Examination System

MansouraBranch ITP 2024 Q2 Bi

Project Team:

Noura Mahmoud Algohary

Farah Mohamed Azmy

Ahmed Hamdy

Mohamed Essam

Supervisors:

Rami Nagi

Nadia Saleh

Table of Contents:

1-Overview

An examination system program is a software application designed to facilitate the management and administration of exams, assessments, and tests. It streamlines the process of creating, scheduling, conducting, and grading exams, making it easier for educational institutions, or businesses to manage their assessment processes efficiently. Here's an overview of the key components and features in The Examination system program:

Admin:

-Insert,UpdateDelete Students

-Insert,Update,Delete Instructors

-Insert,Update,Delete Branches

-Insert,Update,Delete Tracks

-Insert,Update,Delete Courses

Instructors:

-Create Exams: Instructors can create new exams by selecting questions from the question bank or creating new questions.

-Manage Question Bank: Instructors can add, edit, categorize, and delete questions in the question bank.

Students:

-View Information : View Data about himself

-View Courses : View Enrolled Courses

-Take Exams: Take exams online or offline, depending on the mode of delivery, within the scheduled time frame.

-Receive Results: Access exam results and feedback provided by instructors.

Technologies Used:

- SQL Server
- PowerApps
- PowerBI
- SSIS
- SSRS

2-DataBase

2.1 Entity-Relationship Diagram (ERD) Using Darw.io :

Steps:

Initial Planning:

- We Defined the scope of the ERD.
- We Identified the entities, attributes, and relationships to be included.

Accessing draw.io:

- We Accessed the draw.io website.

Creating a New Diagram:

- We Choose appropriate shapes for entities, attributes, relationships from the shapes provided by draw.io.

Adding Entities:

Defining Attributes:

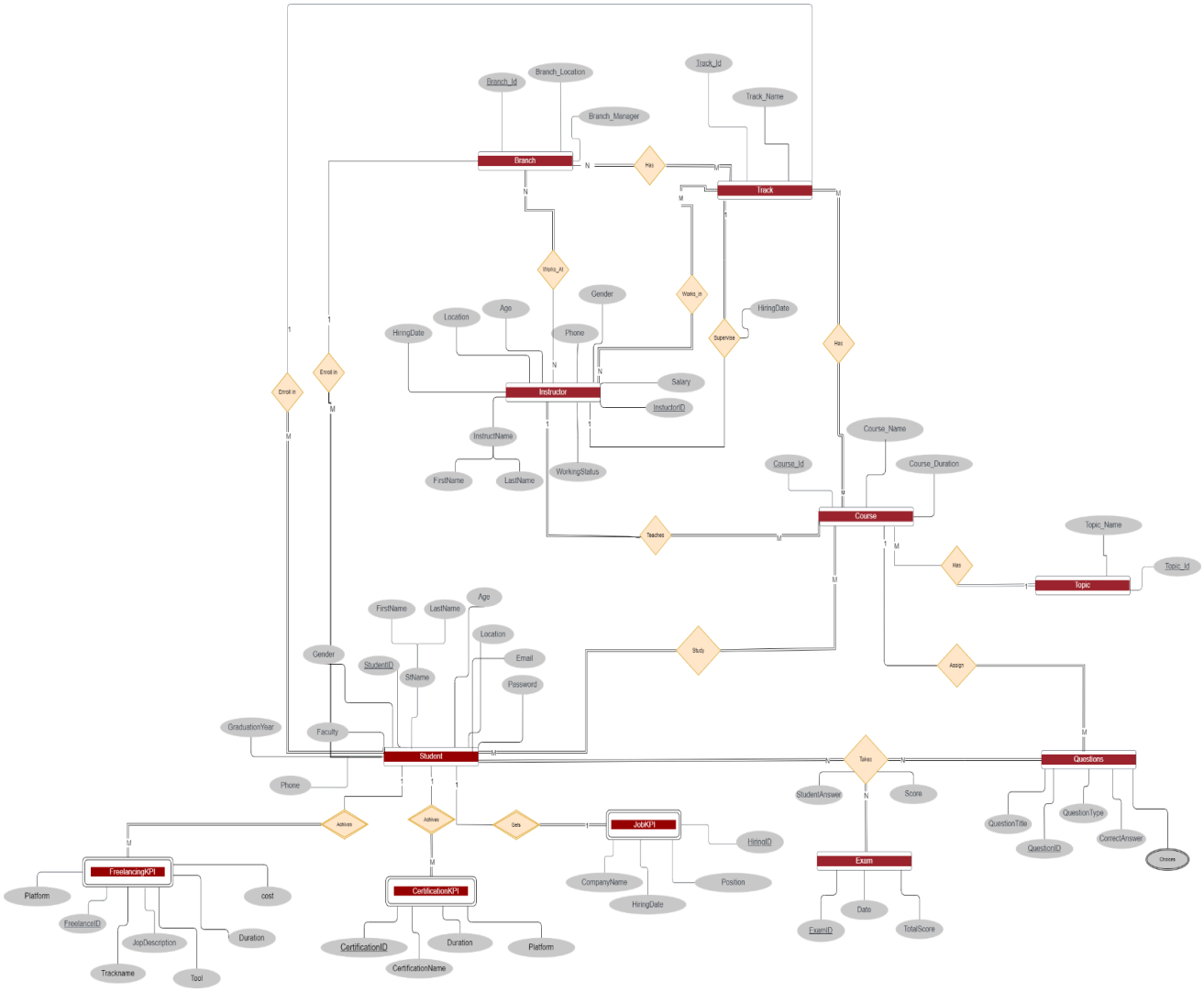
Establishing Relationships:

- Used lines to establish relationships between entities.

Refining the Diagram:

- Organize entities and relationships neatly on the canvas.

Results:


Branch_Id
Branch_Location
Branch_Manager
Track_Id
Track_Name
Branch
Has
Track
Works_At
Works_In
Gender
Age
Location
HiringDate
Phone
Supervise
Has
Enroll In
Salary
Instructor
InstructorID
InstructName
FirstName
LastName
WorkingStatus
Course_Name
Course_Id
Course_Duration
Course
Teaches
Topic_Name
Has
Topic_Id
Topic
FirstName
LastName
Age
Location
Email
Password
Gender
StudentID
SfName
Study
Assign
GraduationYear
Faculty
Student
Phone
Takes
Questions
StudentAnswer
Score
QuestionTitle
QuestionType
QuestionID
CorrectAnswer
Choices
Achives
Achives
Gets
JobKPI
HiringID
CompanyName
Position
HiringDate
Exam
ExamID
Date
TotalScore
FreelancingKPI
Platform
cost
FreelancerID
JopDescription
Duration
Trackname
Tool
CertificationKPI
CertificationID
Duration
Platform
CertificationName

2.2 Mapping:

Steps:

We Reviewed the ERD:
Identified Entities:

- Review each entity in the ERD and identify them for translation into tables in the database schema.

We Translated Entities into Tables:

- We Created a table for each entity identified in the ERD.

We Identified Attributes:

- For each entity, identify the attributes (fields) represented in the ERD.

We Defined Columns:

- Created columns within each table to represent the attributes identified in the ERD.

Primary Keys:

- Identified the primary key attribute(s) for each entity in the ERD.

Translated Relationships:

- Review the relationships between entities in the ERD.
- Translate these relationships into foreign key constraints in the database schema.

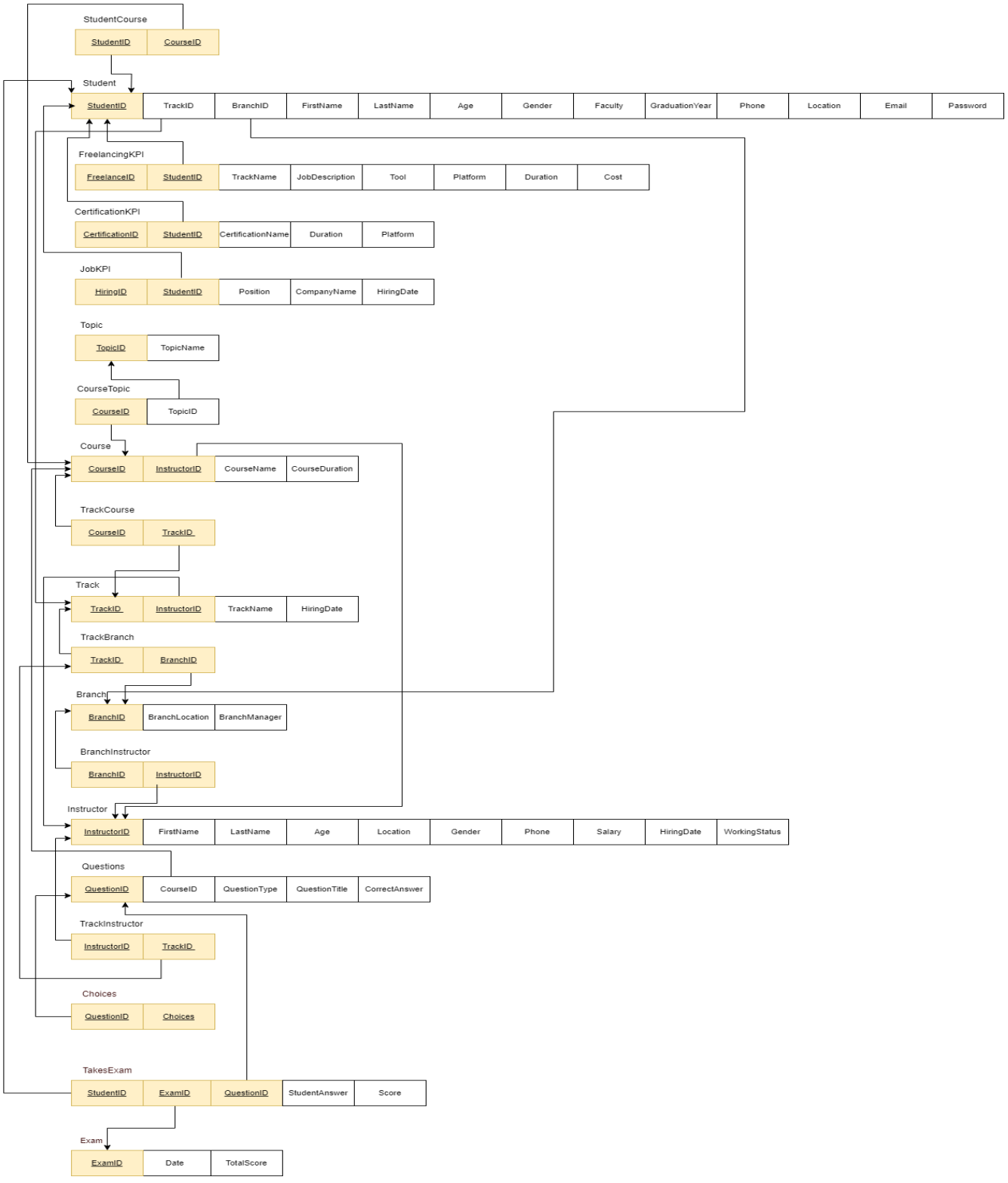
Foreign Keys:

- Add foreign key columns to child tables, referencing the primary key(s) of the related parent table(s).

Normalization:

- Applied normalization techniques to ensure that the database schema is structured efficiently and minimizes redundancy.

Results:


StudentCourse
StudentID
CourseID
Student
StudentID
TrackID
BranchID
FirstName
LastName
Age
Gender
Faculty
GraduationYear
Phone
Location
Email
Password
FreelancingKPI
FreelanceID
StudentID
TrackName
JobDescription
Tool
Platform
Duration
Cost
CertificationKPI
CertificationID
StudentID
CertificationName
Duration
Platform
JobKPI
HiringID
StudentID
Position
CompanyName
HiringDate
Topic
TopicID
TopicName
CourseTopic
CourseID
TopicID
Course
CourseID
InstructorID
CourseName
CourseDuration
TrackCourse
CourseID
TrackID
Track
TrackID
InstructorID
TrackName
HiringDate
TrackBranch
TrackID
BranchID
Branch
BranchID
BranchLocation
BranchManager
BranchInstructor
BranchID
InstructorID
Instructor
InstructorID
FirstName
LastName
Age
Location
Gender
Phone
Salary
HiringDate
WorkingStatus
Questions
QuestionID
CourseID
QuestionType
QuestionTitle
CorrectAnswer
TrackInstructor
InstructorID
TrackID
Choices
QuestionID
Choices
TakesExam
StudentID
ExamID
QuestionID
StudentAnswer
Score
Exam
ExamID
Date
TotalScore

2.3 Creating DataBase on SQL SERVER:

Steps:

We Wrote CREATE TABLE Statements:

- `CREATE TABLE` statement for each entity identified in the ERD.
- Specify the table names and list the attributes (columns) along with their data types and constraints.

Defined Columns:

- For each attribute in the entity,we specified the column names, data type, and any constraints such as `NOT NULL`, `UNIQUE`, or `DEFAULT` values.

Primary Key Constraints:

- Included the primary key constraint using the `PRIMARY KEY` keyword after defining the column(s).
- If the primary key is composed of multiple columns, use the `PRIMARY KEY` constraint followed by the column names enclosed in parentheses.

Foreign Key Constraints:

- Identify relationships between entities that require foreign key constraints.
- Add foreign key columns to child tables referencing the primary key(s) of the related parent table(s).
- Define foreign key constraints using the `FOREIGN KEY` keyword followed by the column name and the `REFERENCES` keyword specifying the parent table and column.

Execute SQL Statements:

- Execute the SQL statements.

Results:

Examples:

```
CREATE TABLE Student(
       StudentID INT PRIMARY KEY  NOT NULL,
       FirstName VARCHAR(50),
       LastName VARCHAR(50),
       Age INT CHECK (Age BETWEEN 22 AND 35),
       Gender VARCHAR(50),
       Phone int,
       [Location] VARCHAR(50),
       Faculty VARCHAR(50),
       GraduationYear INT,
       Email VARCHAR(50) NOT NULL CHECK (EMAIL LIKE '%_@%.%'),
    [Password] VARCHAR(50) NOT NULL);
ALTER TABLE Student
ADD TrackID INT,
CONSTRAINT Fk_TrackIDStudent FOREIGN kEY (TrackID) REFERENCES Track(TrackID);
```

```
CREATE TABLE Course(
       CourseID INT PRIMARY KEY NOT NULL,
       InstructorID INT,
       CourseName VARCHAR(255),
       CourseDuration INT,

CONSTRAINT FK_Course_Instructor FOREIGN KEY(InstructorID) REFERENCES Instructor(InstructorID)
 );
```

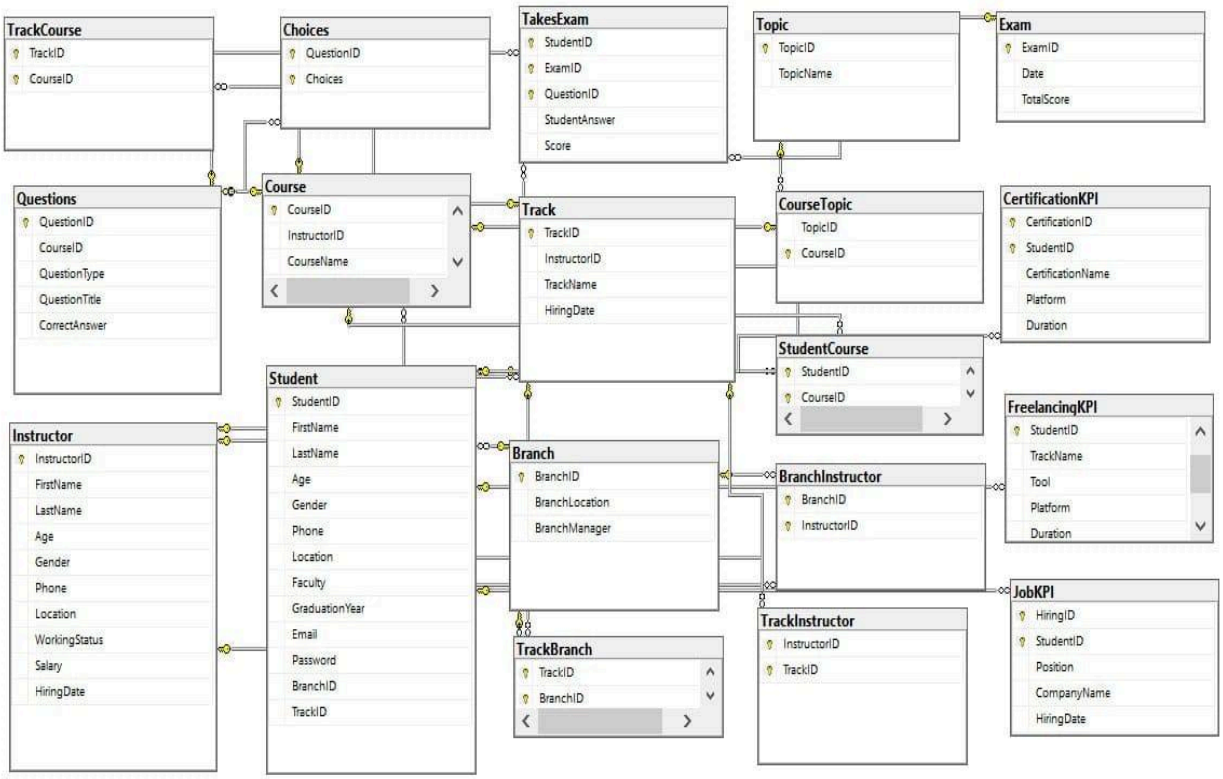
```
CREATE TABLE StudentCourse(
 StudentID INT NOT NULL,
 CourseID INT NOT NULL,

CONSTRAINT Fk_StudentID_StudentCourse FOREIGN kEY (StudentID) REFERENCES Student(StudentID),
CONSTRAINT FK_CourseID_StudentCourse FOREIGN kEY (CourseID)  REFERENCES  Course(CourseID),
CONSTRAINT PK_StudentCourse PRIMARY KEY( StudentID,CourseID)
);
```

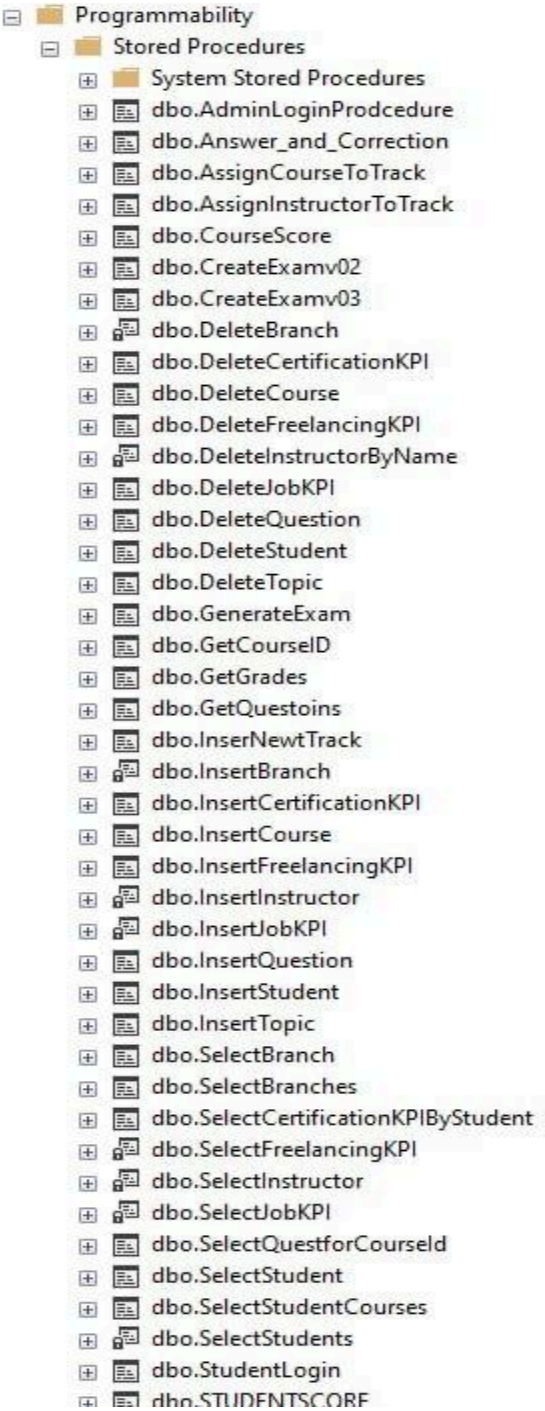
```
CREATE TABLE TakesExam(
       StudentID INT NOT NULL,
       ExamID INT NOT NULL,
       QuestionID INT NOT NULL,
       StudentAnswer VARCHAR(255) NOT NULL ,
       Score INT NOT NULL,

        CONSTRAINT PK_TakesExam  PRIMARY kEY (StudentID,QuestionID,ExamID),
        CONSTRAINT FK_StudentID_TakesExam FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
        CONSTRAINT FK_QuestionID_TakesExam FOREIGN KEY (QuestionID) REFERENCES Questions(QuestionID),
        CONSTRAINT FK_ExamID_TakesExam FOREIGN KEY (ExamID) REFERENCES Exam(ExamID),
        CONSTRAINT CHK_Score CHECK (Score IN (0, 1))
);
```

DataBase Diagram:



2.4 Stored Procedures:



Examples:

```
Create PROCEDURE [dbo].[InsertStudent]
    @firstname VARCHAR(50), @lastname VARCHAR(50), @age int, @gender VARCHAR(6), @phone NCHAR(10),
       @location VARCHAR(255), @faculty VARCHAR(50), @gradutaionyear INT, @Email VARCHAR(50), @Password
VARCHAR(50),
       @branchname VARCHAR(50),@trackname VARCHAR(50)
AS
BEGIN
       SET NOCOUNT ON;
       DECLARE @trackid INT
       SELECT @trackid = TrackID FROM Track WHERE TrackName = @trackname

       DEClARE @BranchID INT
       SELECT @BranchID = BranchID FROM Branch WHERE BranchLocation = @branchname;


       INSERT INTO Student (FirstName, LastName, Age, Gender, Phone, [Location], Faculty,
GraduationYear, Email, [Password],BranchID, TrackID)
       VALUES (@firstname, @lastname, @age, @gender, @phone, @location, @faculty,
@gradutaionyear,@email, @Password,@BranchID ,@trackid)

       Declare @StudentID INT
       Set @StudentID = SCOPE_IDENTITY()

    INSERT INTO StudentCourse (StudentID, CourseID)
    SELECT @StudentID, CourseID
    FROM TrackCourse
    WHERE TrackID = @TrackID;
END;
```

```
CREATE PROCEDURE InsertBranch
    @BranchLocation Varchar(50),
    @BranchManager  Varchar(50) = NULL
with encryption
AS
BEGIN
    -- Check location already exists
    IF NOT EXISTS (SELECT BranchLocation FROM Branch WHERE BranchLocation = @BranchLocation)
    BEGIN
        -- Insert into branch table
        INSERT INTO Branch (BranchLocation, BranchManager)
        VALUES (@BranchLocation, @BranchManager);
    END
    ELSE
    BEGIN
        RAISERROR('Branch already exists.', 16, 1);
    END;
```

3-Data Transformation Process(SSIS)

Generating Data Using Mockaroo Website

- We Accessed the Mockaroo website (https://www.mockaroo.com/).

We Created a New Data Schema:

Defined Data Fields:

- For each entity in database schema, we defined the corresponding data field
- Specify the data types, formats, and any other constraints for each field.

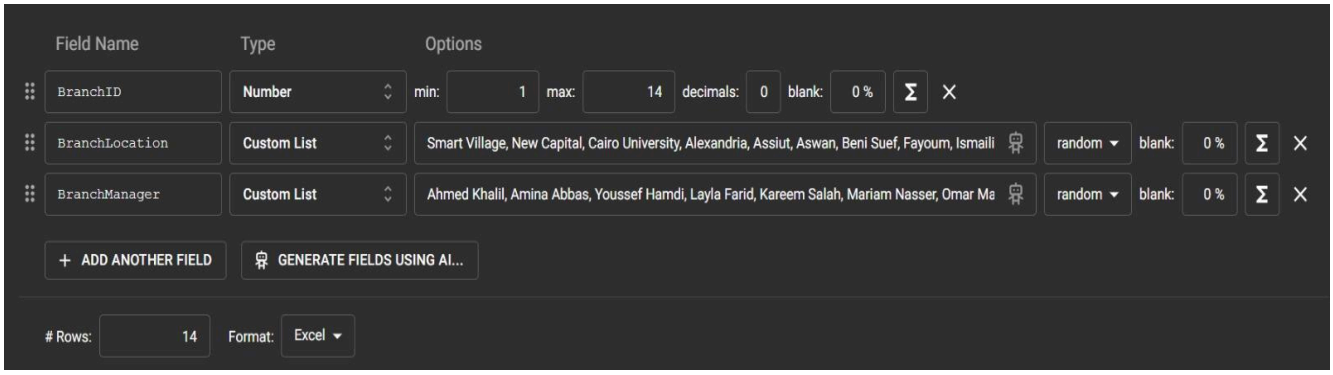
Generate Data:

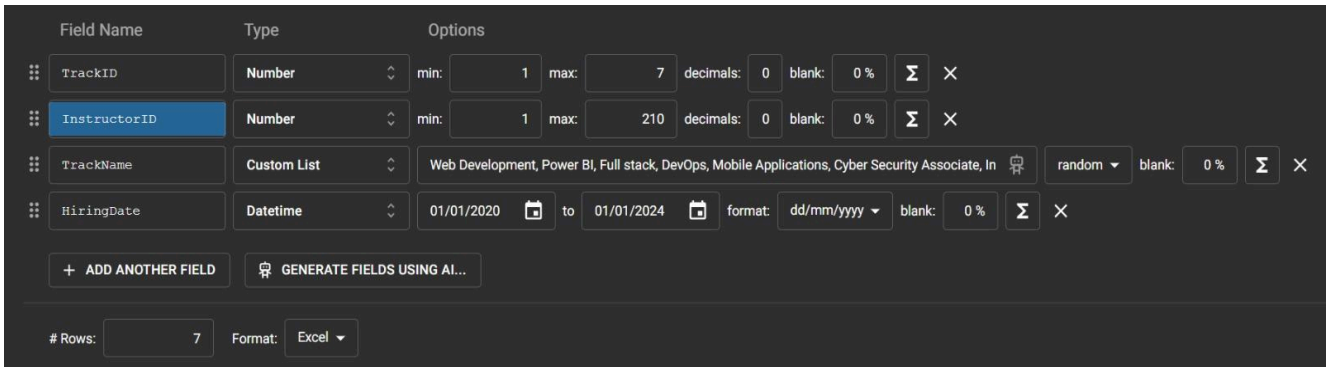
- After defining the data schema and fields, instruct Mockaroo to generate dummy data based specifications.
- We Adjusted the number of rows to generate as needed.

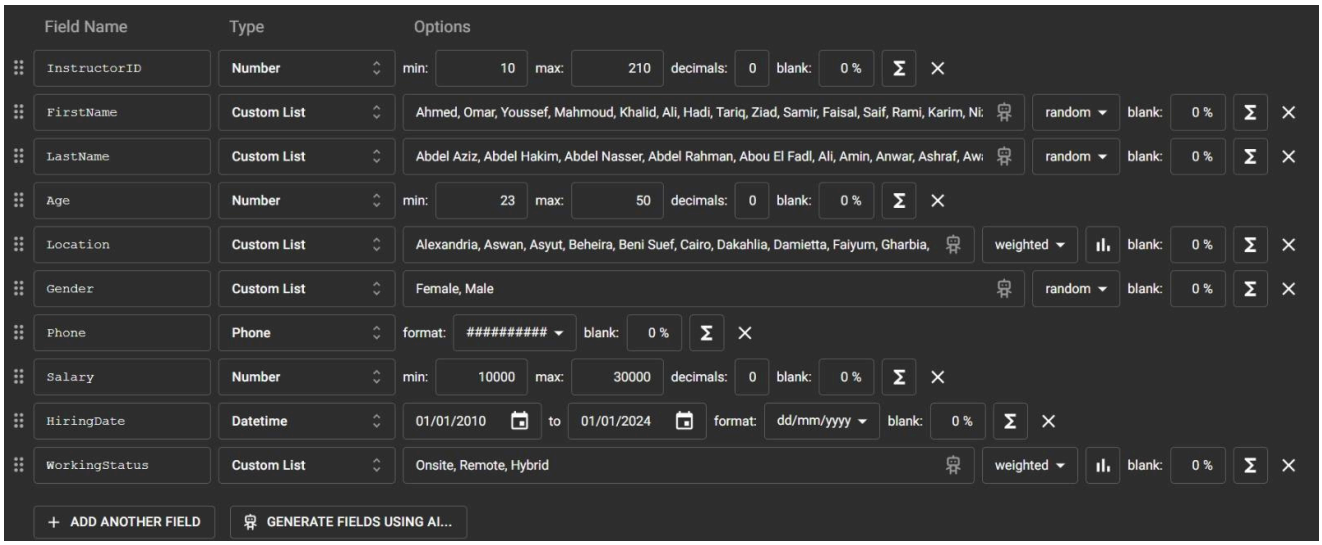
Download Data:

- After data generation is complete,we downloaded the generated data in Csv.

Examples:







ETL Using SSIS

We Created a New SSIS Project:

Added Data Flow Task:

- New Data Flow Task to represent the data transformation process.

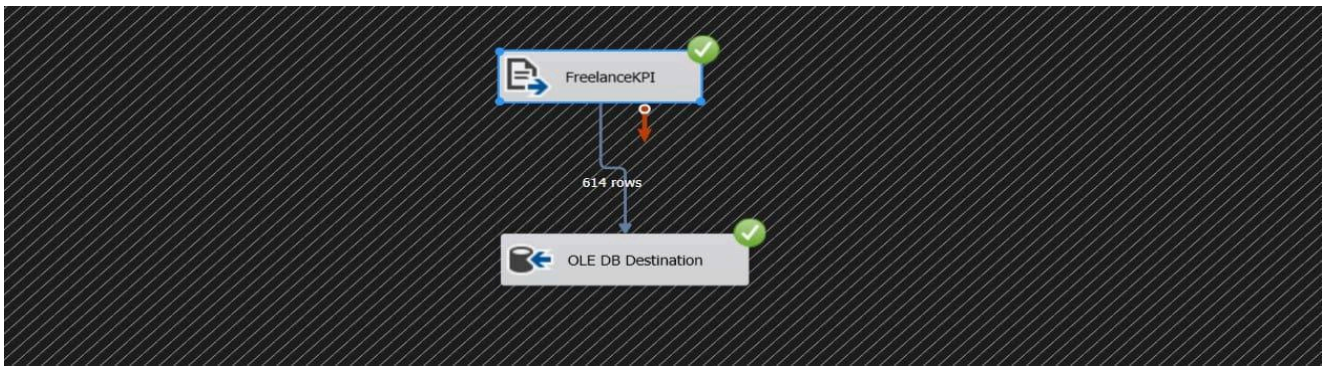
Configure Data Flow Components:

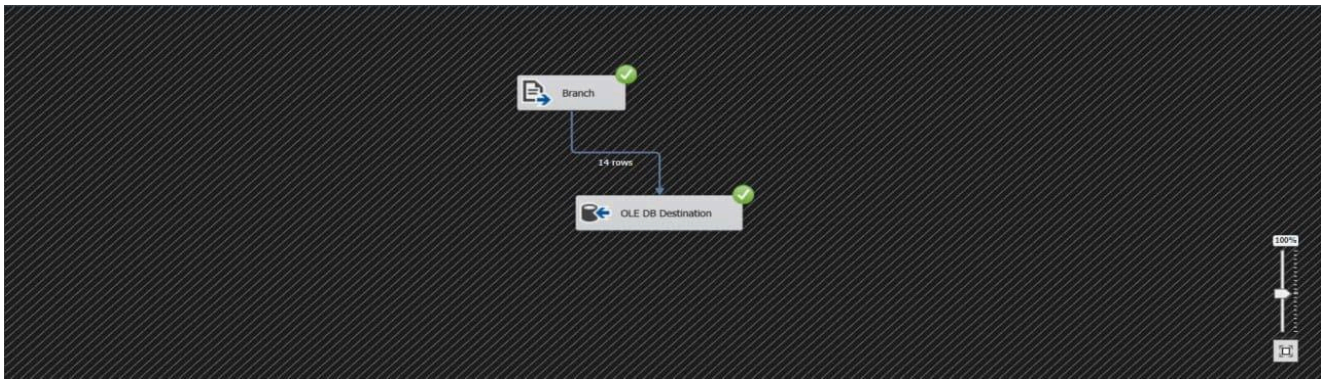
- Within the Data Flow Task, configure various components such as Source, Transformation, and Destination.
- We Used data source as CSV .
- We Specify the destination where transformed data will be loaded (SQL Server database)

Execute SSIS Package:

- Executed the SSIS package to perform the data transformation process.

Examples:





4- Reports Using SSRS:

Instrctor ID 19

1 of 1 100% Find | Ne

Instructor's Courses and Student Count

coursename	Count Students
Critical Thinking and Problem-Solving	983
Presentation Skills	983

student ID 360

1 of 1 100% Find | Next

The Grades of the Student in Each Course.

Student ID	Course Name	TOTALSCORE
360	Database Design and Management	70 %
360	Front-End Frameworks	50 %

EXAMID 1

1 of 1 100%

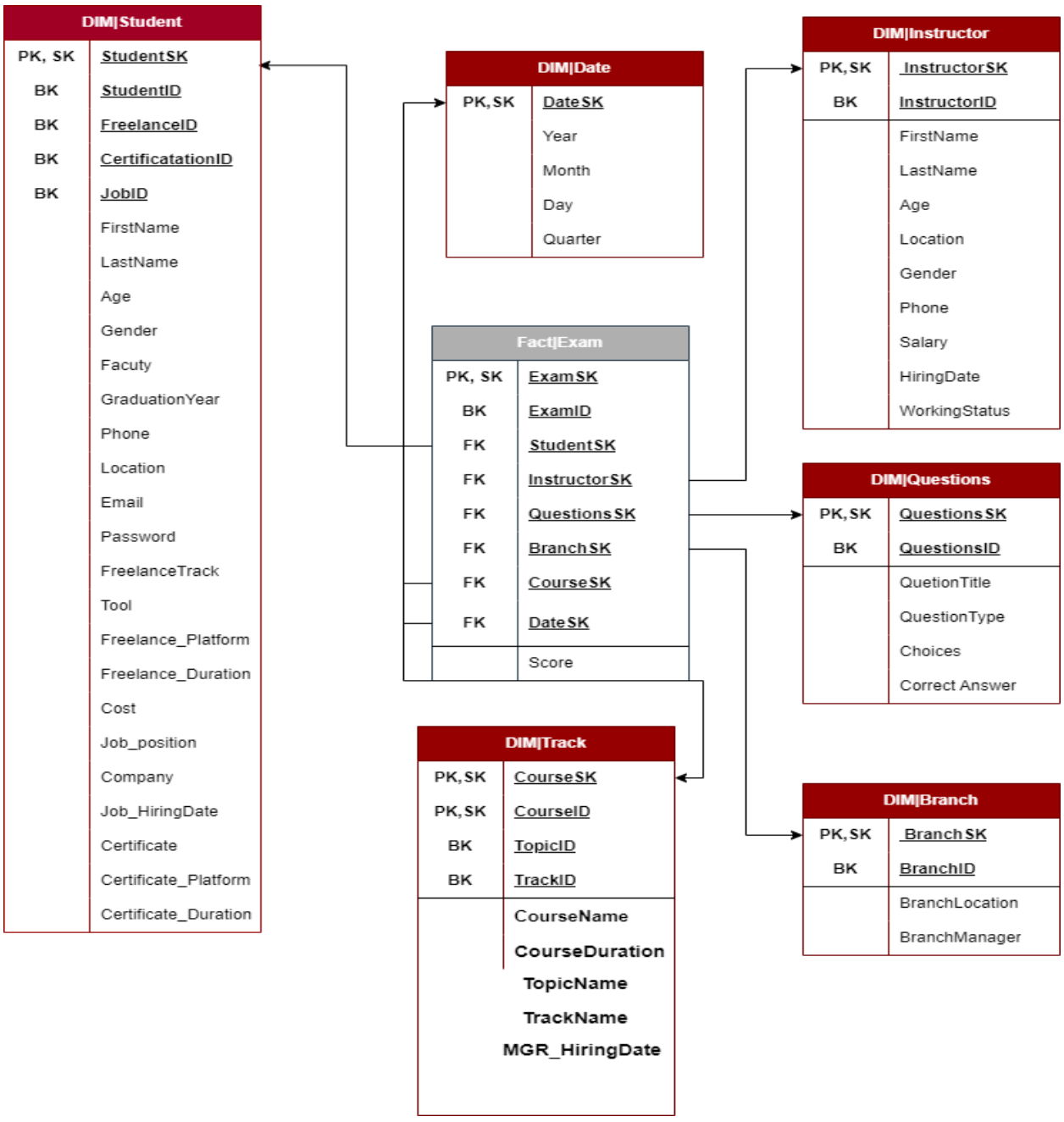
Exam Questions and Choices

Exam ID	Question Title	Choices
1	the ORDER BY clause is used to sort the result set in ascending order by default.	FALSE
1	the ORDER BY clause is used to sort the result set in ascending order by default.	TRUE
1	The SQL INSERT INTO statement is used to add new records into a database table.	FALSE
1	The SQL INSERT INTO statement is used to add new records into a database table.	TRUE
1	SQL stands for Structured Query Language.	FALSE
1	SQL stands for Structured Query Language.	TRUE
1	The SQL TRUNCATE TABLE statement removes all rows from a table without logging individual row deletions.	FALSE
1	The SQL TRUNCATE TABLE statement removes all rows from a table without logging individual row deletions.	TRUE
1	The SQL DELETE statement is used to remove table structure without deleting the data.	FALSE
1	The SQL DELETE statement is used to remove table structure without deleting the data.	TRUE
1	In SQL, the LIMIT clause is used to specify the number of columns to retrieve from a table.	FALSE

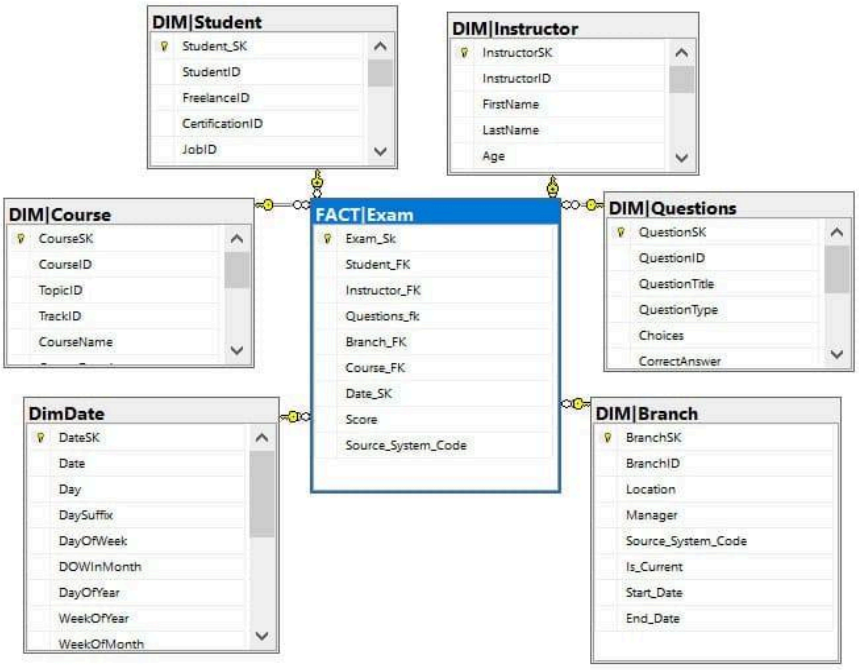
5-Data Warehouse:

5.1- DataModeling

- We Designed the dimensional model for the Data Warehouse, including fact tables, dimension tables, and their relationships.
- Identified key dimensions and measures to support reporting and analysis.



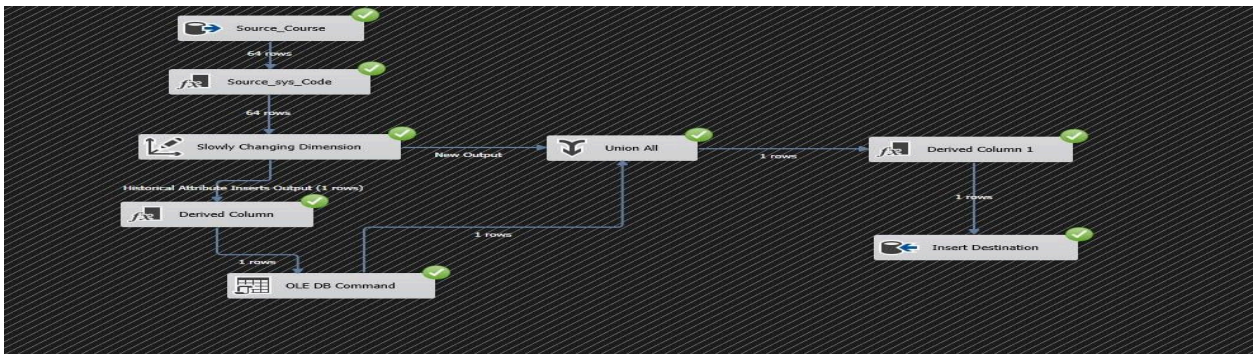
5.2-Create DWH Schema on SQL Server

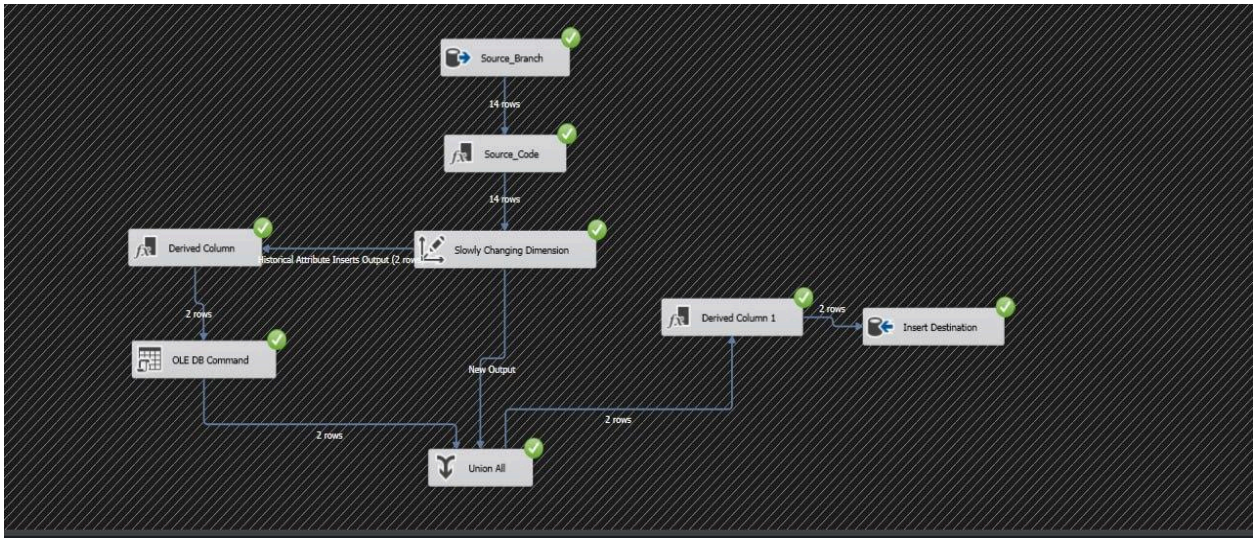


5.3-ETL Using SSIS:

- Identify the data source as OLTP DataBase from SQL SERVER to extract data for the Data Warehouse.
- Use SSIS Data Flow Tasks to extract data from OLTP database.
- Use Data Flow Tasks to transform the extracted data according to the dimensional model.
- Apply various transformations such as data cleansing, data type conversion, aggregation, and surrogate key generation.

Examples:

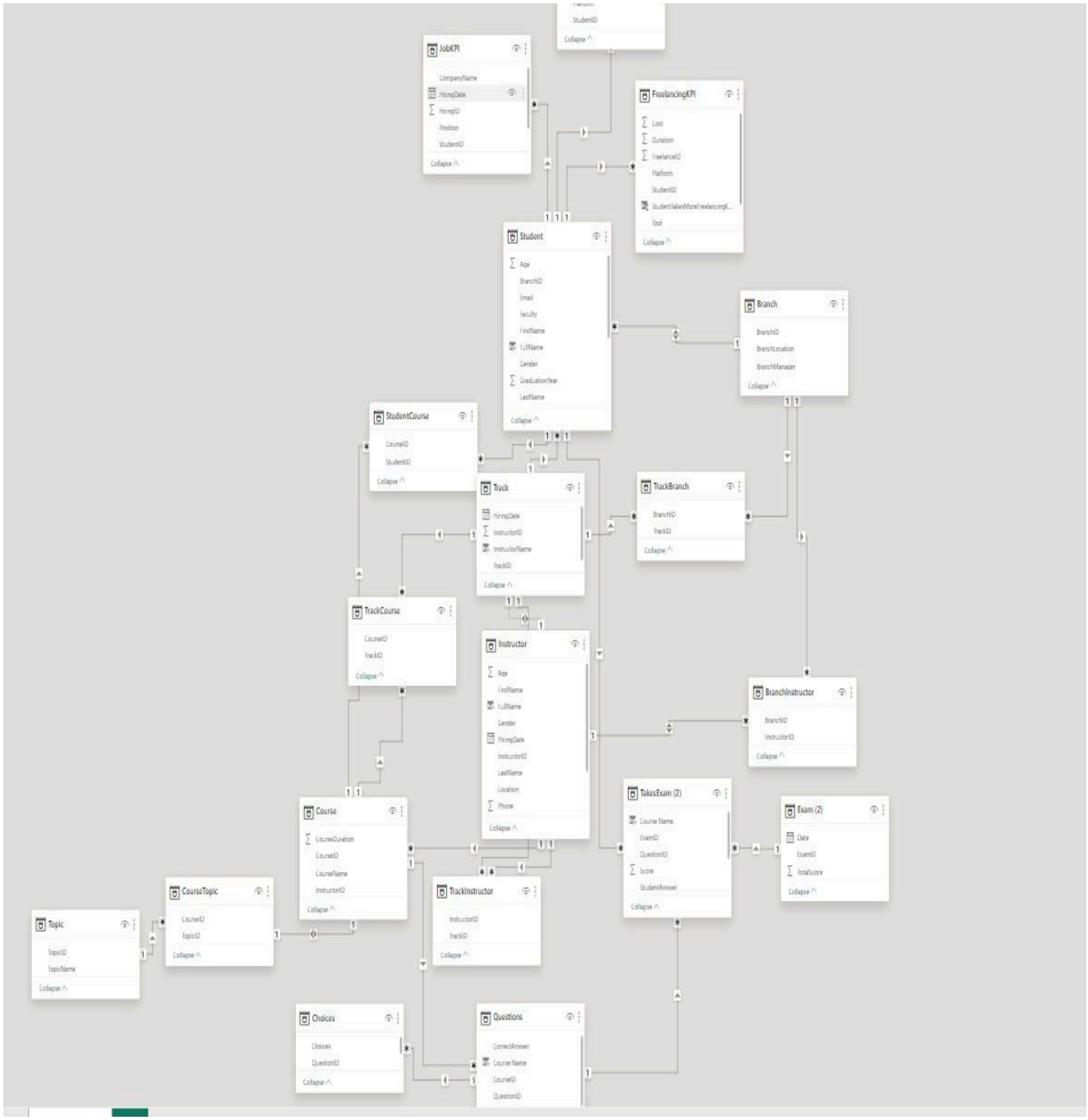



Source_Branch
14 rows
Source_Code
14 rows
Derived Column
Historical Attribute Inserts Output (2 rows)
Slowly Changing Dimension
2 rows
OLE DB Command
New Output
Derived Column 1
2 rows
Insert Destination
2 rows
2 rows
Union All

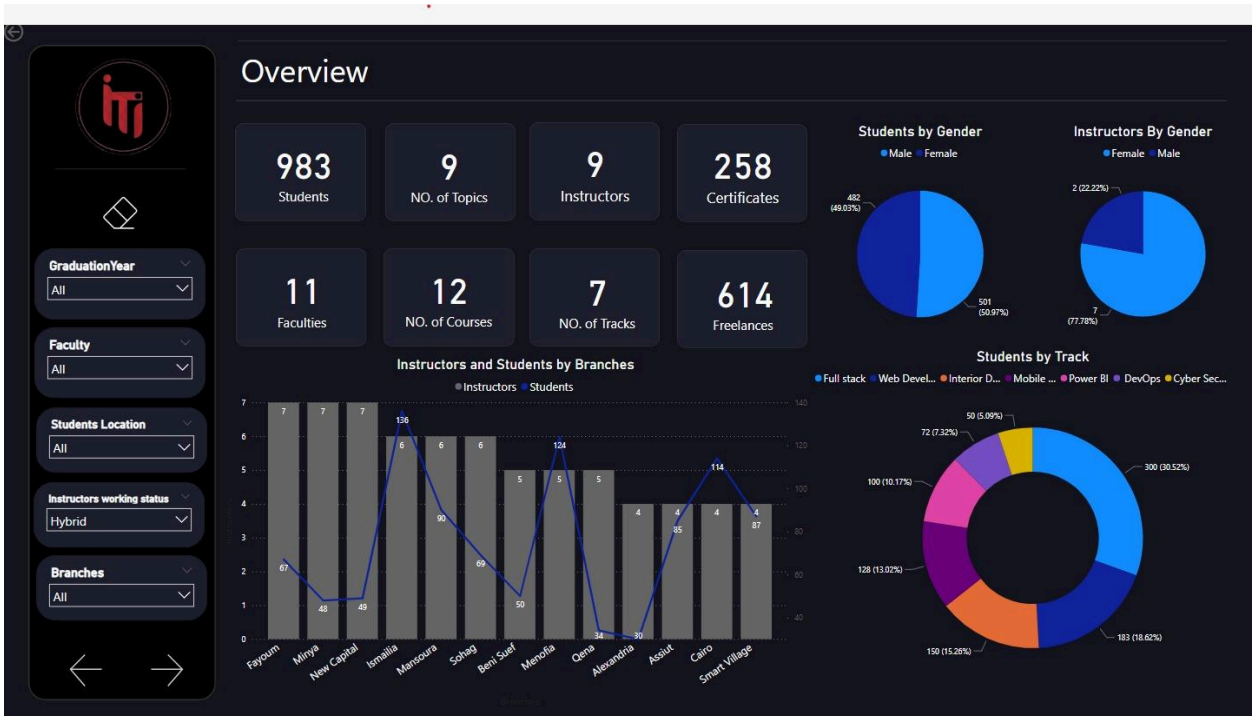

Source_fact
30 rows
Source_Sys_Code
30 rows
Date_Lookup
Lookup Match Output (30 rows)
Student_Lookup
Lookup Match Output (30 rows)
Branch_Lookup
Lookup Match Output (30 rows)
Course_Lookup
Lookup Match Output (30 rows)
Instructor_Lookup
Lookup Match Output (30 rows)
Questions_Lookup
Lookup Match Output (30 rows)
Dest_Fact

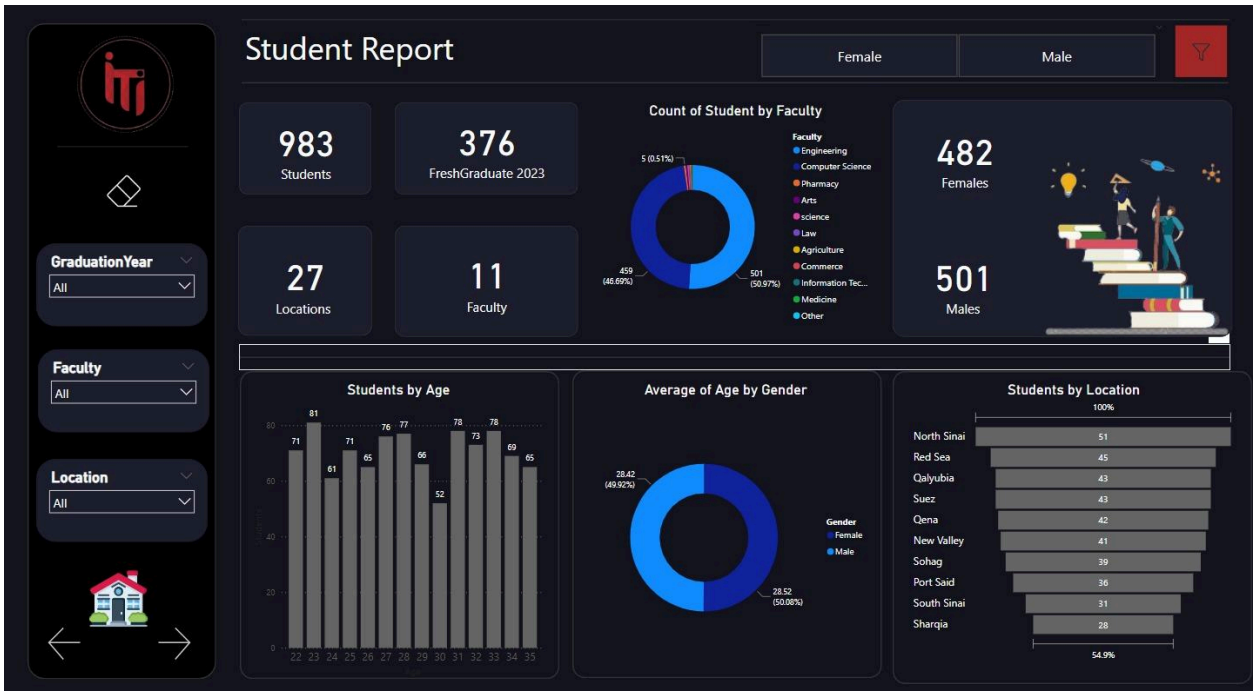
6-PowerBI Dashboards:

StarSchema On PowerBI

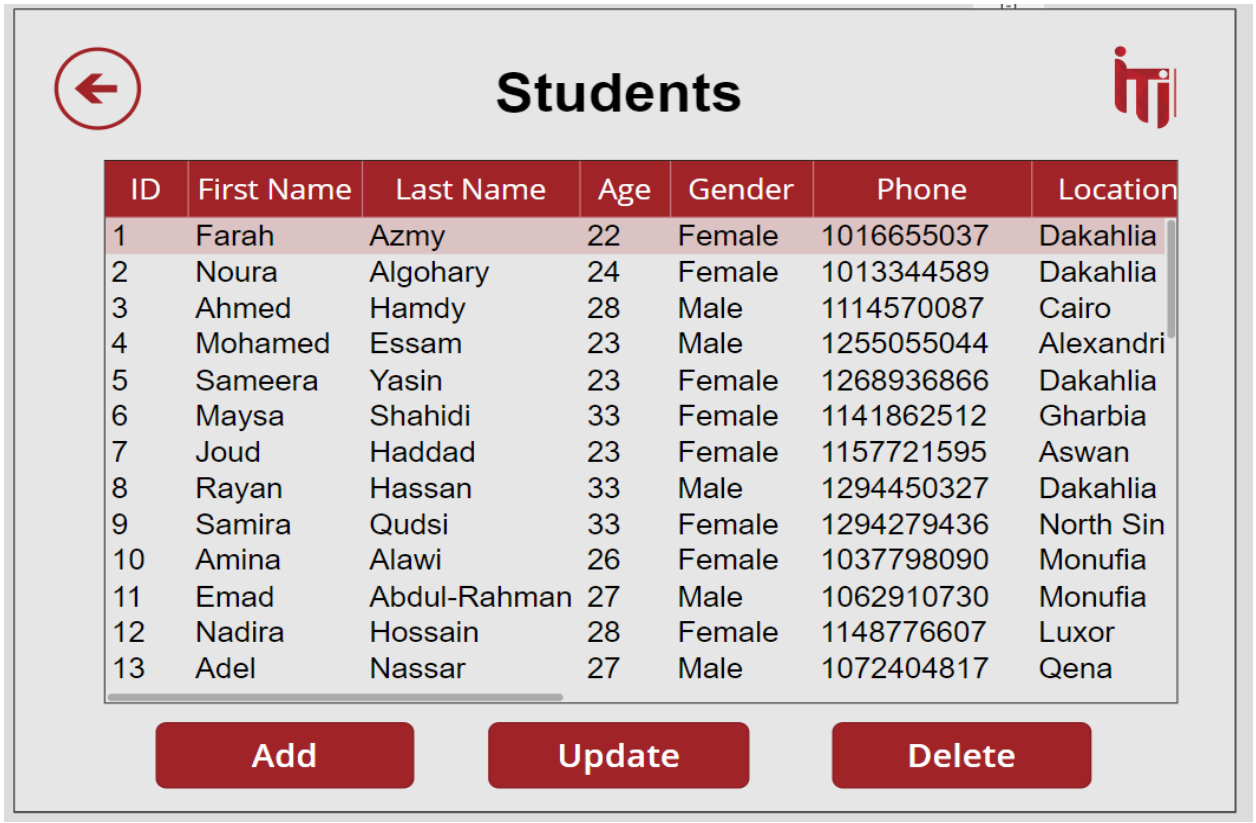


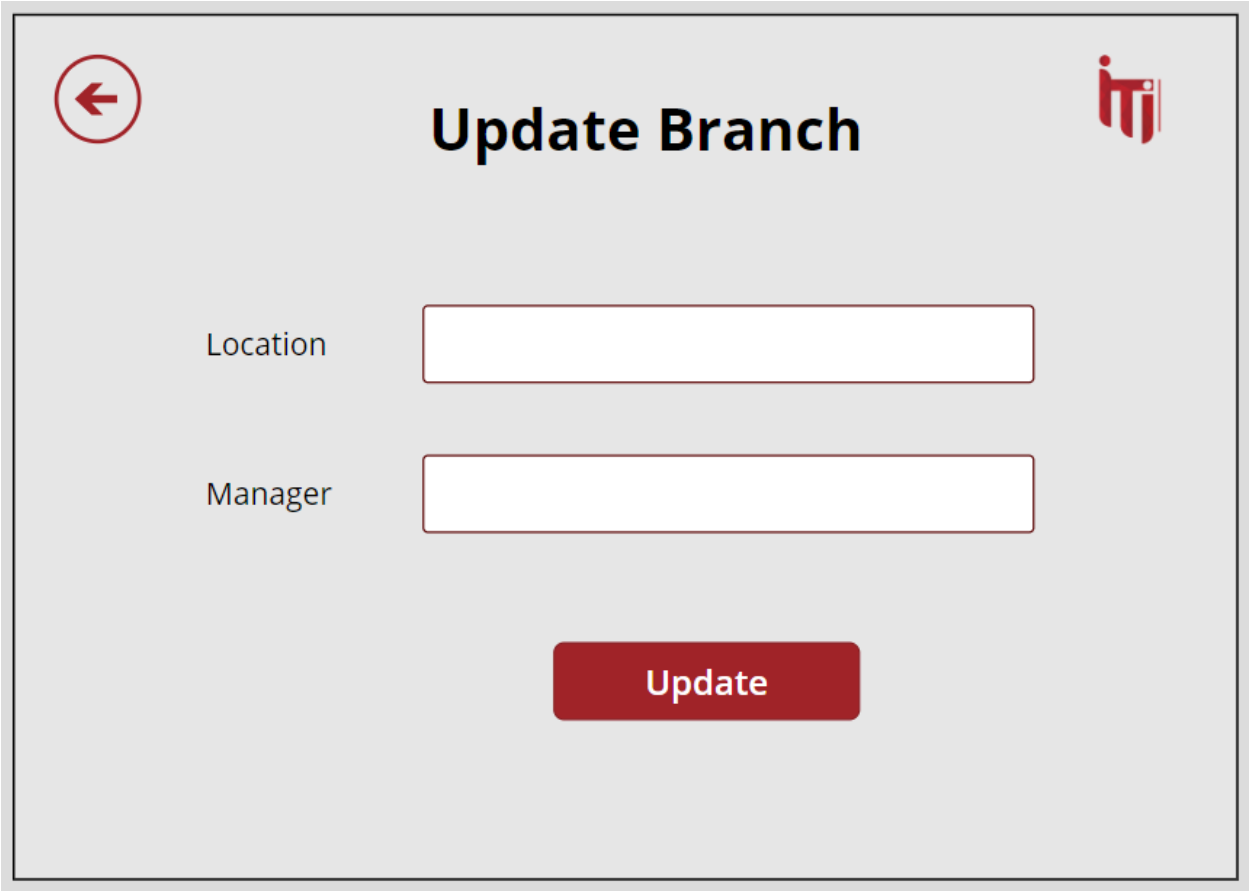
Dashboard Examples:

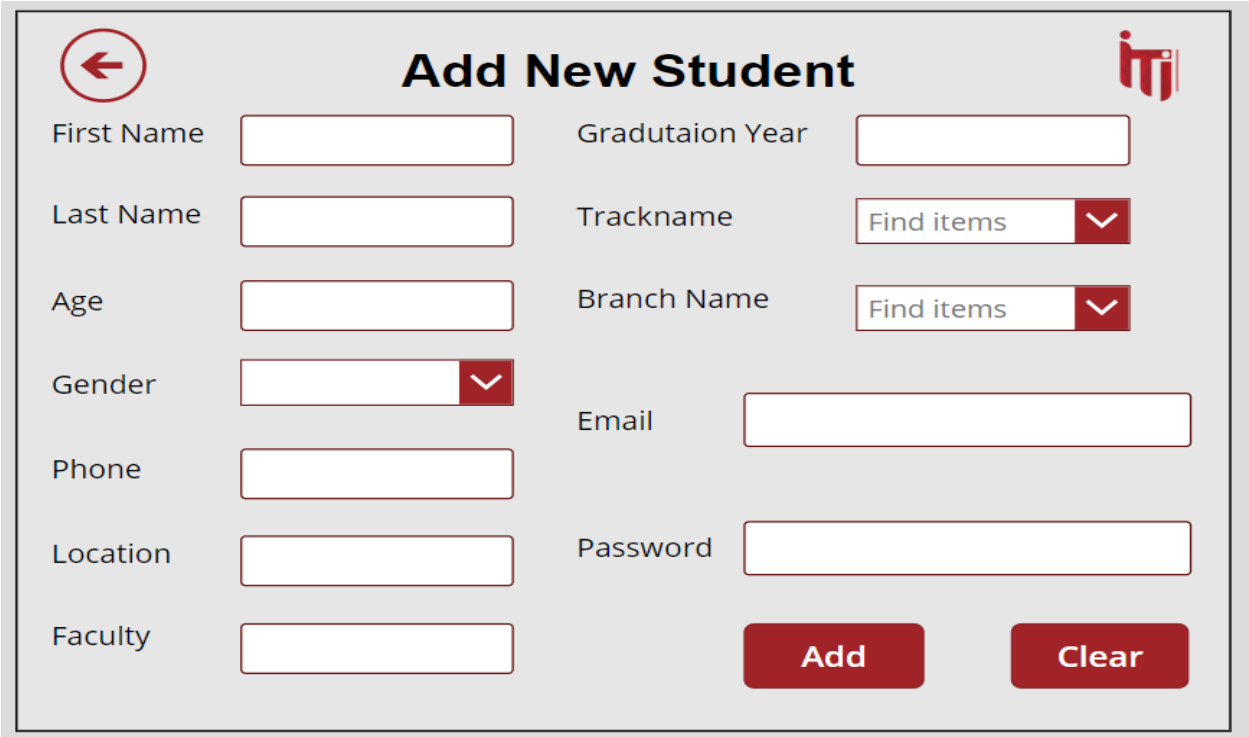



Student Report
Female
Male
983
Students
376
FreshGraduate 2023
27
Locations
11
Faculty
Count of Student by Faculty
482
Females
501
Males
GraduationYear
All
Faculty
All
Location
All
Students by Age
Average of Age by Gender
Students by Location
North Sinai
Red Sea
Qalyubia
Suez
Qena
New Valley
Sohag
Port Said
South Sinai
Sharqia

7-Power Apps:


Students
ID
First Name
Last Name
Age
Gender
Phone
Location
1 Farah Azmy 22 Female 1016655037 Dakahlia
2 Noura Algohary 24 Female 1013344589 Dakahlia
3 Ahmed Hamdy 28 Male 1114570087 Cairo
4 Mohamed Essam 23 Male 1255055044 Alexandri
5 Sameera Yasin 23 Female 1268936866 Dakahlia
6 Maysa Shahidi 33 Female 1141862512 Gharbia
7 Joud Haddad 23 Female 1157721595 Aswan
8 Rayan Hassan 33 Male 1294450327 Dakahlia
9 Samira Qudsi 33 Female 1294279436 North Sin
10 Amina Alawi 26 Female 1037798090 Monufia
11 Emad Abdul-Rahman 27 Male 1062910730 Monufia
12 Nadira Hossain 28 Female 1148776607 Luxor
13 Adel Nassar 27 Male 1072404817 Qena
Add
Update
Delete


Update Branch
Location
Manager
Update


Add New Student
First Name
Last Name
Age
Gender
Phone
Location
Faculty
Gradutaion Year
Trackname
Find items
Branch Name
Find items
Email
Password
Add
Clear

Admin HomePage


Students


Branches

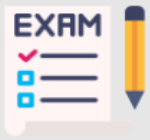

Courses


Instructors


Tracks

Student Portal


Information



Exams


Courses

Grades