**Test Views**

|  |  |  |
| --- | --- | --- |
| Queries | Result | Comment |
| SELECT \* FROM [dbo].[InstructorCourseView]; | Expected result | Done |
| SELECT \* FROM [dbo].[StudentView]; | Expected result | Done |
| SELECT \* FROM [Instructor].[ExamInstructorView]; | Expected result | Done |
| SELECT \* FROM [Instructor].[StudentCourseView]; | Expected result | Done |
| SELECT \* FROM [T\_Manager].[ClassDetailsView]; | Expected result | Done |
| SELECT \* FROM [T\_Manager].[CourseView]; | Expected result | Done |
| SELECT \* FROM [T\_Manager].[StudentCourseView]; | Expected result | Done |
| SELECT \* FROM [T\_Manager].[StudentExamView]; | Expected result | Done |

**Test Triggers**

|  |  |  |
| --- | --- | --- |
| Queries | Result | Comment |
| INSERT INTO Exam (Type\_Exam, Exam\_Date, exam\_StartTime, exam\_TotalDuration, TotalDegree, Crs\_Id, class\_Id, Ins\_Exam)  VALUES ('Midterm', '2024-02-15', '10:00:00', 120, 100, 1, 1, 1); | Expected result | Done |
| UPDATE [dbo].[StudentExam] SET StudentAnswer = 'New Answer' WHERE ExamID = 1 AND QuestionID = 'Q1' AND StudentID = 1; | Expected result | Done |
| INSERT INTO [dbo].[StudentExam] (ExamID, QuestionID, StudentID, StudentAnswer, Result)  VALUES (1, 'Q2', 2, 'True', 1); | Expected result | Done |
| UPDATE [dbo].[StudentExam] SET Result = 2 WHERE ExamID = 1 AND QuestionID = 'Q2' AND StudentID = 2; | Expected result | Done |
| INSERT INTO ExamQuestion (ExamID, QuestionID, Degree)  VALUES (1, 'Q3', 10); | Expected result | Done |
| INSERT INTO [StudentCourse] (StudentID, CourseID, TotalDegree, FinalResult)  VALUES (1, 1, 80, NULL); | Expected result | Done |
| UPDATE [StudentCourse] SET TotalDegree = 40 WHERE StudentID = 1 AND CourseID = 1; | Expected result | Done |

**Test Functions**

|  |  |  |
| --- | --- | --- |
| Queries | Result | Comment |
| SELECT \* FROM dbo.InstructorDataOrderedBy\_Function('InstructorName'); | Expected result | Done |
| SELECT \* FROM Instructor.InstructorSearchByPattern\_FN('InstructorName', 'John'); | Expected result | Done |
| SELECT \* FROM Student.StdCourseInfoByStudentID\_FN(1); | Expected result | Done |
| SELECT \* FROM T\_Manager.InstructorSearchByPattern\_FN('InstructorName', 'John'); | Expected result | Done |
| SELECT \* FROM T\_Manager.ManagerSearchByPattern\_FN('StudentName', 'John'); | Expected result | Done |
| SELECT \* FROM T\_Manager.SearchByPatternStdTable\_FN('Name', 'John'); | Expected result | Done |

**Test Procedures**

|  |  |  |
| --- | --- | --- |
| Queries | Result | Comment |
| -- Declare variables  DECLARE @InstructorID INT = 2;  DECLARE @CourseID INT = 1;  DECLARE @RandomSelection VARCHAR(15) = 'Random';  DECLARE @ExamID INT = 1;  DECLARE @NumberOfRandomQuestion INT = 3;  -- Declare a table variable for manual questions  DECLARE @QuestionDegrees QuestionDegreesType;  INSERT INTO @QuestionDegrees (QuestionID, QuestionDegree)  VALUES ('Q1', 50), ('Q2', 100), ('Q3', 30);  -- Execute the stored procedure  EXEC [Instructor].[AddQuestions\_Proc]  @InstructorID,  @CourseID,  @RandomSelection,  @ExamID,  @NumberOfRandomQuestion,  @QuestionDegrees; | Expected result | Done |
| -- Declare variables  DECLARE @InstructorID INT = 1;  DECLARE @StudentIDs NVARCHAR(MAX) = '1,2,3';  DECLARE @ExamID INT = 1;  -- Execute the stored procedure  EXEC [Instructor].[AddStudentsToExam]  @InstructorID,  @StudentIDs,  @ExamID; | Expected result | Done |
| -- Declare variables  DECLARE @Type NVARCHAR(50) = 'Midterm';  DECLARE @ExamDate DATE = '2024-02-20';  DECLARE @StartTime NVARCHAR(8) = '10:00:00';  DECLARE @TotalTime INT = 120;  DECLARE @TotalDegree INT = 100;  DECLARE @Crs\_Id INT = 1;  DECLARE @Class\_Id INT = 1;  DECLARE @InstructorId INT = 1;  -- Execute the stored procedure  EXEC [Instructor].[CreateExam]  @Type,  @ExamDate,  @StartTime,  @TotalTime,  @TotalDegree,  @Crs\_Id,  @Class\_Id,  @InstructorId; | Expected result | Done |
| -- Declare variables  DECLARE @year INT = 2022;  DECLARE @courseId INT = 1;  DECLARE @instructorId INT = 1;  -- Execute the stored procedure  EXEC [Instructor].[GetExamsByYearCourseInstructor]  @year,  @courseId,  @instructorId; | Expected result | Done |
| -- Declare variables  DECLARE @option1 NVARCHAR(100) = 'Age';  -- Execute the stored procedure  EXEC [Instructor].[OrderBYStd\_Proc]  @option1; | Expected result | Done |
| -- Declare variables  DECLARE @OrderByColumn VARCHAR(100) = 'Name';  -- Execute the stored procedure  EXEC [Instructor].[StudentDataOrderedBy\_Proc]  @OrderByColumn; | Expected result | Done |
| -- Declare variables  DECLARE @std\_id INT = 1;  DECLARE @exam\_id INT = 1;  -- Execute the stored procedure  EXEC [Instructor].[UpdateResults]  @std\_id,  @exam\_id; | Expected result | Done |
| -- Declare variables  DECLARE @std\_id INT = 1;  DECLARE @exam\_id INT = 1;  -- Declare a table variable for student answers  DECLARE @student\_answers AnswerTableType;  INSERT INTO @student\_answers (QuestionID, StudentAnswer)  VALUES  ('Q1', 'Option A'),  ('Q2', 'True'),  ('Q3', 'Oxygen');  -- Execute the stored procedure  EXEC [Student].[StoreStudentAnswers]  @std\_id,  @exam\_id,  @student\_answers; | Expected result | Done |
| -- Declare variables  DECLARE @Name NVARCHAR(255) = 'NewUser';  DECLARE @Password NVARCHAR(255) = 's123';  DECLARE @UserType NVARCHAR(255) = 'Student';  -- Execute the stored procedure  EXEC [T\_Manager].[CreateUserLogin]  @Name,  @Password,  @UserType; | Expected result | Done |
| -- Declare variables  DECLARE @CourseID INT = 1;  -- Execute the stored procedure  EXEC [T\_Manager].[crs\_std\_inst\_INFO\_by\_course\_id]  @CourseID; | Expected result | Done |
| -- Declare variables  DECLARE @OrderByColumn NVARCHAR(MAX) = 'InstructorName';  -- Execute the stored procedure  EXEC [T\_Manager].[InstructorDataOrderedBy\_Proc]  @OrderByColumn; | Expected result | Done |
| -- Declare variables  DECLARE @Option1 NVARCHAR(MAX) = 'StudentName';  -- Execute the stored procedure  EXEC [T\_Manager].[MangerDataOrderedBy\_Proc]  @Option1; | Expected result | Done |
| -- Declare variables  DECLARE @option1 VARCHAR(100) = 'Name';  -- Execute the stored procedure  EXEC [T\_Manager].[OrderBYStd\_Proc]  @option1; | Expected result | Done |
| -- Declare variables  DECLARE @inputYear INT = 2022;  -- Execute the stored procedure  EXEC [T\_Manager].[ShowDataByYear]  @inputYear; | Expected result | Done |
| -- Declare variables  DECLARE @intake VARCHAR(50) = '2025';  DECLARE @id INT = 1;  -- Execute the stored procedure  EXEC [T\_Manager].[UpdateYearOnIntakeInsert]  @intake,  @id; | Expected result | Done |