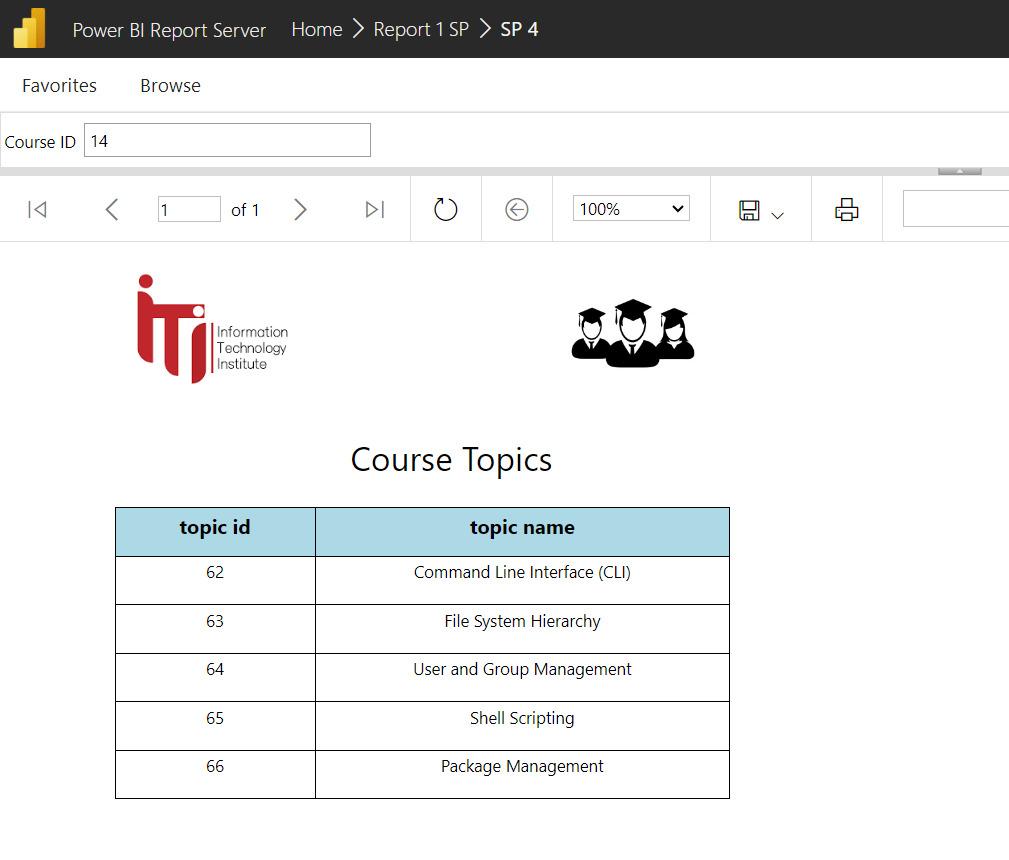
* **Reports Review**

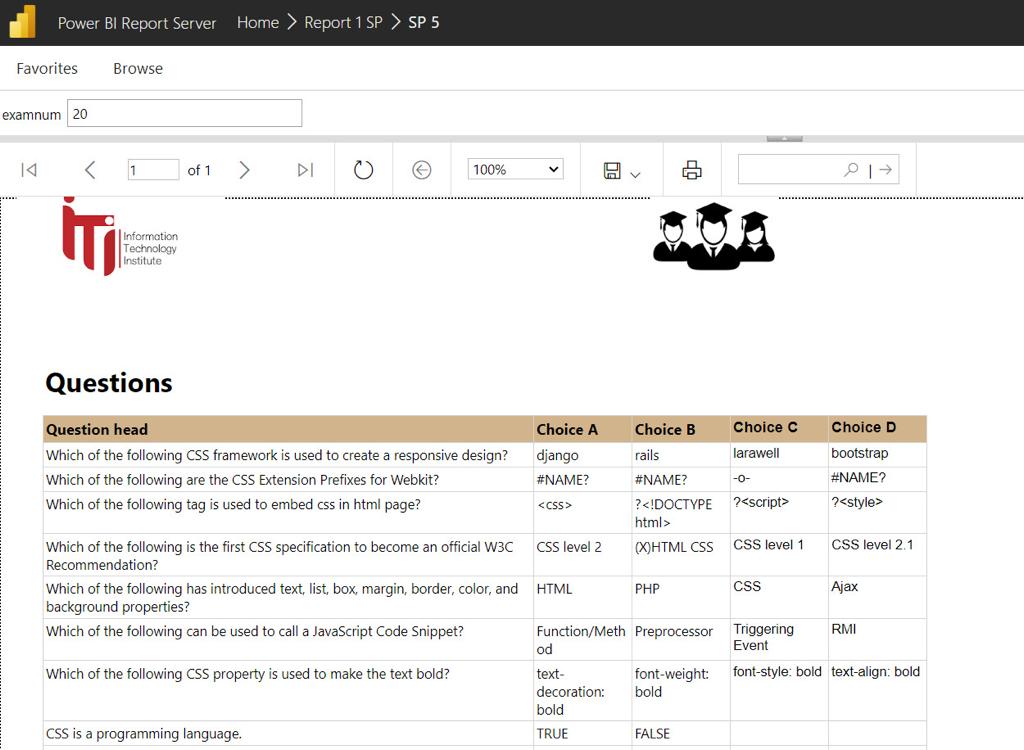
1. Students Grades Report   
   A screenshot of a computer

   Description automatically generated with medium confidence
2. Number Of students in each Course



1. Course Topics report

  
4- Questions Report



1. Students’ information Report

A screenshot of a computer

Description automatically generated with medium confidence

1. Questions and students Answers  
     
   A screenshot of a computer

   Description automatically generated with medium confidence

* Dashboards review

1. **Tableau Dashboards**

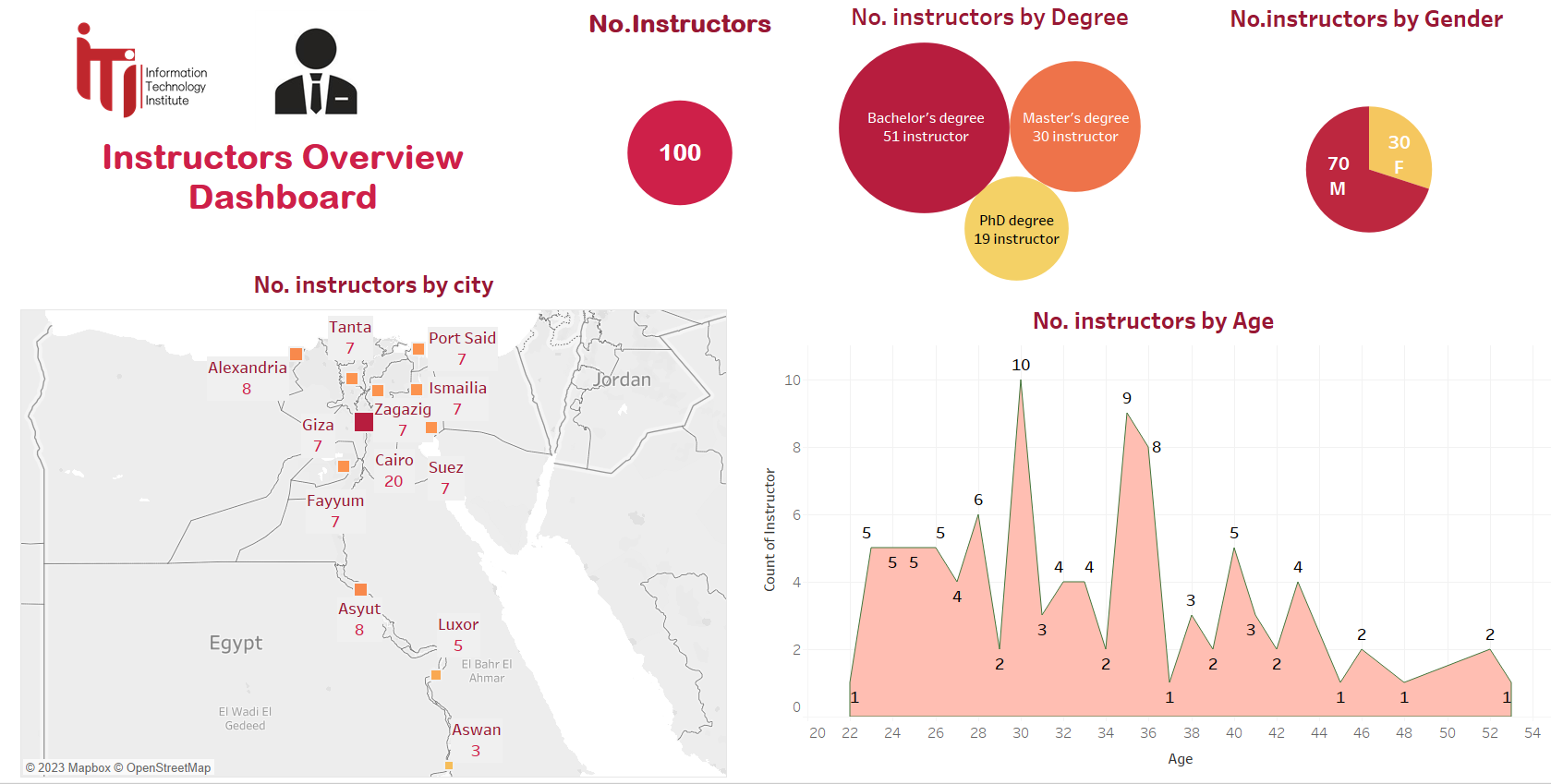
A screenshot of a graph

Description automatically generated with low confidence

A screenshot of a graph

Description automatically generated with low confidence

3-



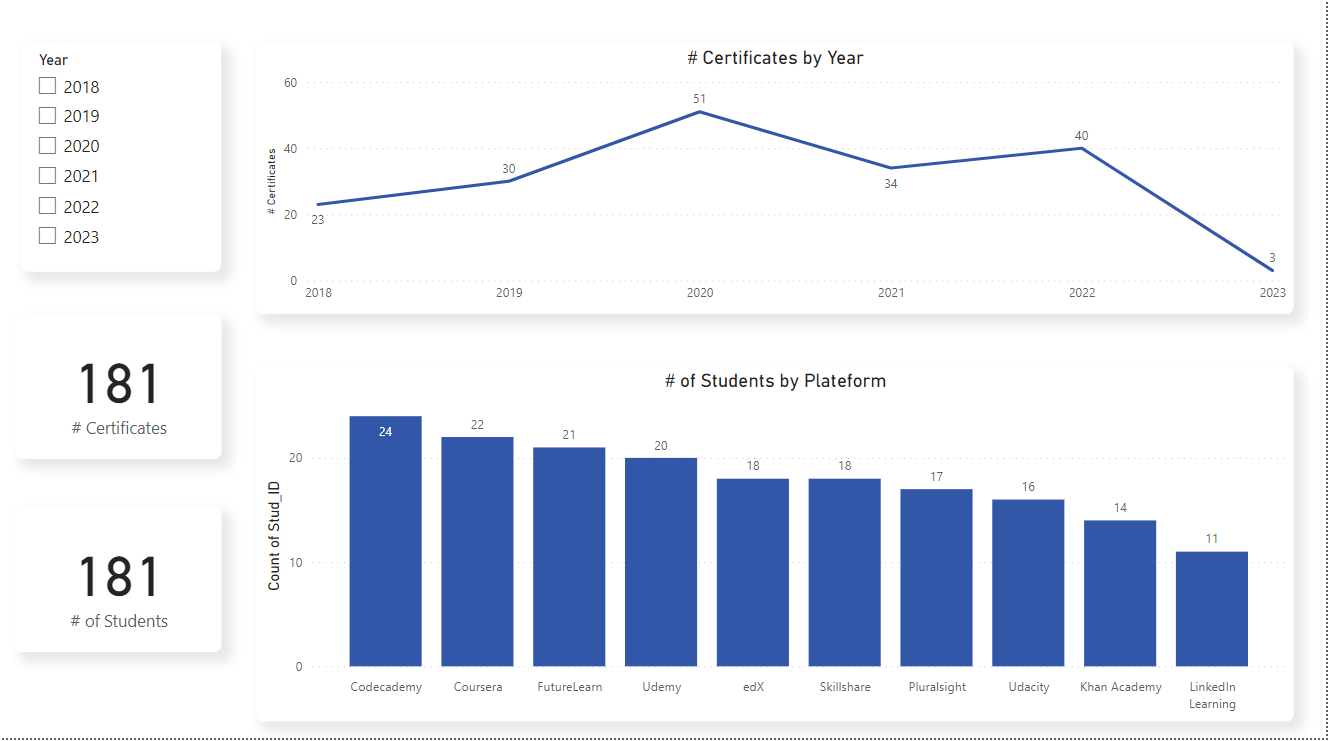
4-  
A close-up of a graph

Description automatically generated with low confidence

1. A screenshot of a graph

   Description automatically generated with low confidence  
     
     
   **(2) Power Bi Dashboards**

1-



2-



3-



4-

