

**ITI Examination System**

**By:**

**Tarek Mohamed Emam**

**Zyead Hassan Eid**

**Hossam Hamdy El-Ghandour**

**Ahmed Anwar**

**Khaled Samy Abdelfattah**

**Content**

1. **Summary**
2. **Conceptual schema**
3. **Mapping**
4. **Physical schema**
5. **Data Generation**
6. **Important SPs for the System**
7. **ETL with Python & loading to Snowflake**
8. **Data warehouse modeling**
9. **Reporting using SSRS**
10. **Insights & visualization using Power BI**
11. **Examination System Website**

# 1. Summary

The ITI Examination System is designed to manage all aspects of student enrollment, coursework, assessment, and graduation at the Information Technology Institute (ITI) across its 21 branches in Egypt. The system supports one program type—the 9-month Professional Training Program (PTP) following its batch schedule. Students enroll in tracks composed of multiple courses, take specific exams, and must fulfill requirements including freelancing engagements, certifications, and a capstone project to graduate. Instructors deliver courses, manage branches; branch managers oversee operations. The application will cover the end-to-end lifecycle from user management through reporting.

# 2. Conceptual Schema

Below is a high-level ER diagram description capturing entities, key attributes, and relationships:

# Entities

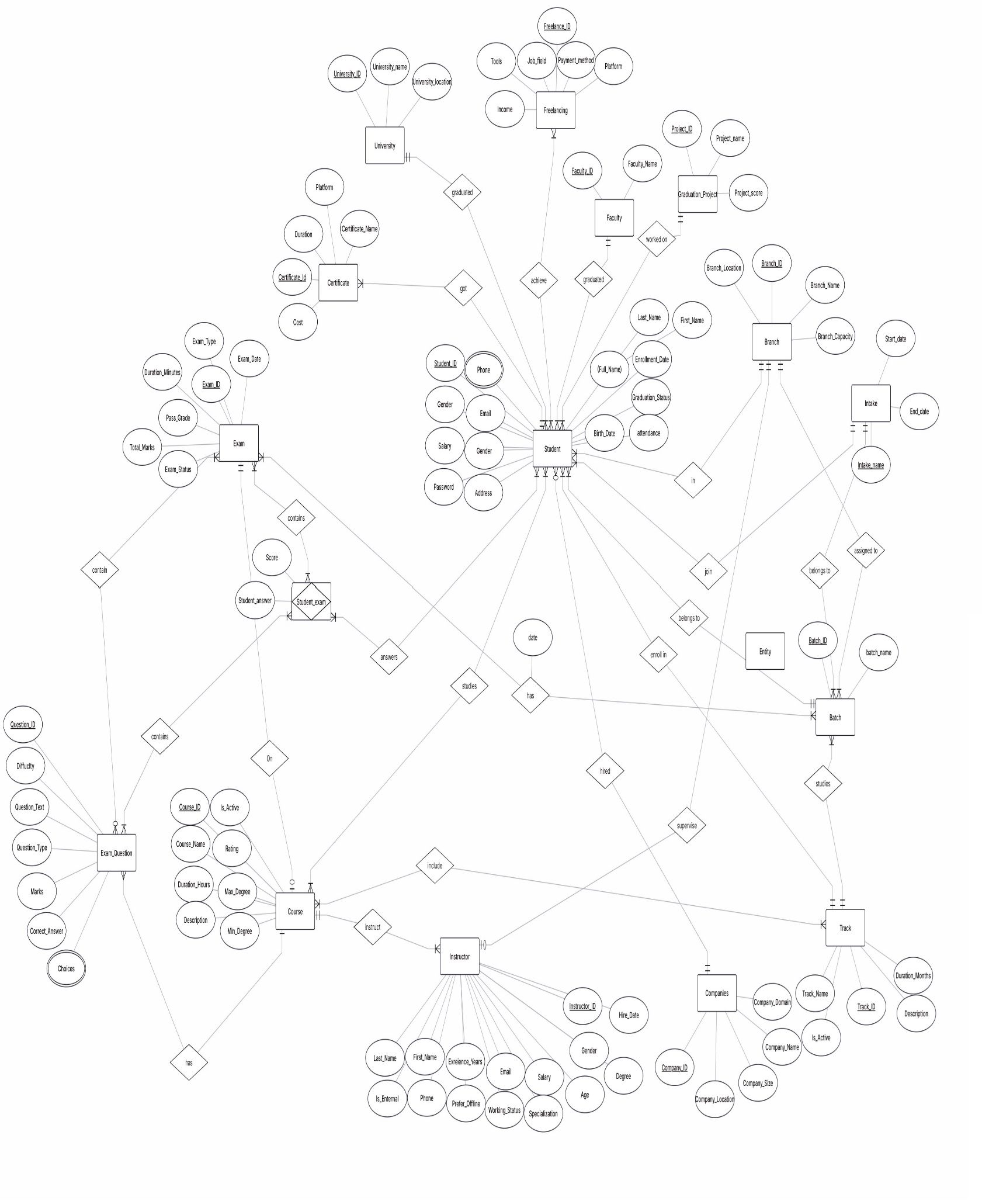
* **Branch** (*Branch\_ID*, Branch\_Name, Branch\_Location, Branch\_Capacity)
* **Batch** (*ID*, Name)
* **Track** (*ID*, Name, Is\_Active, Description, Duration\_Months)
* **Intake** *(ID, Name, Start Date, End Date)*
* **Course** (*ID*, Name, Duration\_Hours, Description, Min\_Degree, Max\_Degree, Rating, Is\_Active)
* **Instructor** (*ID*, FirstName, LastName, Email, Age, Gender, Phone, Address, Degree, Salary, Exreience\_Years, Hire\_Date, Specialization, Working\_Status, Prefer\_Offline, Is\_Enternal)
* **Student** (*ID*, FirstName, LastName, DOB, Enrollment\_Date, Graduation\_Status , Gender, Phone, Address, Email, Salary, attendance, Password)
* **Company** (*ID*, Name, Size, Location, Company\_Domain)
* **Freelance** (*ID*, Field, Platform, Income, Payment\_method, tools)
* **Certificate** (*ID*, Name, Duration, Platform, Cost,)

## •GraduationProject (*ID*, Name, Score)

* **Exam** (*ID*, Type, Date, Status, Total Marks, Pass Grade, Duration Minutes )
* **Question** (*ID*, Text, Type, Level, Mark, correct answer, choices)
* **Faculty** (ID, Name)
* **University** (ID, Name, Location)

# Key Relationships

1. **Supervise**: Manager ↔ Branch (1:1)
2. **Assigned to**: Batch ↔ Branch (1:M)
3. **In**: Branch ↔ Student (1:M)
4. **Belongs to**: Intake ↔ Batch (1:M)
5. **Belongs to**: Batch ↔ Student (1:M)
6. **Studies**: Batch ↔ Track (1:M)
7. **Has**: Batch ↔ Exam (M:M)
8. **Join**: Intake ↔ Student (1:M)
9. **Enroll In**: Track ↔ Student (1:M)
10. **Include**: Track ↔ Courses (M:M)
11. **Hired**: Company ↔ Student (1:M)
12. **Graduated From**: Faculty ↔ Student (1:M)
13. **Graduated From**: University ↔ Student (1:M)
14. **Achieve**: Freelancing ↔ Student (M:M)
15. **Worked On**: Graduation Project ↔ Student (1:M)
16. **Got**: Student ↔ Certificate (1:M)
17. **Studies**: Course ↔ Student (M:M)
18. **Student Exam**: Exam ↔ Student ↔ Question (M:M:M) (Ternary relationship)
19. **Lead**: Student ↔ Student (Unary 1:M)
20. **On**: Course ↔ Exam (1:M)
21. **Contain**: Exam ↔ Questions (1:M)
22. **Instruct**: Instructor ↔ Course (1:M)
23. **Has**: Course ↔ Question (1:M)

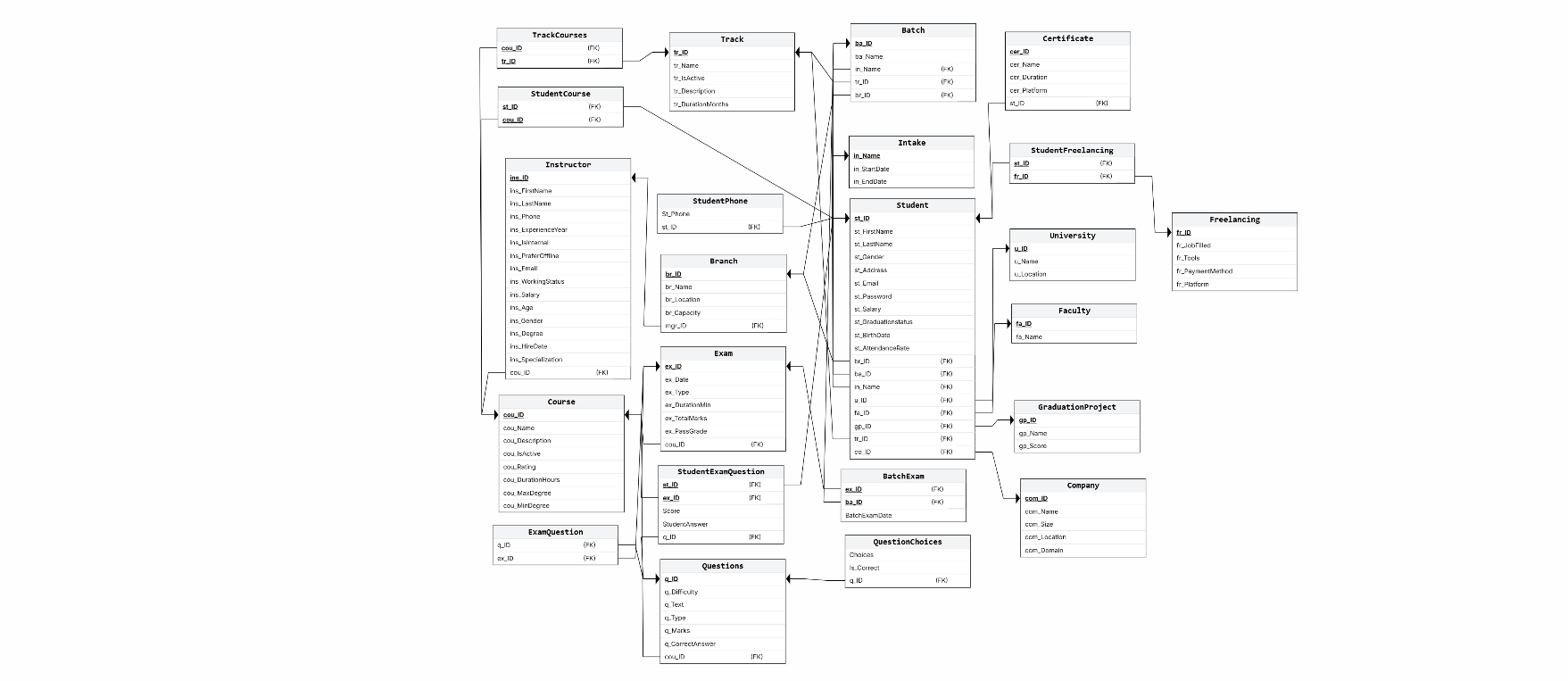
**The Entity Relationship Diagram:**

# 3. Mapping

This section maps conceptual entities and relationships to relational tables, primary keys (PK), foreign keys (FK), and associative tables for M:N relationships.

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Columns** | **PK** | **FK** |
| **Intake** | IN\_Name,Start Date End Date | IN\_Name |  |
| **Branch** | BranchID, Name, Location, Capacity, Mgr\_ID | BranchID | Mgr\_ID → Instructor |
| **Batch Exam** | Ex\_ID, Batch\_ID, Exam\_Date | (EX\_ID,Ba\_ID) |  |
| **University** | U\_ID, Name, Location | U\_ID |  |
| **Faculty** | Fa\_ID, Name | Fa\_ID |  |
| **Batch** | BatchID, Name, In\_Name, Tr\_ID, Br\_ID | BatchID | In\_Name →Intake name  Tr\_ID → Track ID  Br\_ID → Branch ID |
| **Student** | StudentID, FirstName,  LastName, DOB, Gender, Address, Salary, Fa\_ID, Attendance Rate, Graduation Status, Email, Password, Salary, BranchID,  BatchID, LeaderID, Intake\_ID, Track ID, U\_ID, GP\_ID, Co\_ID | StudentID | BranchID → Branch  BatchID → Batch  LeaderID → Student  Intake\_ID → Intake  Track\_ID → Track  U\_ID → University  Fa\_ID → Faculty  GP\_ID → Graduation Project  Co\_ID → Company |
| **Student Phone** | St\_ID, St\_Phone | (St\_ID, St\_Phone) | St\_ID |
| **Student Courses** | St\_ID, Cou\_ID | (St\_ID, Cou\_ID) | St\_ID → Student, Cou\_ID → Course |
| **Instructor** | InstructorID, FirstName, LastName, Email, DOB, Gender, Phone, Address, Degree, Salary, experience years, Is internal, Prefer Offline, working status, age, hire date, specialization | InstructorID |  |
| **Track** | TrackID , Name, Description, Is\_Active, Duration | TrackID |  |
| **Track Courses** | Tr\_ID, Cou\_ID | (Tr\_ID, Cou\_ID) | Tr\_ID → Track  , Cou\_ID → Course |
| **Company** | CompanyID, Name, Size, Location, Domain | CompanyID |  |
| **Freelance** | FreelanceID, Field, Platform, Income, Tools, Payment Method | FreelanceID |  |
| **Student Freelancing** | St\_ID, Fr\_ID | (St\_ID, Fr\_ID) | St\_ID → Student  Fr\_ID → Freelancing |
| **Certificate** | CertificateID, Name, Duration, Platform, Cost, St\_ID | CertificateID | St\_ID → Student |
| **GraduationProject** | ProjectID, Name, Score | ProjectID |  |
| **Course** | CourseID, Name, Description, Is\_Active, Rating, Duration, Max Degree, Minimum Degree, Ins\_ID | CourseID | Ins\_ID → Instructor |
| **Question** | QuestionID, Text, Type, Difficulty, Mark, Correct Answer, Cou\_ID | QuestionID | Cou\_ID → course |
| **Question Choices** | Choice , QuestionID, IsCorrect | (Choice,QuestionID) | QuestionID → Question |
| **Exam** | ExamID, Name, Type, Duration Minutes, Total Marks, Pass Grade, cou\_ID | ExamID | Cou\_ID → course |
| **ExamQuestion** | ExamID, QuestionID | (ExamID,  QuestionID) | →ExamID → Exam  , QuestionID → Question |
| **StudentExamQuestion** | StudentID, ExamID, QuestionID, Score, Student Answer | (StudentID,  ExamID,  QuestionID) | StudentID → Student,  ExamID → Exam,  QuestionID → Question |

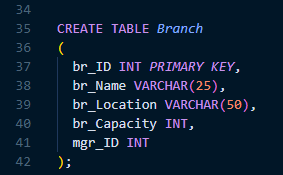
**The Mapping Diagram:**



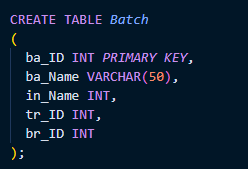
# 4. Physical Schema

Below are the SQL Data Definition Language (DDL) statements for implementing the database schema:

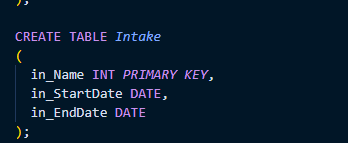
* **Branch**: Stores information about ITI branches and their associated Supervisor details.



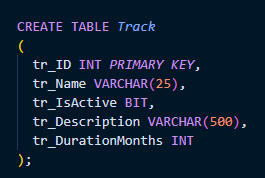
* **Batch**: Represents Batches cycles across intakes and branches and tracks.



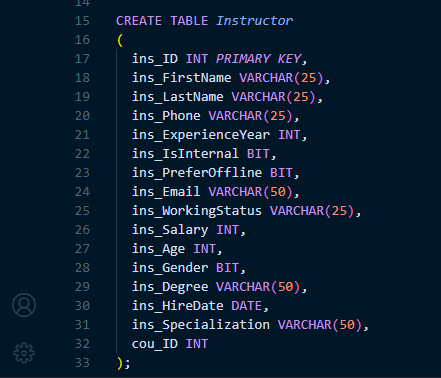
* **Intake**: Represents 31 intakes across ITI history.



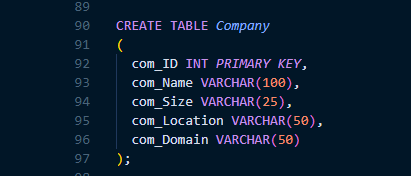
* **Track**: Represents different Tracks ITI provides with its description and activity.



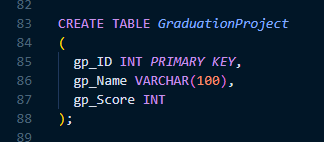
**Instructor**: Stores instructor details, qualifications, contact info, and salary.



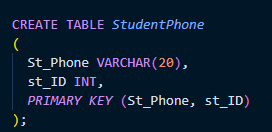
* **Company**: Holds company data involved in hiring students.

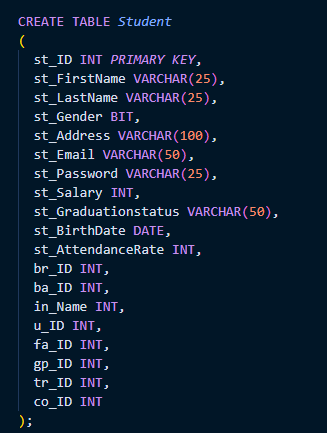


* **Graduation Project**: Details about graduation projects students do as a KPI.

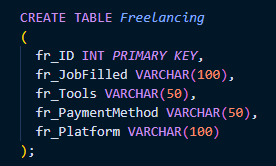


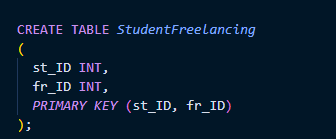
**Student Phone :** contains student phones.

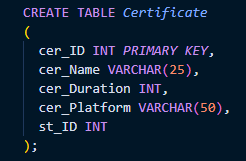
****

* **Student**: Central table for students, linking to tracks, branches, batches, and graduation projects.

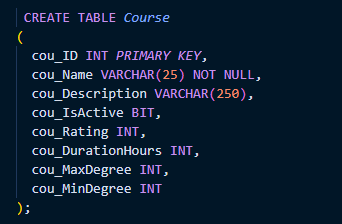
* **Freelance**: Contains data about freelancing work students undertake.



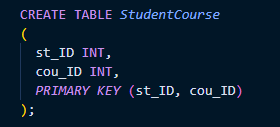
* **Student\_Freelance**: Associates students with their freelance work.
* **Certificate**: Information on certificates earned by students with platform.



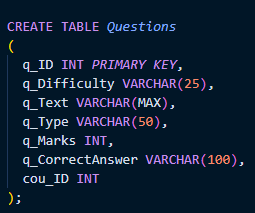
* **Course**: Describes courses with duration and degrees.

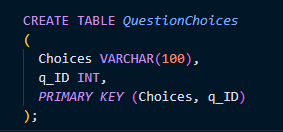
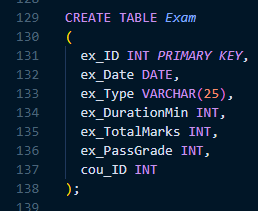


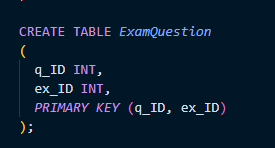
* **Student\_Course**: Tracks course enrollments of students.



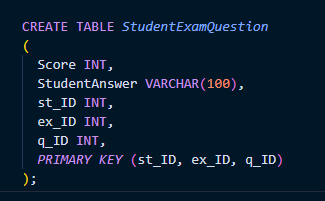
* **Question**: Questions for exams, categorized by type and difficulty.



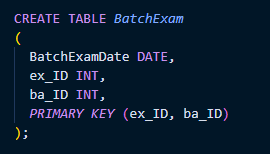
* **Choices**: Contains the multiple choices for each question.
* **Exam**: Represents exam events with time and location details.
* 
* **Exam\_Question**: Maps questions to the exams they appear in.



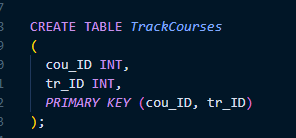
* **Student\_Exam\_Question**: Captures student answers and points per question.

****

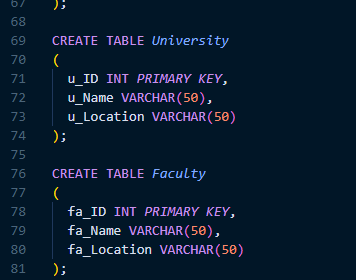
* **Batch Exam :** records exams for each Batch.



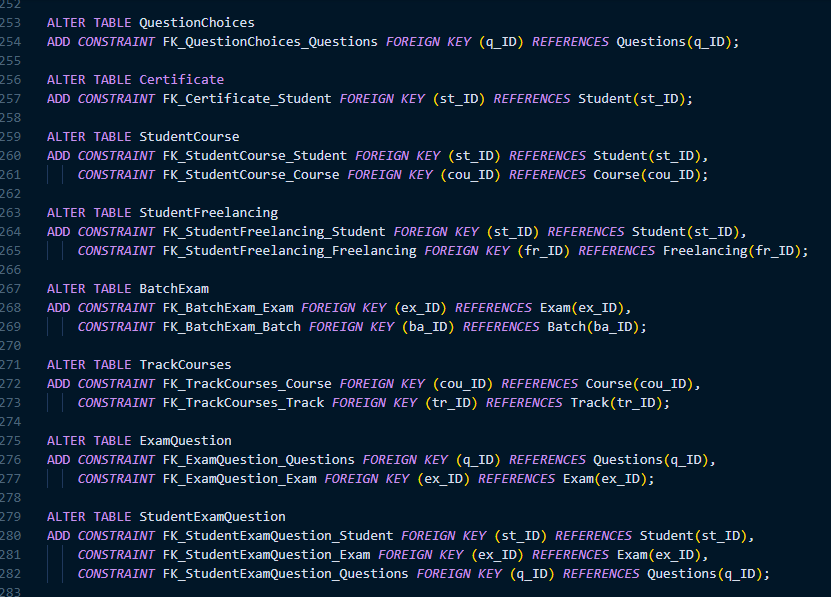
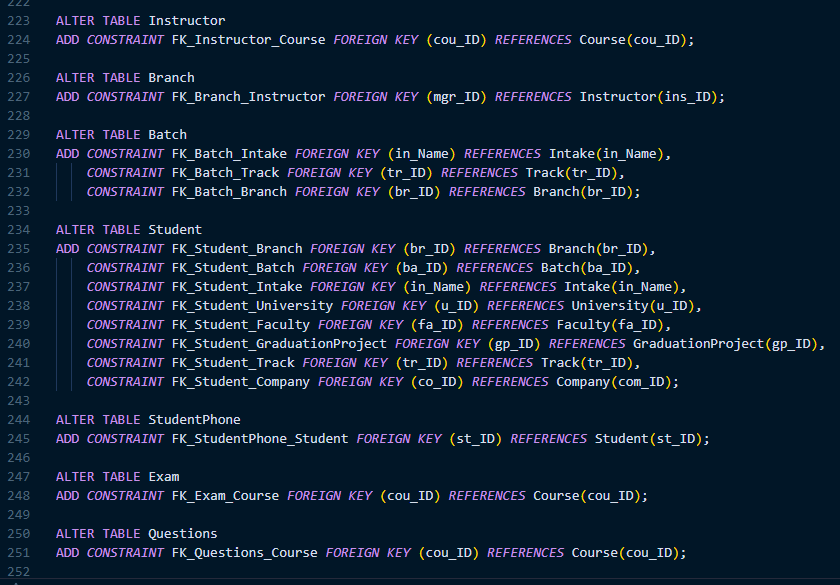
* **Track Courses:** records courses included in each track.

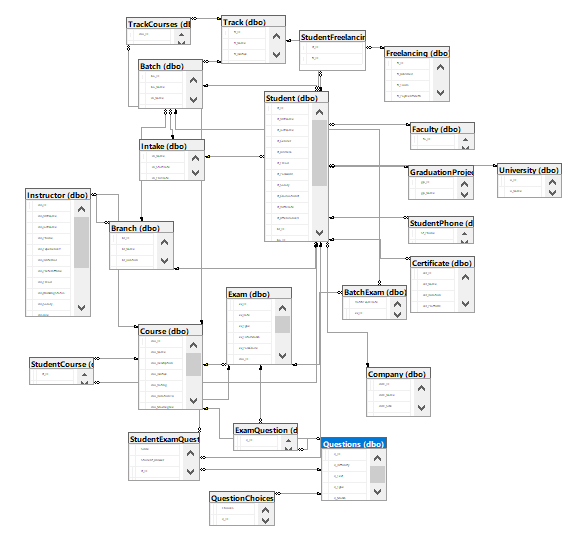
****

* **Faculty and university Tables: represents the education our students have.**

****

**Define Tables constraints (PK,FK)**

****

**The Diagram in SQL Server Management System (SSMS):**

# 5. Data Generation

Generated synthesized data using AI tools like Claude and Gemini, We first generated tables that do not have FK in it then generated tables with FK references to other tables PK with efficient linking and got the final data as CSVs. Then used Python script to import data into Data Base.



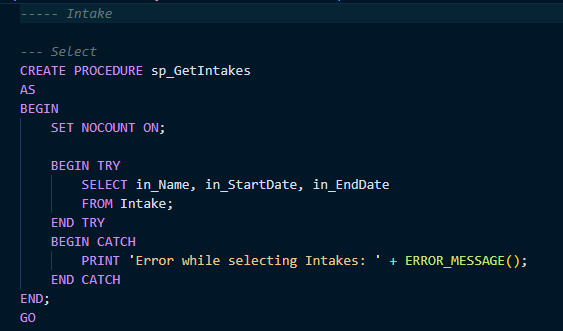
# 6. Important SPs for the System

This chapter includes key stored procedures that automate and facilitate common database operations in the system.

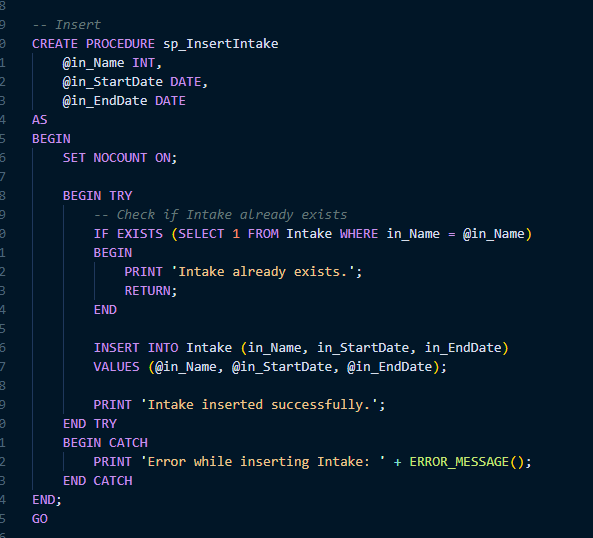
Additionally, full CRUD (Create, Read, Update, Delete) operations have been implemented for every single table in the system. However, for brevity, only the CRUD procedures for the Intake table are documented below as a representative example.

* **Intake CRUD Operations:** To demonstrate standardized database operations, the following procedures are implemented only for the Intake table.

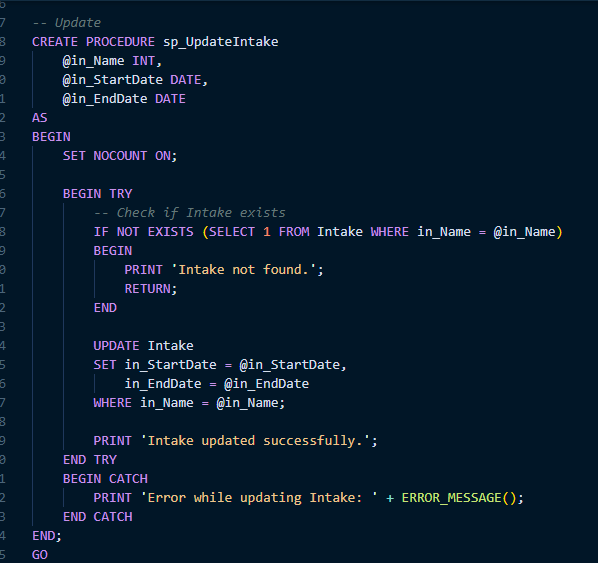
* 1. Select a Intake by ID



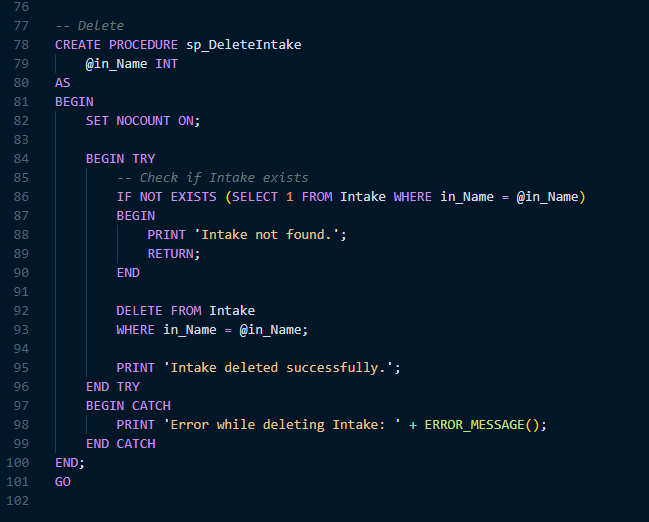
* 1. Insert a new Intake



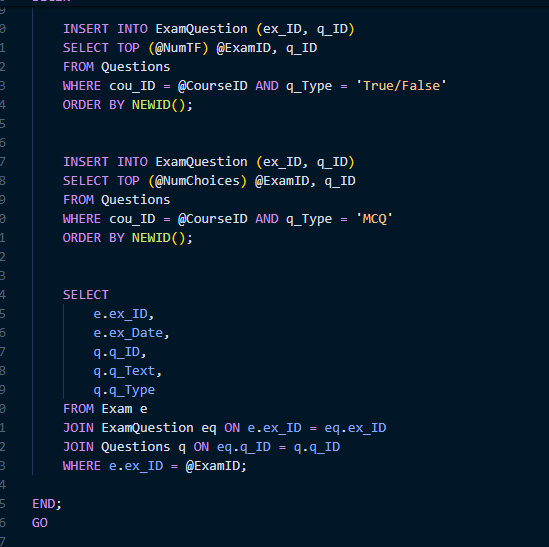
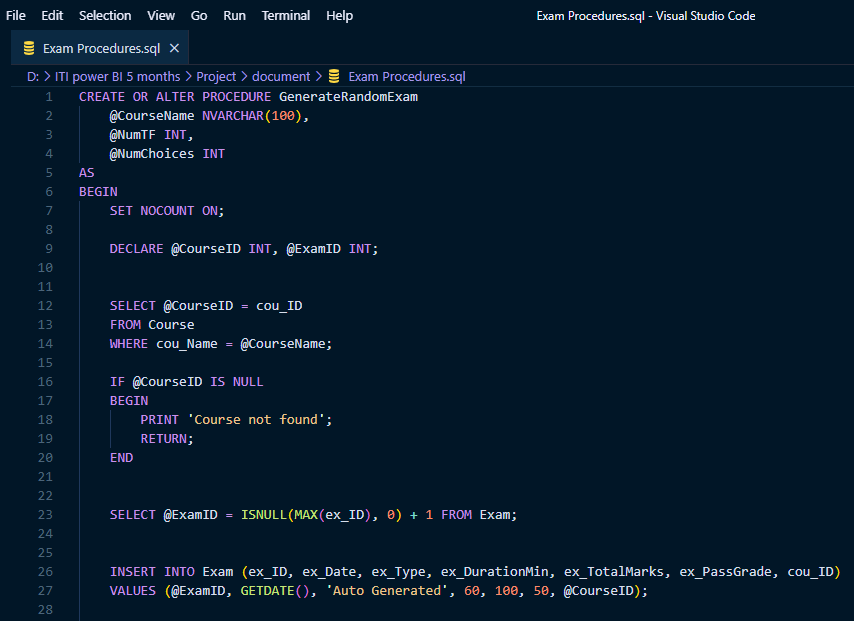
* 1. Update an existing Intake



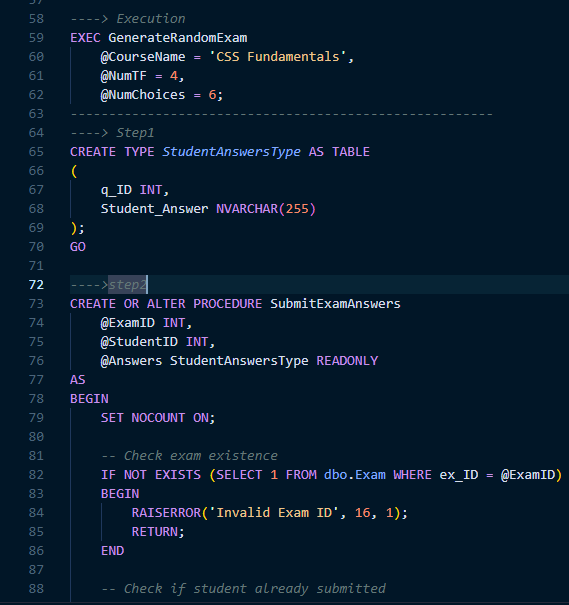
* 1. Delete a Intake by ID

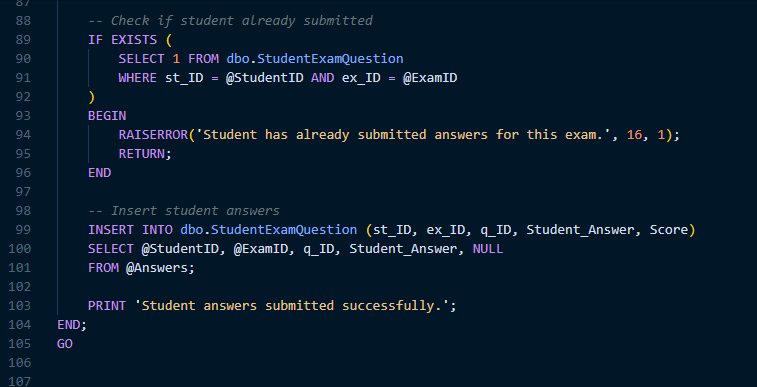


* **Generate Exam:** Automatically creates an exam by randomly selecting a specified number of MCQ and True/False questions from a course. It calculates the total exam mark, stores the exam, and maps it to selected questions



* **ExamAnswer:** Records a student's answer to a specific exam question. It compares the submitted answer with the correct one and awards the appropriate points.



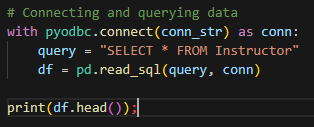


**7. ETL with Python & loading to Snowflake:**

**7.1- Initial Connection and Data Test**

This serves as an **initial connection test and proof-of-concept for extraction** from the Azure SQL Database. It imports necessary libraries, defines the secure connection parameters (server, database, username, password, and driver), and constructs the connection string. Using the pyodbc connection object within a with statement ensures proper resource management. It then executes a specific query (SELECT \* FROM Instructor), reads the result directly into a **Pandas DataFrame**, and prints the first few rows (df.head()) to immediately verify successful connectivity, authentication, and data retrieval from the source system.





**7.2- Dynamic Table List Retrieval**

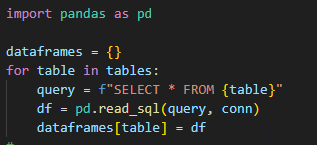
This section focuses on **establishing connectivity and listing all available tables** in the source database. It establishes a direct pyodbc connection using a simplified connection string. It then creates a **cursor** object to execute a standard SQL query against the INFORMATION\_SCHEMA.TABLES view. This query specifically filters for BASE TABLE types, and the results are fetched and stored in a list named tables, intended to hold the names of all source tables for subsequent processing.

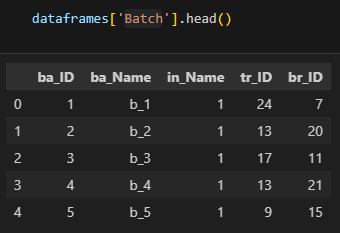
A computer code on a black background

AI-generated content may be incorrect.

**7.3 Reading All Source Data into Memory (DataFrames Dictionary)**

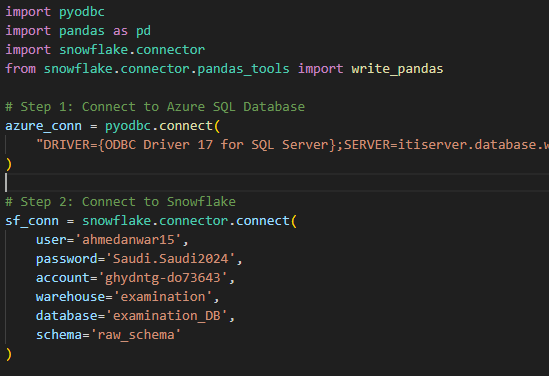
This block iterates through the list of table names retrieved in the previous section (tables) and **extracts the complete content of every source table into memory**. It initializes an empty dictionary, dataframes. For each table name, it constructs a generic SELECT \* query, executes it using the existing database connection (conn), and reads the entire result set into a new **Pandas DataFrame**. Finally, it stores each DataFrame in the dataframes dictionary, keyed by the table name, effectively holding all source data in a collection of Pandas objects.

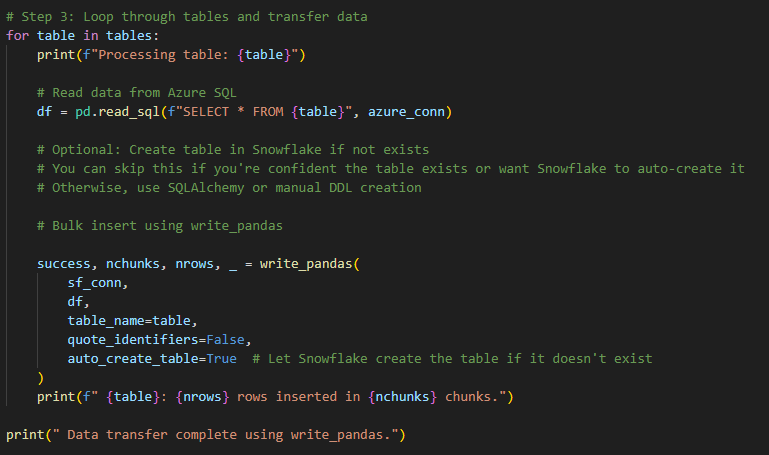


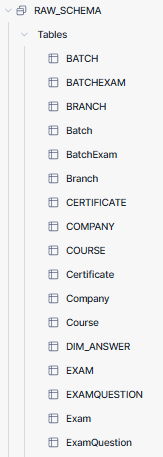


**7.4- ETL - Extract and Load (Azure SQL to Snowflake raw\_schema)**

This is the core **Extract and Load (E/L)** block, moving data from the Azure SQL source to the Snowflake staging area. It establishes separate connections for the **Azure SQL Source (azure\_conn)** and the **Snowflake Destination (sf\_conn)**. A hardcoded list of 23 source table names is defined. The script then iterates through this list, reading the full content of each table into a Pandas DataFrame. The crucial step is the use of the highly efficient **write\_pandas** function, which bulk loads the DataFrame into the corresponding table in the Snowflake examination\_DB.raw\_schema, utilizing auto\_create\_table=True to handle schema creation dynamically. The process prints status updates, including the number of rows inserted, for each table transferred.

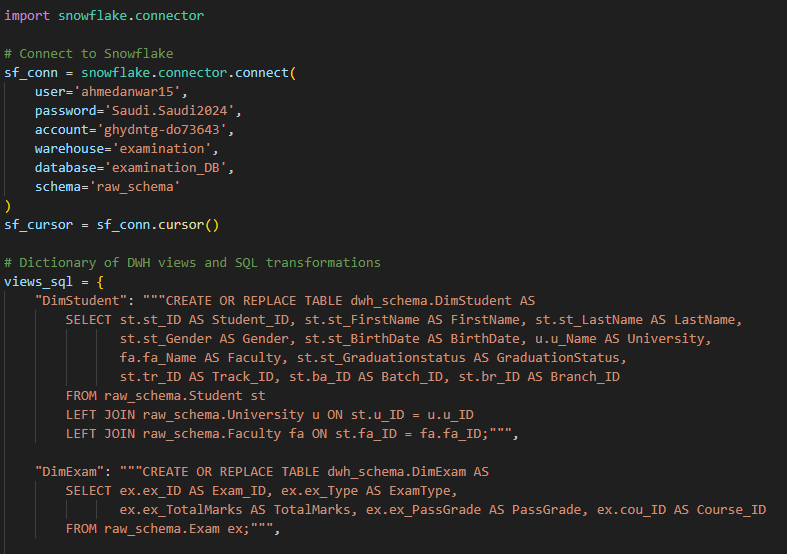


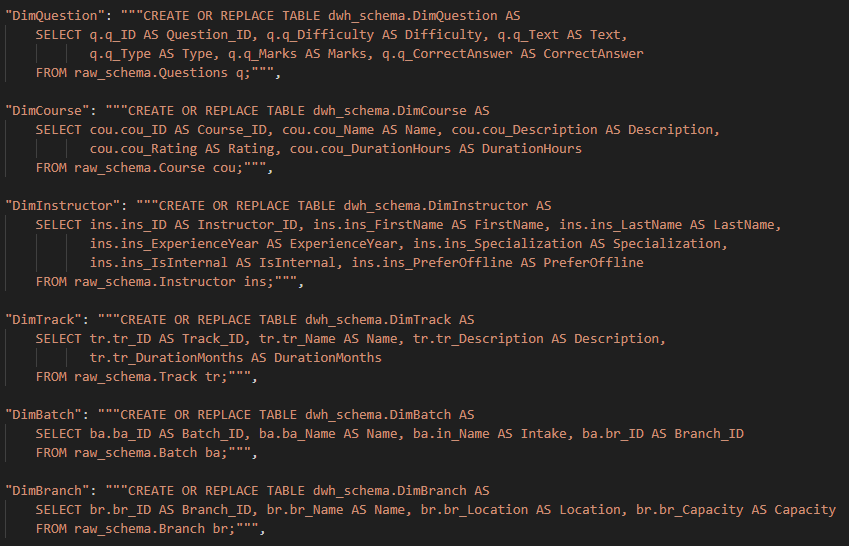


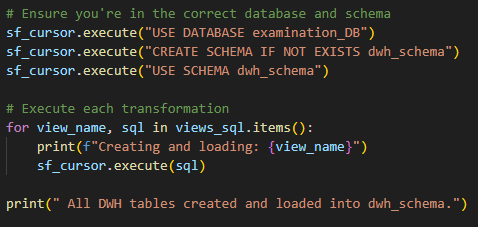


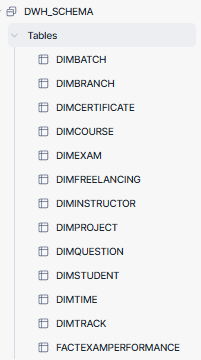
**7.5- ETL - Transformation and Load (Snowflake raw\_schema to dwh\_schema)**

This is the **Transformation (T)** and final **Load (L)** block, building the Dimensional Model within Snowflake. It establishes a connection to Snowflake and defines a comprehensive Python dictionary, **views\_sql**, where keys are the target DWH table names (Dimensions like DimStudent and the Fact table FactExamPerformance) and values are the full SQL CREATE OR REPLACE TABLE AS SELECT ... statements. It ensures the dwh\_schema exists and sets the context. The script then iterates through this dictionary, executing each complex SQL statement. This process joins, selects, transforms, and aggregates data from the raw\_schema tables, effectively **creating and populating the star schema tables** directly into the dwh\_schema.





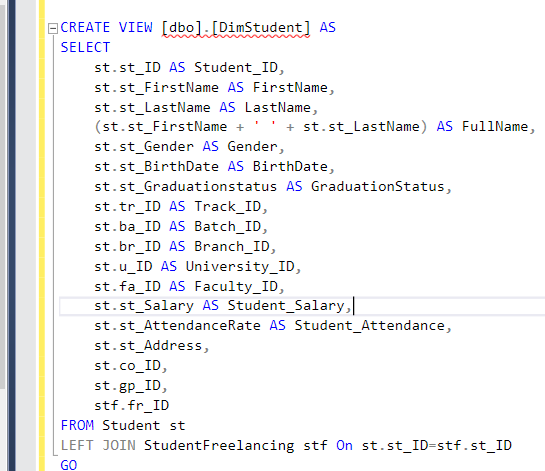




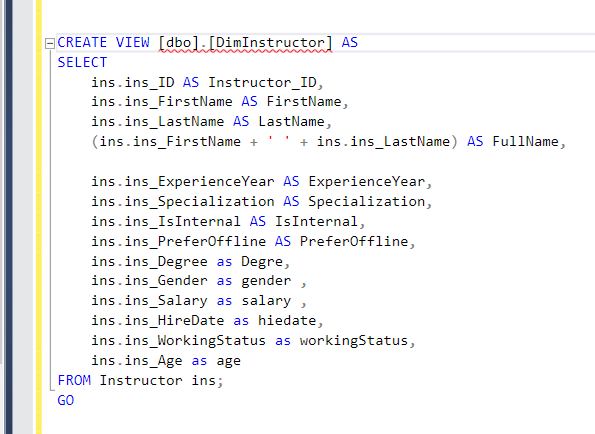
# 8. Data Warehouse Modeling

Below are the SQL Views statements for implementing the data OLAP modeling schema:

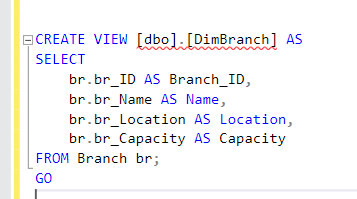
* **Dim\_Student:** Stores student-related information and foreign Keys.



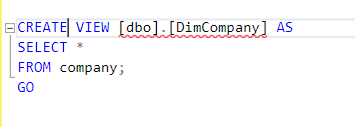
* **Dim\_Instructor:** Holds personal and professional details of instructors.



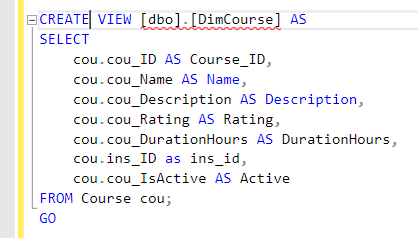
* **Dim\_Branch:** Contains data about ITI branches.



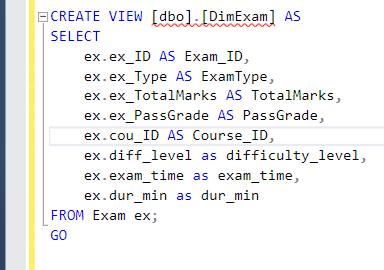
* **Dim\_Company:** Includes company details and employment info for students .



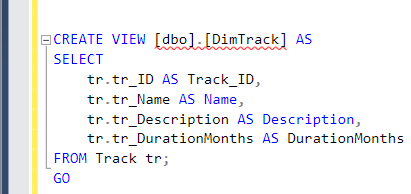
* **Dim\_Course:** Describes course data and associated topic information.



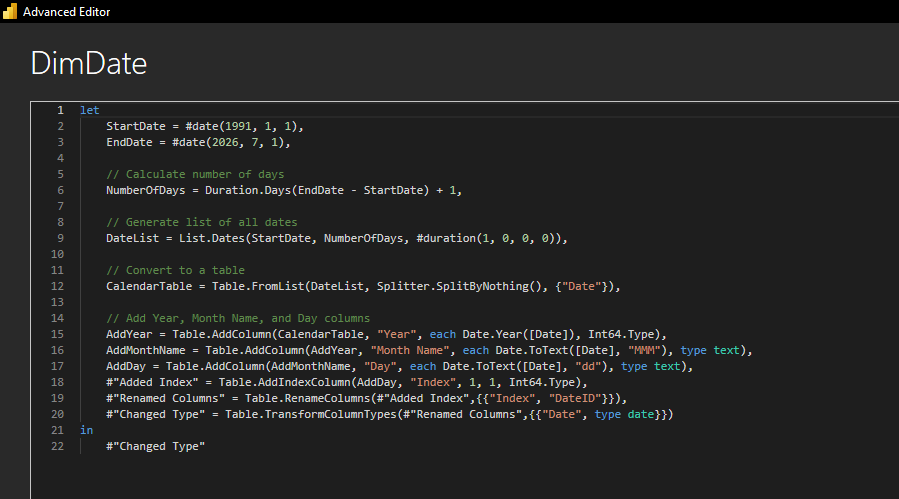
* **Dim\_Exam:** Captures exam details, including related questions and choices.



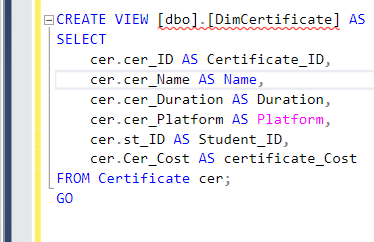
* **Dim\_Track:** Defines track information along with associated program and batch data.



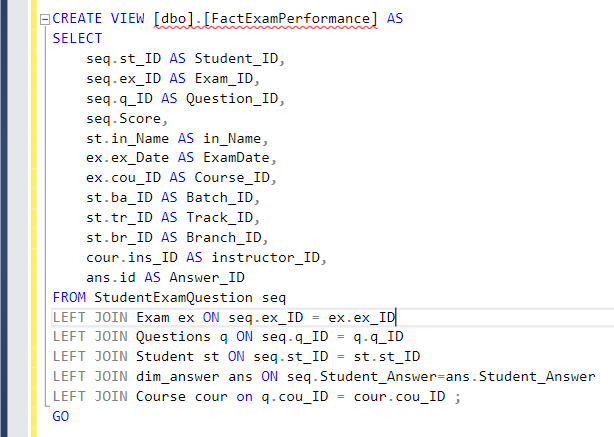
* **DimDate:** A standard date dimension for time-based analytics. (M language)



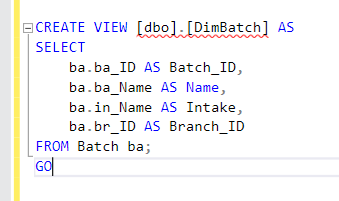
* **Certificate:** Fact-like dimension that links students to certificates with validity tracking.



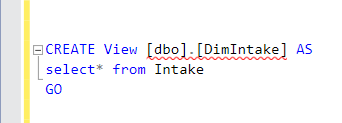
* **Fact\_Table:** Central fact table capturing exam performance and connections to dimensions.



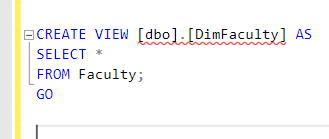
* **Dim Batch :** include data for all Batches



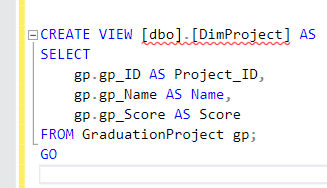
* **Dim Intake :** data of all intake history.



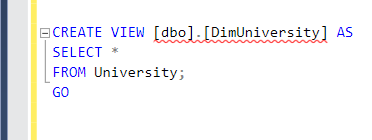
* **Dim Faculty** : data of Faculties.



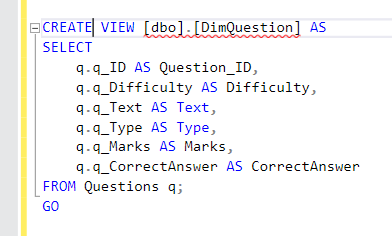
* **Dim graduation Project**

****

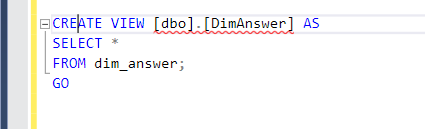
* **Dim University**

****

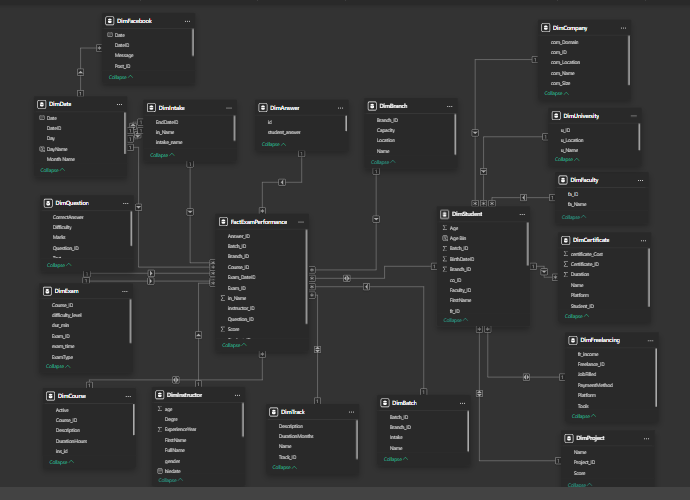
* **Dim Question: holds all question data.**

****

**Dim Answer :** holds unique answers **s**tudent could choose

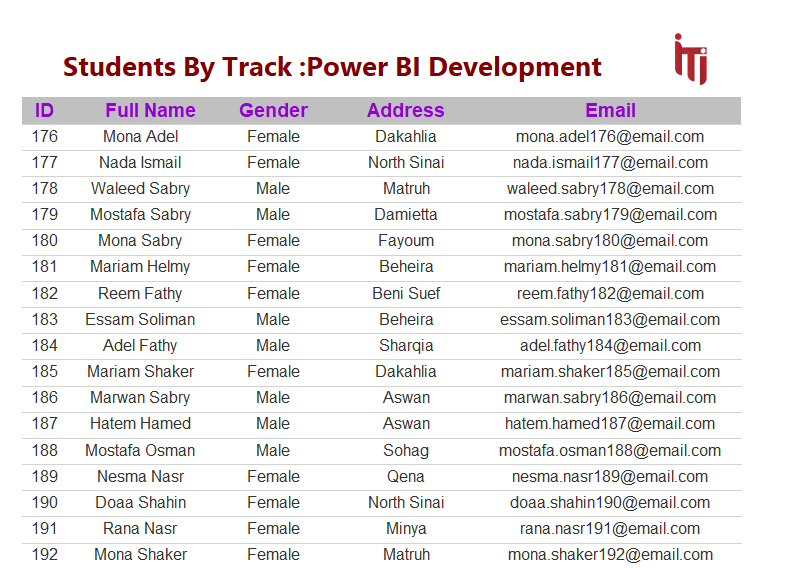
****

**The Model in Power BI Model view :**

****

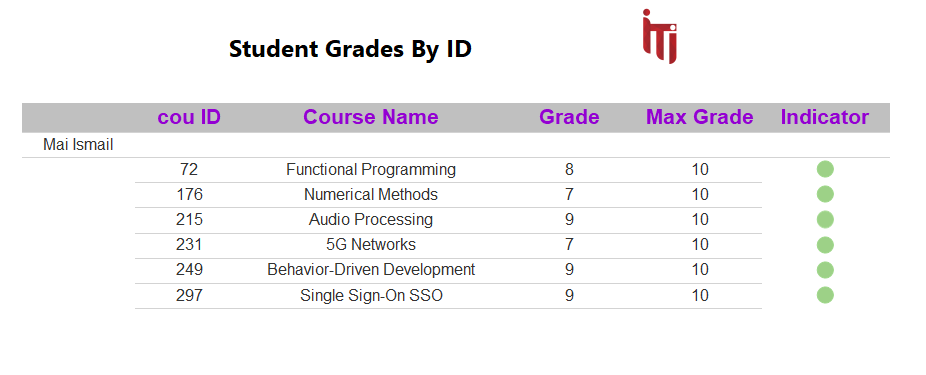
**9. The Reports in SQL Server Reporting Service (SSRS):**

•**Report 1:** Report Students and their information by track ID.

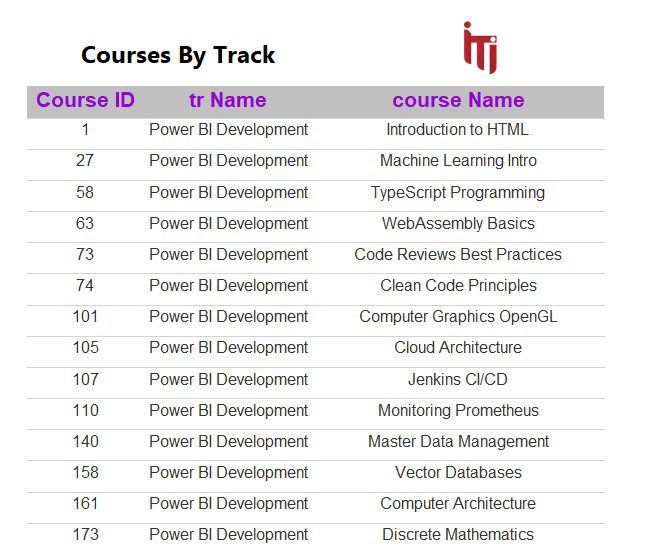


**Report 2**: Report that takes the student ID and returns the grades of the

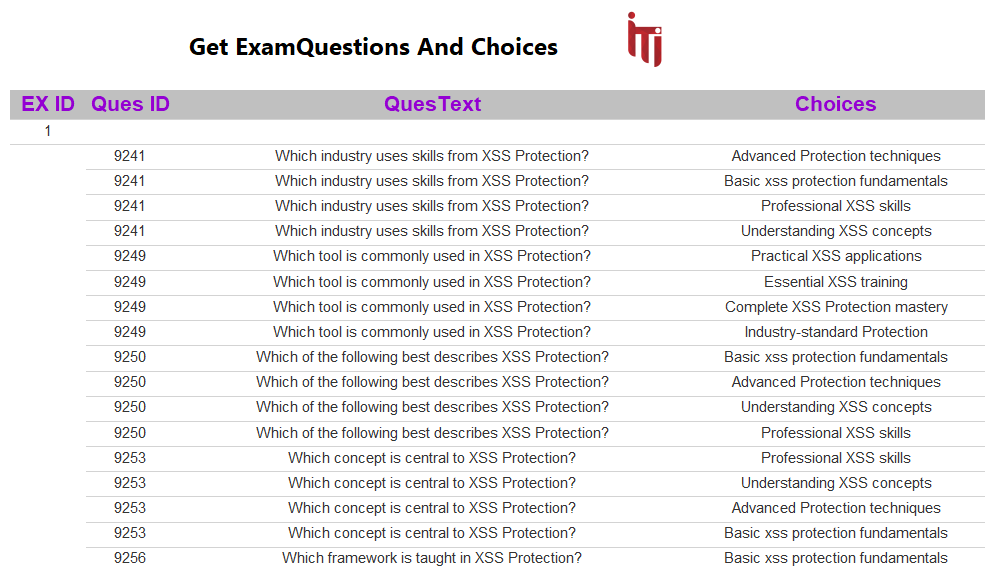
student in all exams.



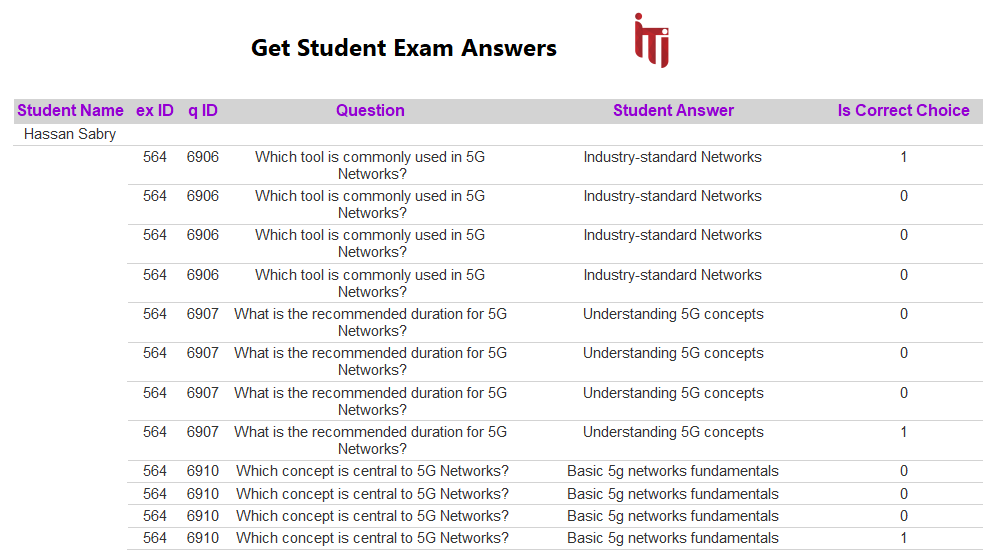
**Report 3**: report get all courses included in a track by track ID

****

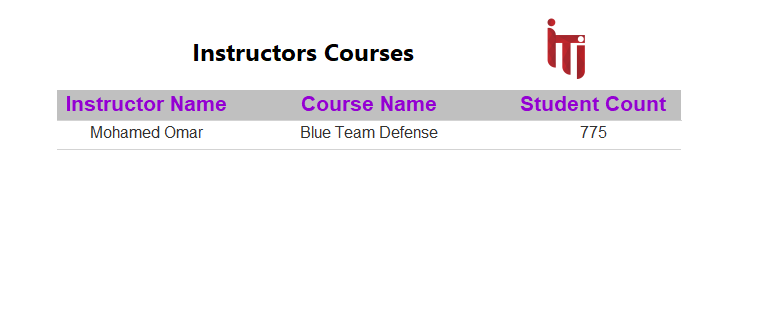
**Report 4:** Get questions of an exam with its choices by exam ID.

****

**Report 5:** get student answers on exam by student ID and Exam ID

****

**Report 6:** report get the instructor who instructs the course by course ID



# 10. Insights & visualization using Power BI

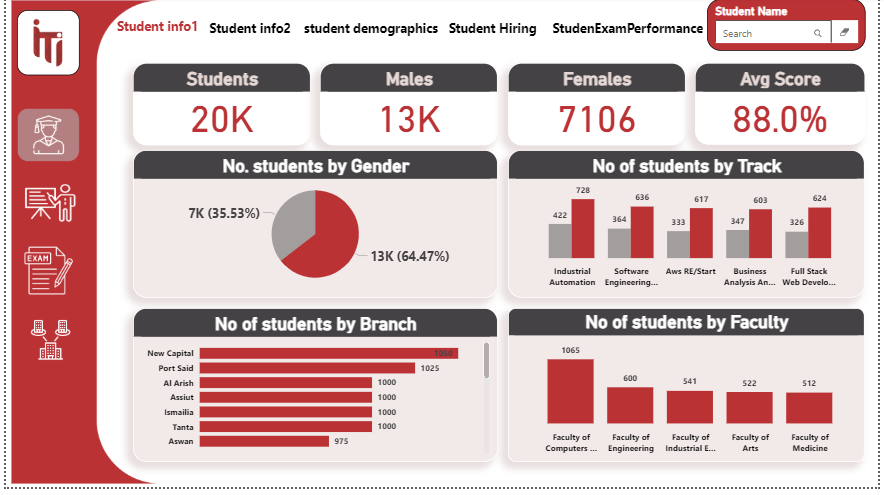
As part of the project, 20 interactive dashboards were developed using Power BI to provide meaningful insights into various aspects of the ITI Examination System. These dashboards cover key areas such as students, instructors, exams, courses, branches, partnering companies, certificates, freelance activities, and graduation projects.

Each dashboard was designed to support data-driven decision making, offering visual summaries, trends, and performance indicators.

The use of filters and interactive elements allows users to explore the data dynamically and focus on specific segments or metrics as needed.

The main objective of these visualizations was to identify patterns, track performance, and improve the overall efficiency of academic and administrative processes. These insights empowered stakeholders to make informed decisions based on real-time data and comprehensive analytics.

## • Overview dashboard

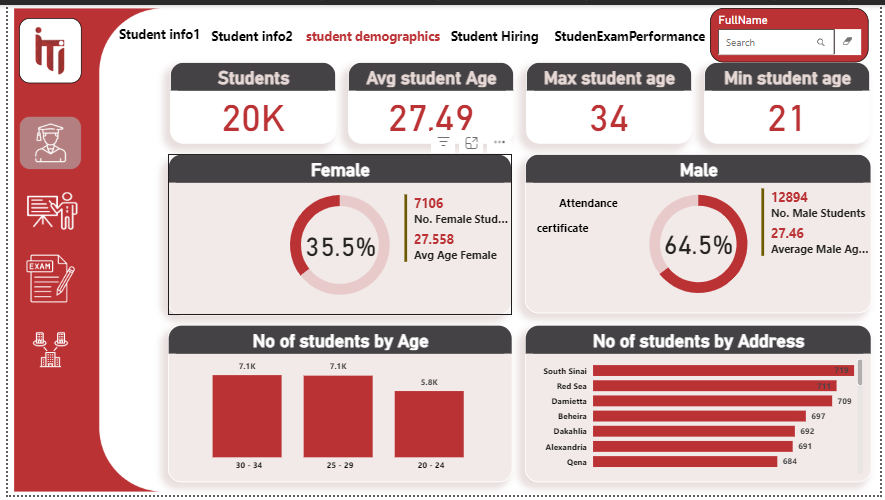
dashboard provides an overview of ITI students, highlighting total student count, gender distribution, and average performance. It visualizes student numbers by gender, branch, track, and faculty, offering insights into enrollment patterns and academic diversity across the institute.

## Student overview dashboard

dashboard summarizes student performance and graduation outcomes, showing total students, average attendance, graduation rate, and employment rate. It visualizes graduation trends by year, university, and faculty, providing insights into academic success and post-graduation hiring performance.**A screenshot of a computer screen

AI-generated content may be incorrect.**

## • Student demographic dashboard

dashboard presents student demographic insights, including total count, average age, and gender distribution. It highlights male and female participation rates, age ranges, and regional representation, offering a clear overview of student diversity across different locations and age groups.

## Student Hired dashboard

dashboard tracks Student Hiring Performance, displaying key outcomes like the Hired Student Percent and total Hired Graduates Count. It provides breakdowns of hiring by Company Size, Location, and Field to analyze job placement trends effectively. This is crucial for evaluating program success and informing career guidance strategies.**A screenshot of a graph

AI-generated content may be incorrect.**

## • Student Exam Performance

This Student Exam Performance dashboard summarizes overall academic activity, showing the total number of students, exams, and courses. It visualizes the average score trend over time, highlights the Lowest 5 Exams by student score, and tracks individual student performance. The tool is essential for monitoring educational **A screenshot of a computer screen

AI-generated content may be incorrect.**

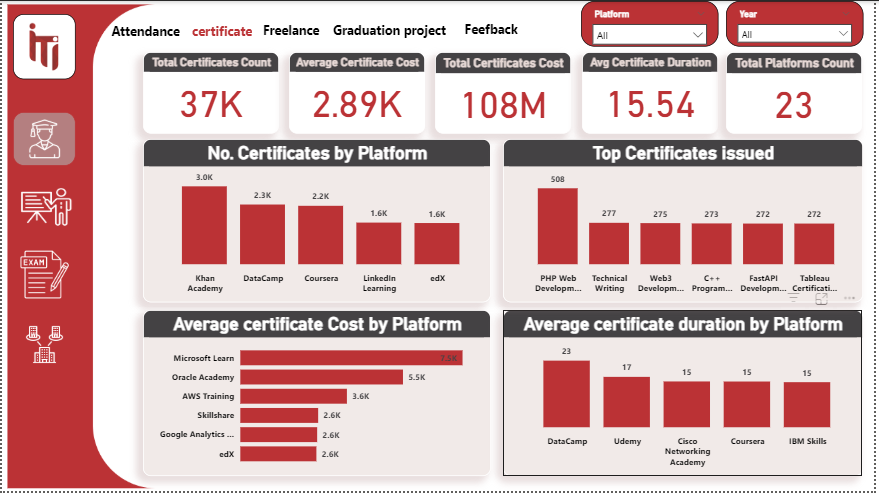
**Attendance Analysis dashboard**

Attendance Analysis dashboard tracks student presence with key metrics like total Certificates issued and the overall Average Attendance percentage. It identifies students with low attendance and shows trends over time and by intake name and branch location. This is vital for monitoring engagement and ensuring effective program delivery across all sites .A screenshot of a graph

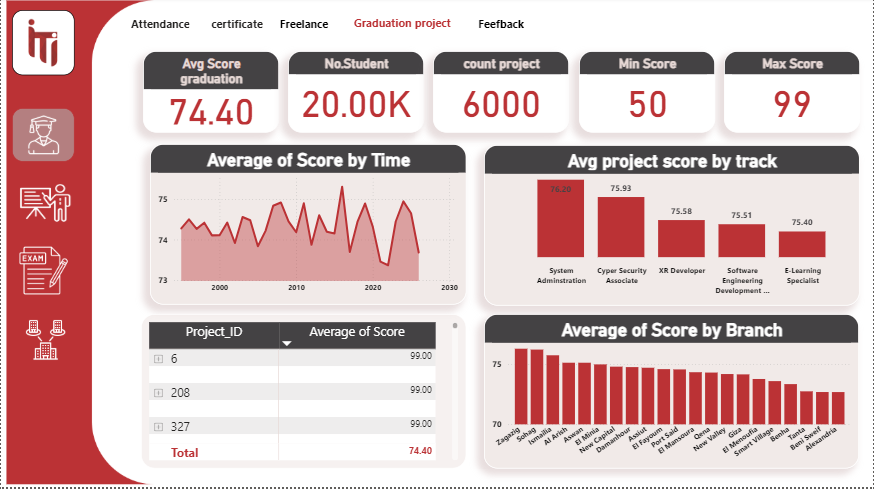
AI-generated content may be incorrect.

## • Certificates Dashboard

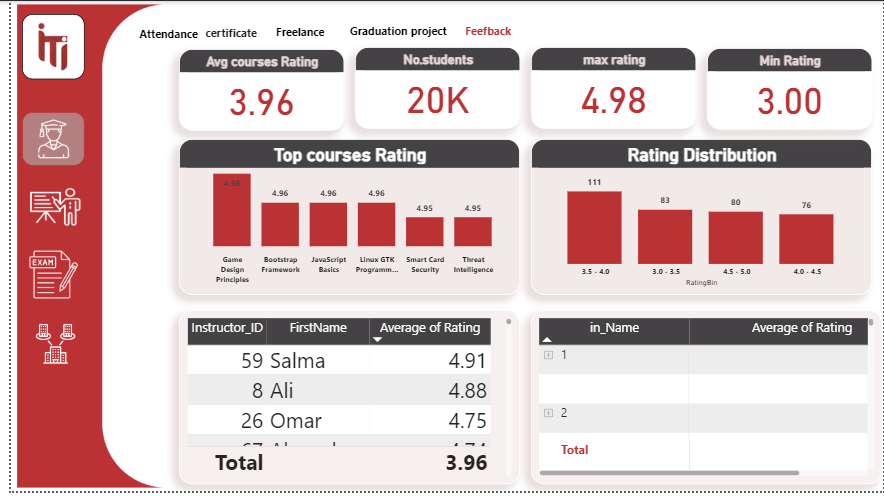
This Certificates Dashboard provides an overview of training accomplishments, showing the Total Certificates Count and Total Feedback Count. It analyzes certificate acquisition by Platform and details the Top Certificates Issued and their average duration and cost. This allows for assessment of external training efficacy and investment return.



**Graduation Project dashboard**

Graduation Project dashboard analyzes final project performance, highlighting the Average Score for Graduation and overall feedback count. It tracks score trends over time, compares the Average Project Score by Track, and provides a breakdown of scores by Branch. This is key for assessing the final academic quality and comparing performance across different specialties and locations. 

## • Feedback dashboard

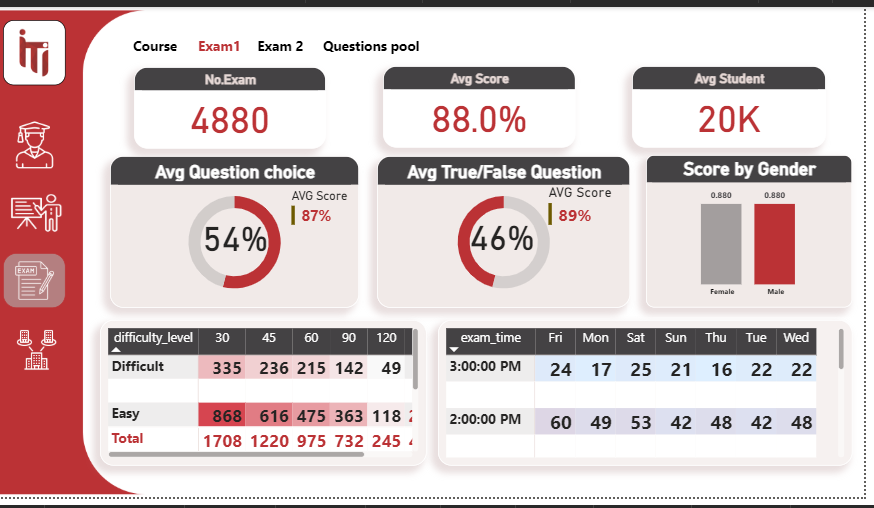
Feedback Dashboard provides a deep dive into course and instructor evaluation, showing the Average Course Rating and total number of students. It visualizes the Top Courses by rating and the overall Rating Distribution across all feedback received. The dashboard also tracks the Average Rating by Instructor for quality assurance and performance review. 

## Freelance dashboard

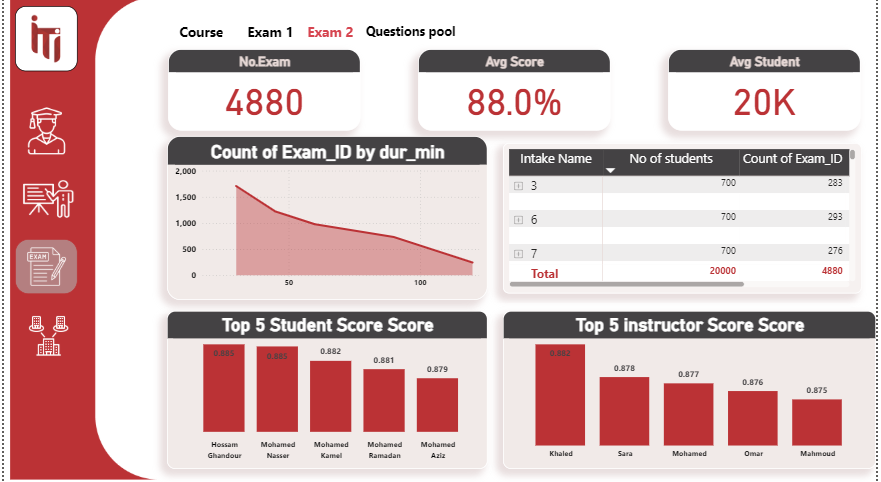
Freelance Dashboard analyzes graduates' external work, showcasing the Total Freelance Count and the number of students involved. It tracks freelance income by platform, breaks down the work by the job field and the tools utilized. This is a critical view for understanding the real-world applicability of skills and graduate earning potential. A screenshot of a computer

AI-generated content may be incorrect.

## • Exam dashboard

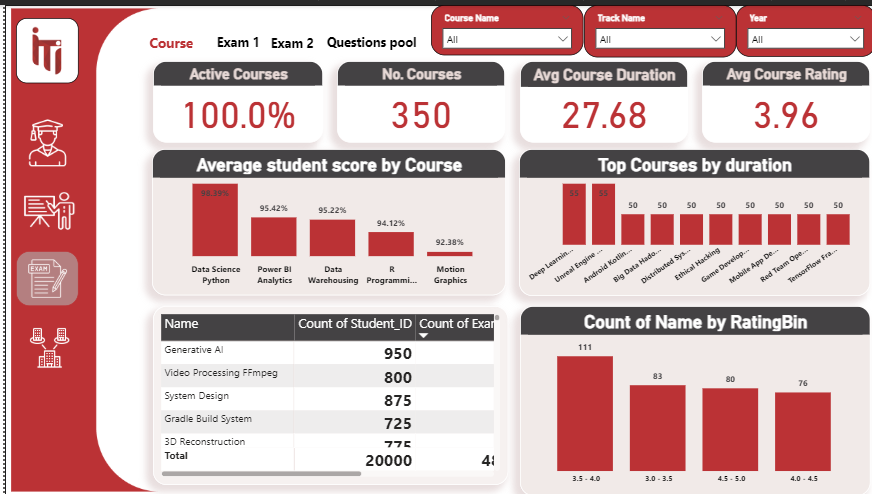
This Exam Analysis dashboard focuses on initial assessment performance, displaying the Total Number of Exams and the overall Average Score of 88.0%. It provides a breakdown of performance by question type (choice vs. true/false) and difficulty level. The dashboard also tracks Score by Gender and exam timing for comprehensive test evaluation.

## Exam dashboard

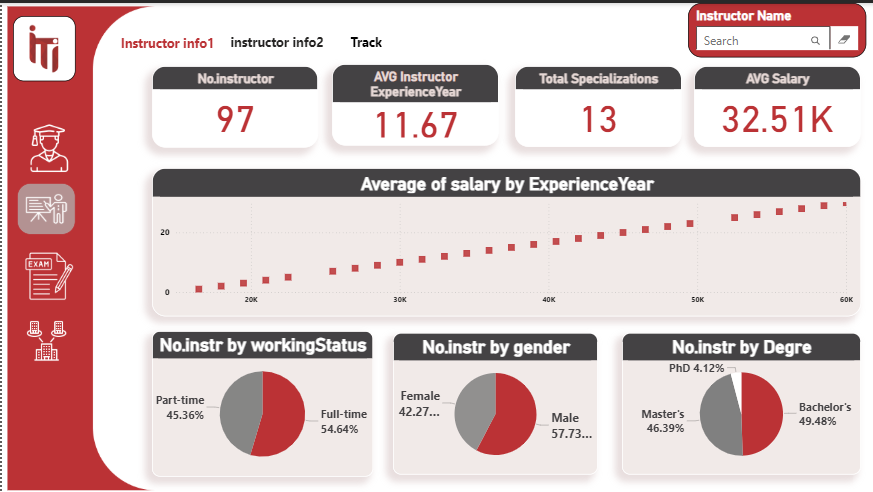
Exam Analysis dashboard details subsequent exam performance, again showing the Total Number of Exams. It tracks the Count of Exam IDs by duration, segments data by Intake Name, and highlights the Top 5 Student and Instructor Scores. This provides comparative insights into progress and educator effectiveness after the initial assessment. 

## • Course dashboard

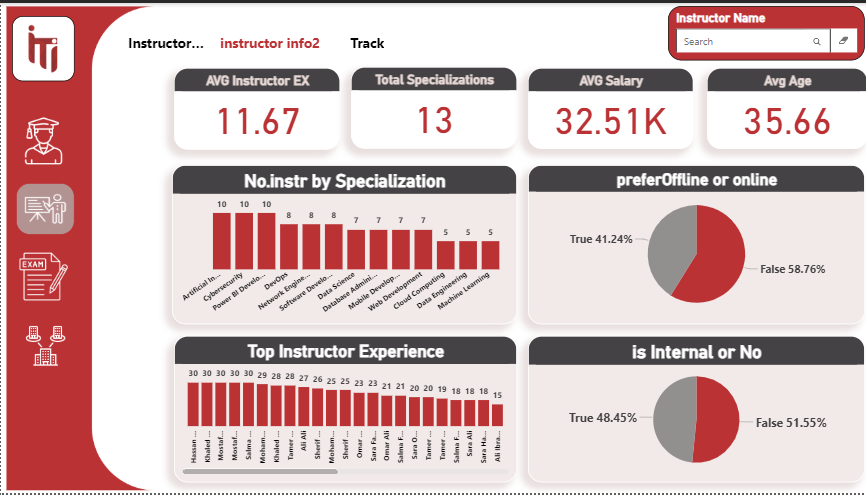
Course Overview dashboard tracks educational delivery, displaying the Number of Courses, Average Course Duration, and overall Average Course Rating. It highlights the Average Student Score by Course and identifies the Top Courses by Duration and the Count of Students by Exam taken. curriculum performance and student success within specific course modules.



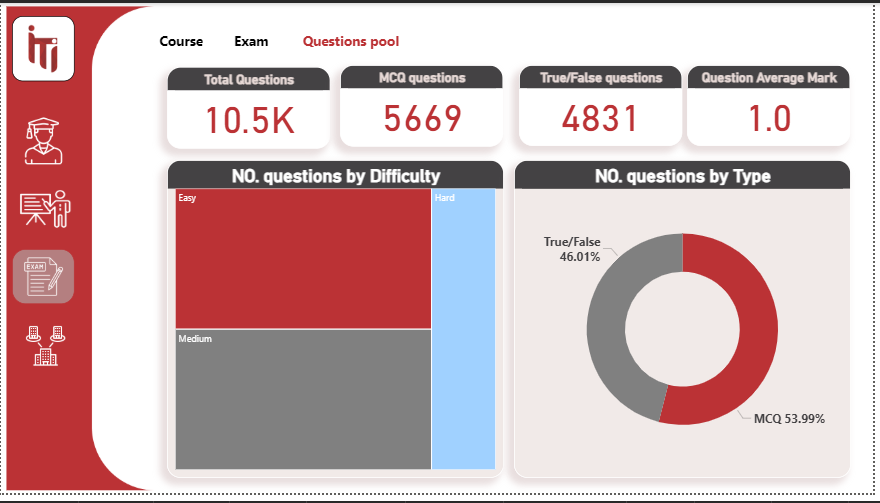
## Instructor dashboard

Instructor Information dashboard summarizes faculty demographics and compensation, showing the Total Number of Instructors and their Average Experience Year. It tracks the Average Salary and visualizes the relationship between salary and experience, segmenting instructors by working status, gender, and degree level. This is crucial for human resource planning and compensation analysis within the program.

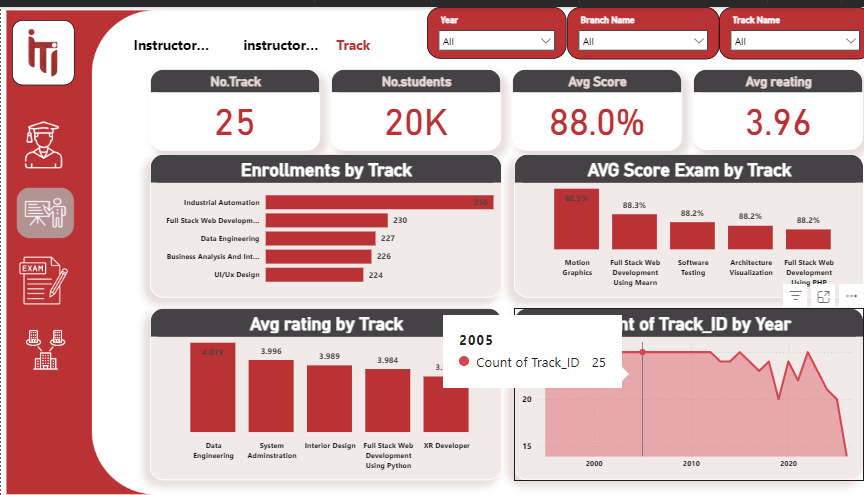
## • Instructor dashboard

This Instructor Tracking dashboard focuses on instructor specializations and work preferences, displaying the Average Instructor Experience and Total Specializations. It visualizes the distribution of instructors by specialization, their preference for offline or online work, and their status as internal or external faculty. This helps in resource allocation and understanding the profile of the teaching staff. 

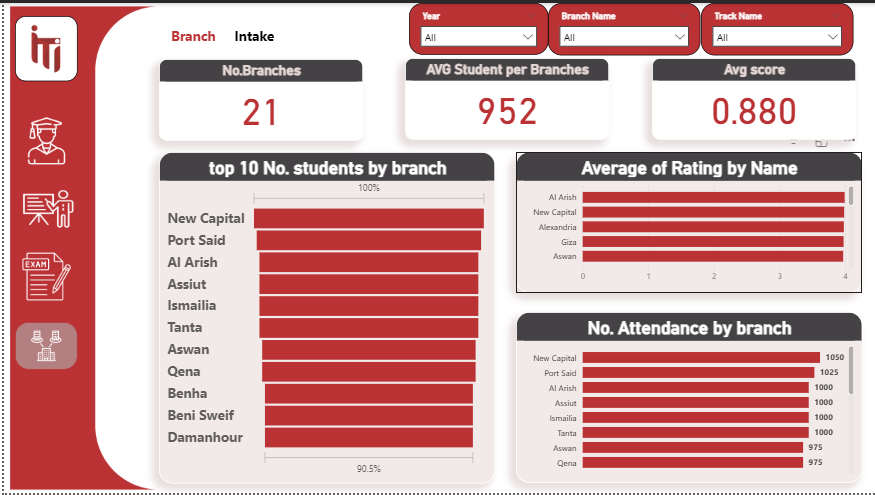
## Question pool dashboard

This Questions Pool dashboard provides an inventory of the available test content, showing the Total Questions and a breakdown of MCQ vs. True/False questions. It visualizes the Number of Questions by Difficulty (Easy, Medium, Hard) and the overall distribution by type. This is critical for managing exam resources and ensuring balanced question creation. 

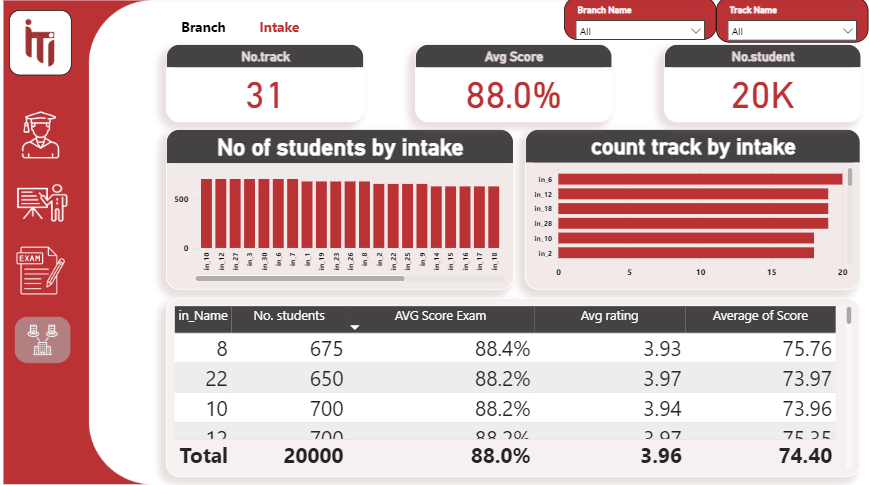
## • Track performance dashboard

This Track Performance dashboard offers a comparative analysis of different educational paths, showing the Number of Tracks and Total Students enrolled. It highlights Enrollments by Track, compares Average Exam Scores by Track, and visualizes the Average Rating by Track. This is essential for evaluating the relative success and popularity of each specialization. 

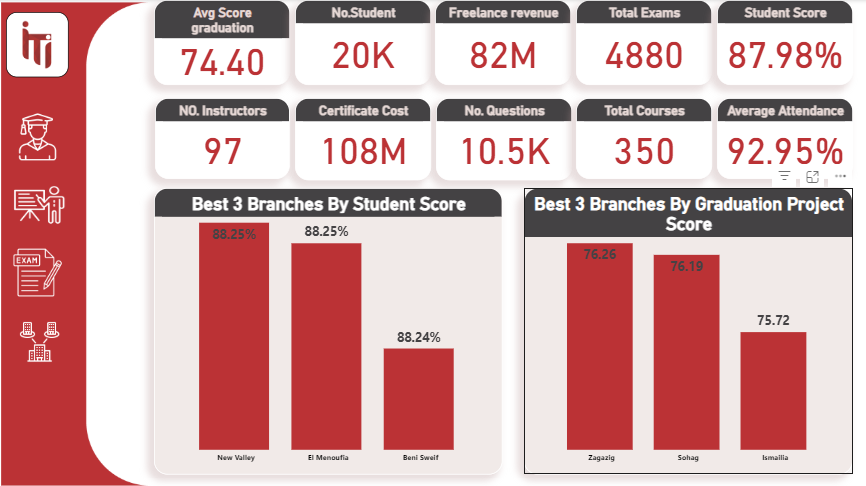
## Branch dashboard

Branch Performance dashboard analyzes institutional operations across different locations, displaying the Number of Branches and the Average Student Count per Branch. It highlights the Top 10 Branches by Student Count, their respective Attendance numbers, and the Average Rating by Branch Name. This is crucial for comparing regional performance and managing resource allocation effectively. 

## • Intake dashboard

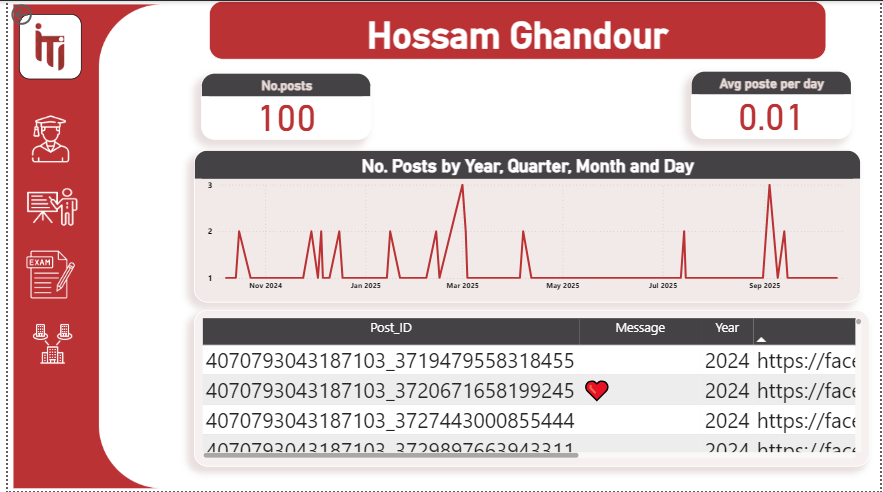
This Intake Analysis dashboard tracks student groups based on their start date, displaying the Number of Tracks, Total Students, and overall Average Score. It visualizes the Number of Students by Intake and the corresponding Track Count by Intake, detailing their exam scores and average ratings. This provides essential data for monitoring cohort performance and capacity planning. 

**Key Performance Indicators (KPIs) Dashboard**

This Key Performance Indicators (KPIs) Dashboard provides a high-level summary of program effectiveness across core areas. It tracks major metrics like Student Score, Freelance Revenue, and Average Attendance, along with the Best 3 Branches based on both student and graduation project scores. This is a vital executive tool for quick strategic oversight and performance monitoring 

**API page:**

API gets information from facebook page of one of students.



# 11. Examination System Website

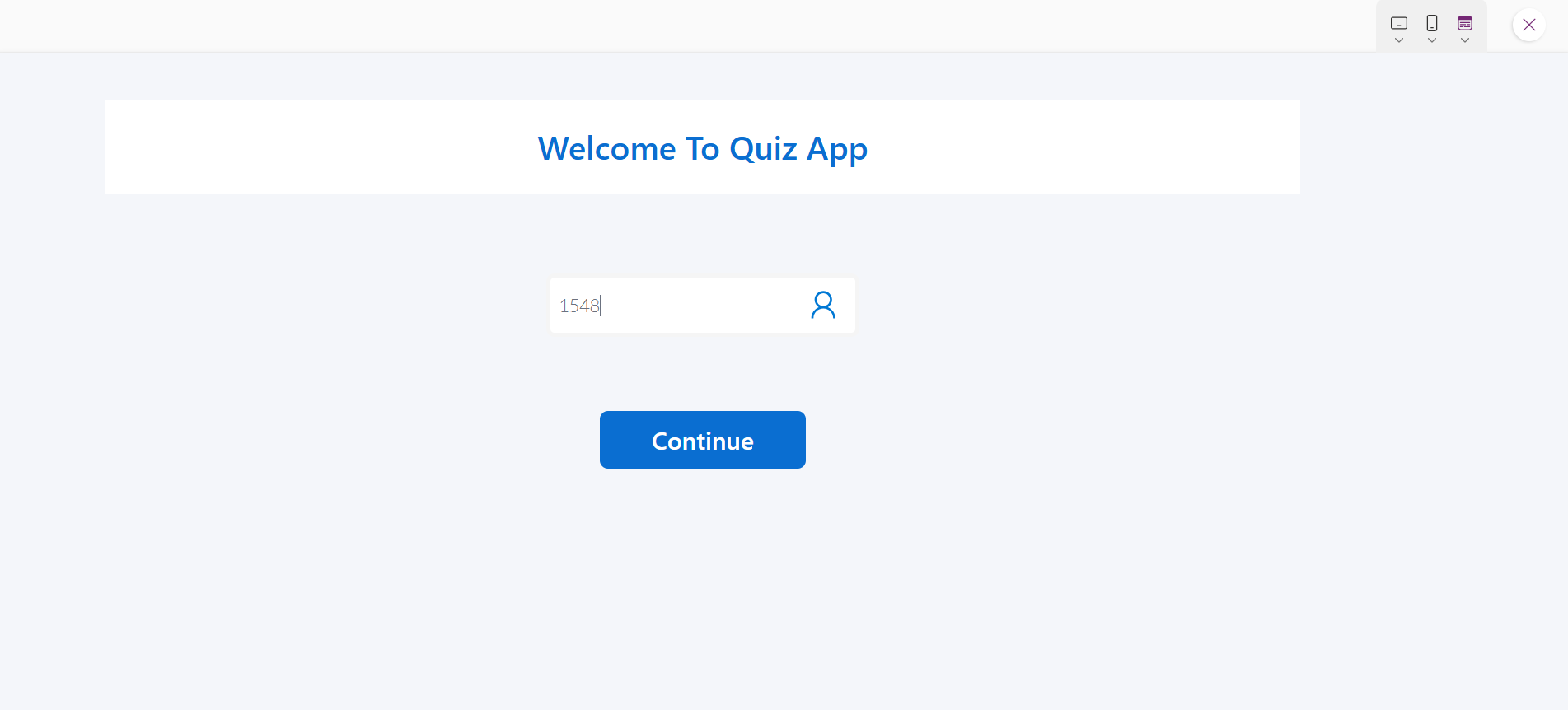
This chapter describes the main user interface components of the Examination System Website. It outlines the core pages available for both instructors and students, along with the functionalities offered on each page. Screenshots are available for each of these pages.

**11.1 Login Screen**

Purpose: Authenticate student before accessing the app.  
Components: Text input for Student ID or Email.

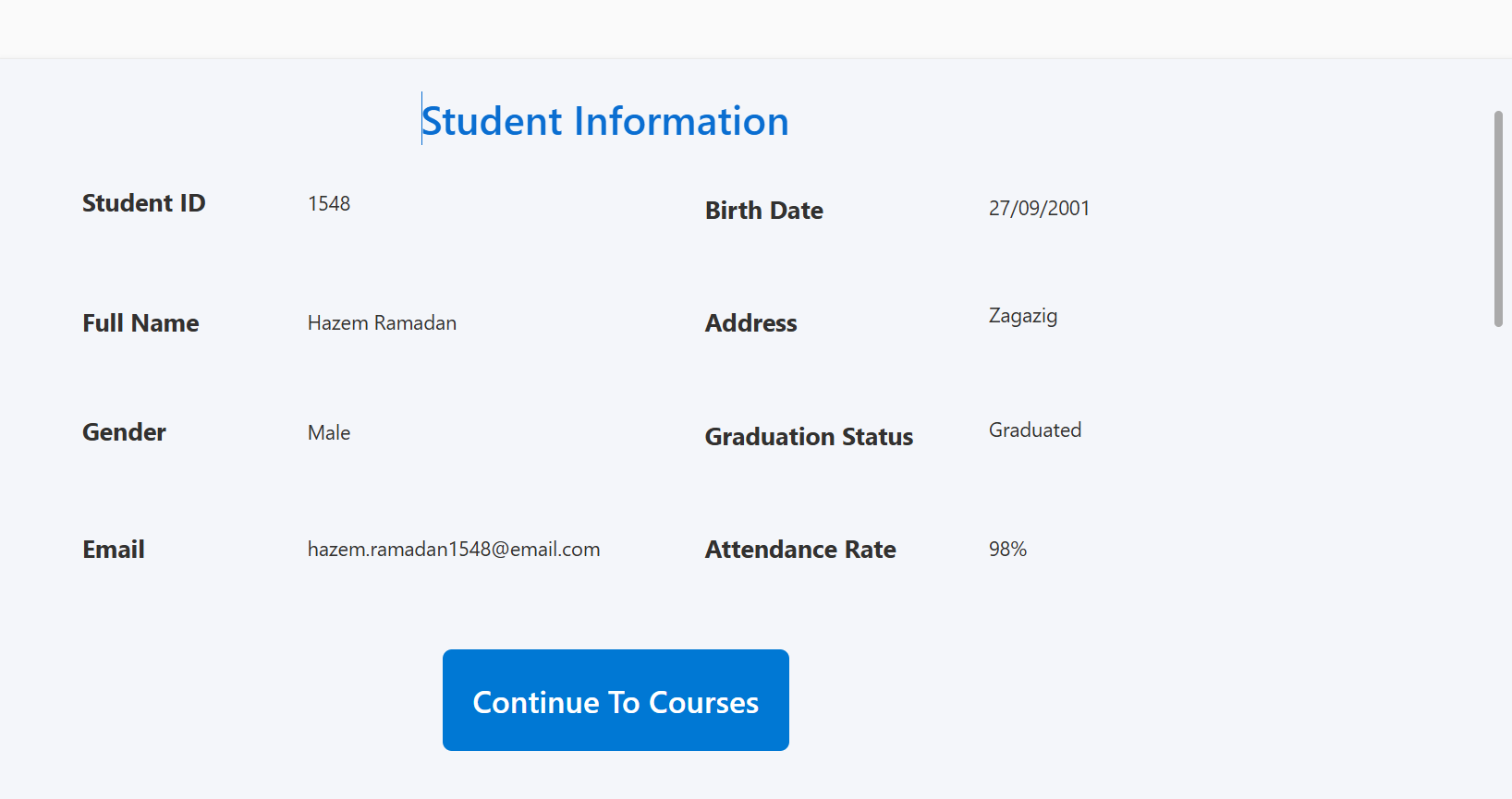
Password input for authentication.

Login button (navigates to Student Info screen if valid).



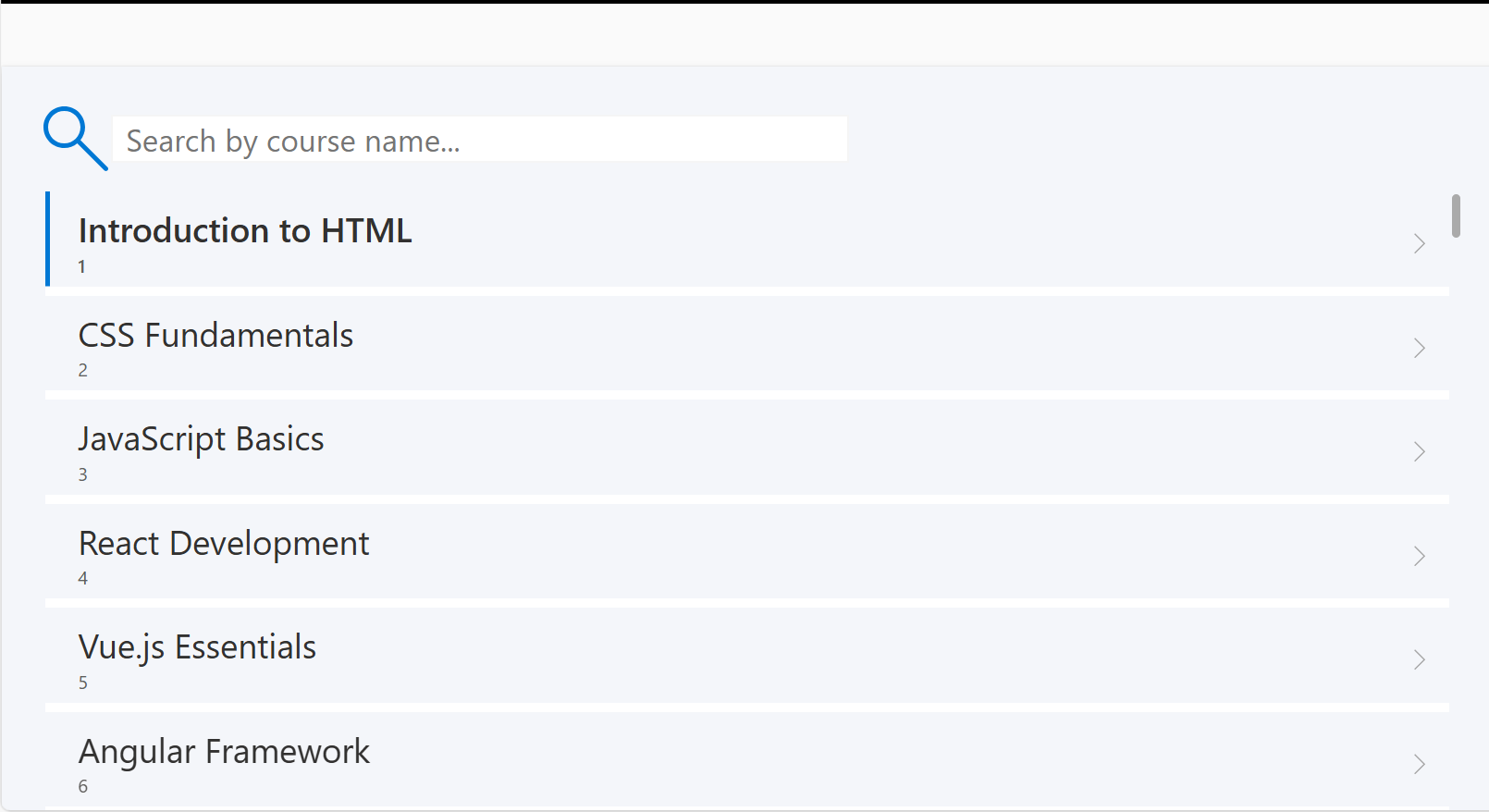
**11.2 Student Info Screen**

Purpose: Display and confirm student details after login.  
Components:  
Labels showing student name, ID, and department.  
Continue button → navigates to Courses screen.



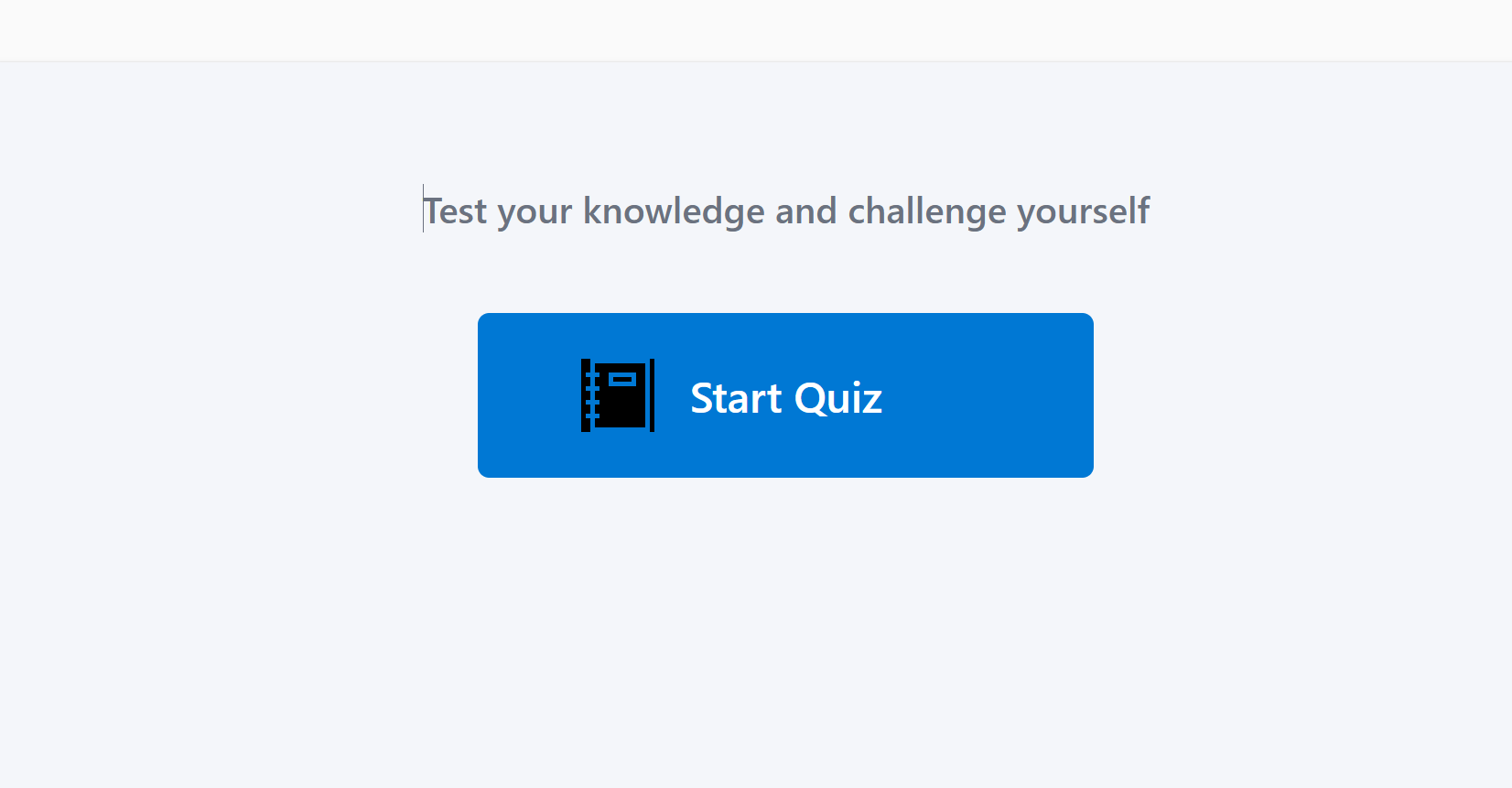
**11.3 Courses Screen**

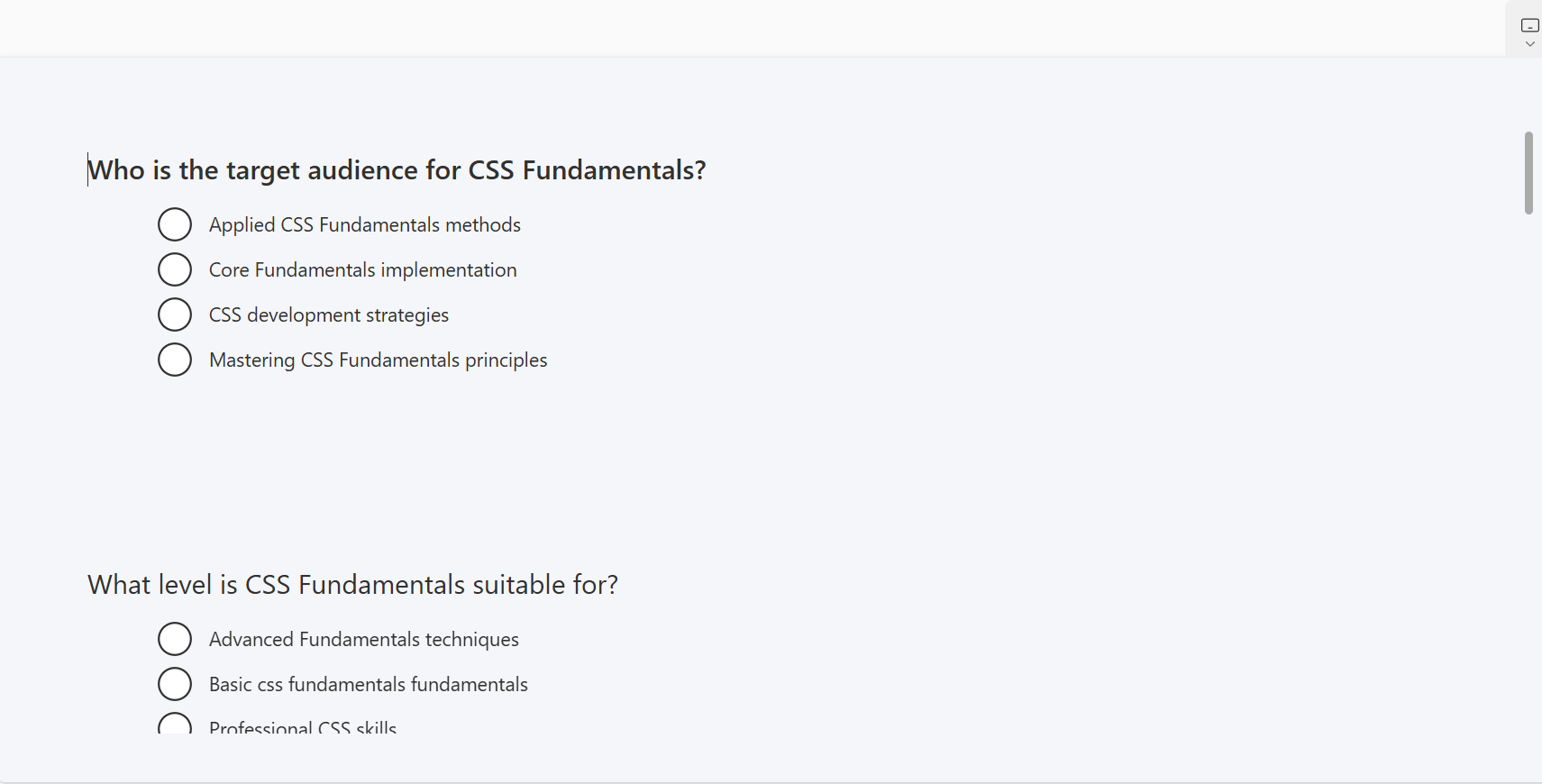
Purpose: Allow students to select which course they want to take a quiz in.  
Components:  
- Dropdown or gallery listing available courses.  
- Start Quiz button → runs stored procedure to fetch questions.



**11.4 Quiz Screen**

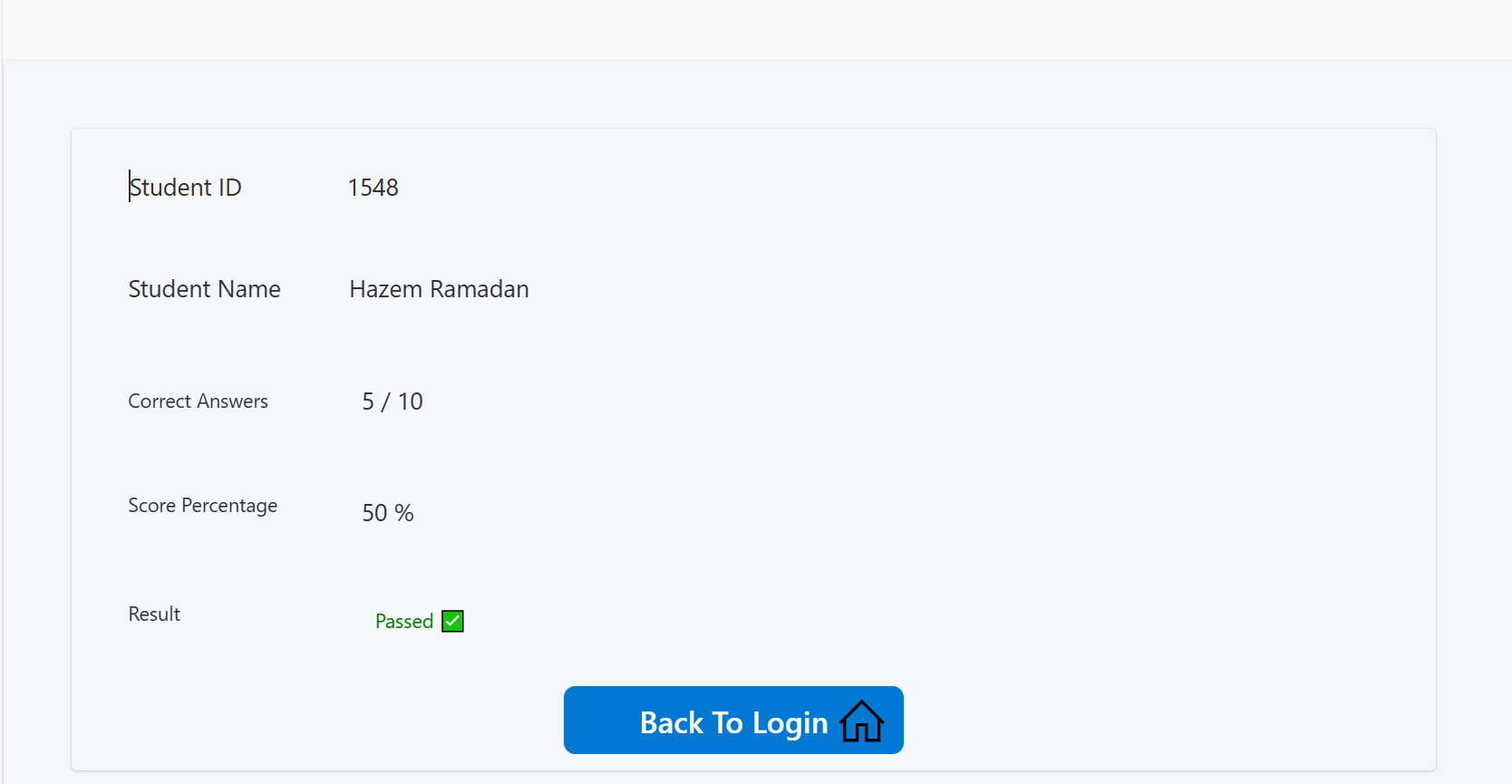
Purpose: Display quiz questions and collect answers.  
Components:  
- Question label (bound to quiz Questions).  
- Choice buttons (radio buttons or gallery).  
- Next button to navigate between questions.





**11.5 Results Screen**

Purpose: Show the student’s final score after submitting the quiz.  
Components:  
- Label showing total score and correct answers.  
- Save Result button to record the result in the SQL database.



**11.5 Database Data Storing.**

Tables Used:  
1. Students – holds student credentials and info.  
 - Columns: StudentID, Name, Email, Password  
2. Courses – stores available courses.  
 - Columns: CourseID, CourseName  
3. Questions – stores all quiz questions.  
 - Columns: QuestionID, QuestionText, CourseID  
4. QuestionChoices – holds multiple choices per question.  
 - Columns: ChoiceID, q\_ID, ChoiceText, IsCorrect  
5. Results – stores quiz attempts and scores.  
 - Columns: ResultID, StudentID, CourseName, Score, Date

**11.6 Conclusion**

The Quiz App provides a structured and efficient digital assessment solution using Power Apps and SQL Server.   
It automates quiz generation, result calculation, and data management while ensuring simplicity for both students and administrators.