irDevelopers.com [irDevelopers.com]

Examination System Database project

Table of contents

Legend

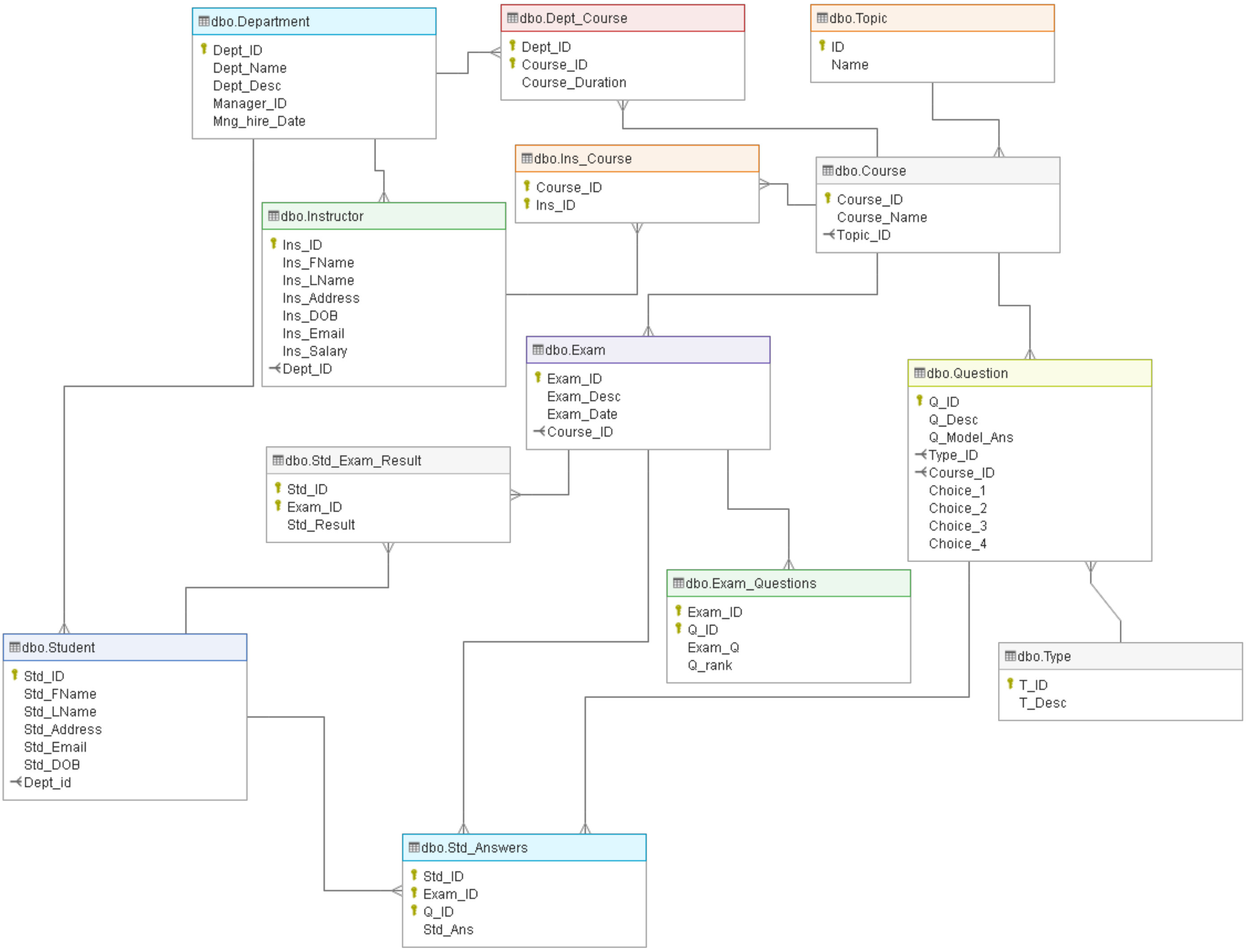
- 🔑 Primary key
- 🔑 Primary key disabled
- 🔑 User-defined primary key
- 🔑 Unique key
- 🔑 Unique key disabled
- 🔑 User-defined unique key
- ⚡ Active trigger
- ⚡ Disabled trigger
- >— Many to one relation
- >— User-defined many to one relation
- —< One to many relation
- —< User-defined one to many relation
- — One to one relation
- — User-defined one to one relation
- →@ Input
- @→ Output
- →@→ Input/Output
- 🗔 Uses dependency
- 🗔 User-defined uses dependency
- 🗔 Used by dependency
- 🗔 User-defined used by dependency

Database examination system

This project is designed for an examination system that takes place using random questions in differenet courses that are being taught by multiple instructors to multiple students in differenet departments.

It also generates for the students random exams and they answer them online, then we take their answers, correct their exams and give them grades related to each subject they have had exams in it.

1. ERD



The diagram shows relations between all the enitities of the project and the connection between them and column names.

2. Other

2.1. Tables

2.1.1. Table: dbo.Course

Status: Active

The course plays an important role in the examination system as it is connected to more than one table such as the department, Exams, instructors, topic & questions.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Course_ID	int	Identity / Auto increment
▦		Course_Name	nvarchar(50)	Nullable
▦		Topic_ID	int	Nullable References: dbo.Topic

Links to

	Table	Join	Title / Name / Description
→	dbo.Topic	dbo.Course.Topic_ID = dbo.Topic.ID	FK_Courses_Topic

Linked from

	Table	Join	Title / Name / Description
←	dbo.Dept_Course	dbo.Course.Course_ID = dbo.Dept_Course.Course_ID	FK_Dept_Course_Course
←	dbo.Exam	dbo.Course.Course_ID = dbo.Exam.Course_ID	FK_Exams_Courses
←	dbo.Ins_Course	dbo.Course.Course_ID = dbo.Ins_Course.Course_ID	FK_Ins_Course_Course
←	dbo.Question	dbo.Course.Course_ID = dbo.Question.Course_ID	FK_Question_Course

Unique keys

	Columns	Name / Description
🔑	Course_ID	PK_Courses

Uses

Name
▦ dbo.Course
→ dbo.Topic

Used By

Name
▦ dbo.Course
⚙ dbo.INSERTINTOCOURSES
⚙ dbo.SP_DELETECOURSE
⚙ dbo.Sp_GenerateExam

Name
dbo.SP_INSERTDEPARTMENTCOURSE
dbo.SP_INSERTINSCOURSES
dbo.SP_INSERTINTOEXAM
dbo.SP_INSERTQUESTION
dbo.SP_RP_COURSETOPIC
dbo.SP_RP_INSTRUCTORCOURSESWITHSTUDENT
dbo.SP_SELECTALLCOURSESINDEPARTMENT
dbo.SP_SELECTFROMCOURSE
dbo.SP_SELECTINSCOURSE
dbo.SP_SELECTQUESTION
dbo.SP_SELECTSTUDENTRESULTS
dbo.SP_UPDATECOURSE
dbo.Dept_Course
dbo.Exam
dbo.Ins_Course
dbo.Question

2.1.2. Table: dbo.Department

Status: Active

The Department is the connection between the instructors and the students.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Dept_ID	int	Identity / Auto increment
▦		Dept_Name	nvarchar(50)	
▦		Dept_Desc	nvarchar(50)	Nullable
▦		Manager_ID	int	Nullable
▦		Mng_hire_Date	date	Nullable

Linked from

	Table	Join	Title / Name / Description
—<	dbo.Dept_Course	dbo.Department.Dept_ID = dbo.Dept_Course.Dept_ID	FK_Dept_Course_Department
—<	dbo.Instructor	dbo.Department.Dept_ID = dbo.Instructor.Dept_ID	FK_Instructor_Department
—<	dbo.Student	dbo.Department.Dept_ID = dbo.Student.Dept_id	FK_Student_Department

Unique keys

	Columns	Name / Description
🔑	Dept_ID	PK_Department

Used By

Name
▦ dbo.Department
⚙ dbo.INSERTINTODEP
⚙ dbo.SP_INSERTDEPARTMENTCOURSE
⚙ dbo.SP_INSERTINTOINST
⚙ dbo.SP_INSERTINTOSTD
⚙ dbo.SP_RP_INSTRUCTORCOURSESWITHSTUDENT
⚙ dbo.SP_RP_STDINF
⚙ dbo.SP_SELECTALLCOURSESINDEPARTMENT
⚙ dbo.SP_SELECTFROMDEP
⚙ dbo.SP_UPDATEDEP
⚙ dbo.SP_UPDATEDEPMANAGER
—< dbo.Dept_Course
—< dbo.Instructor
—< dbo.Student

2.1.3. Table: dbo.Dept_Course

Status: Active

The table illustrates the relation between the Department and the course.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Dept_ID	int	**References**: dbo.Department
▦	🔑	Course_ID	int	**References**: dbo.Course
▦		Course_Duration	nvarchar(50)	**Nullable**

Links to

	Table	Join	Title / Name / Description
⤚	dbo.Course	**dbo.Dept_Course.**Course_ID = dbo.Course.Course_ID	FK_Dept_Course_Course
⤚	dbo.Department	**dbo.Dept_Course.**Dept_ID = dbo.Department.Dept_ID	FK_Dept_Course_Department

Unique keys

	Columns	Name / Description
🔑	Dept_ID, Course_ID	PK_Dept_Course

Uses

Name
▦ dbo.Dept_Course
⤚ dbo.Course
⤚ dbo.Department

Used By

Name
▦ dbo.Dept_Course
⚙ dbo.SP_DELETEDEPARTMENTCOURSE
⚙ dbo.SP_INSERTDEPARTMENTCOURSE
⚙ dbo.SP_SELECTALLCOURSESINDEPARTMENT
⚙ dbo.SP_UPDATEDEPTCOURSESDURATION

2.1.4. Table: dbo.Exam

Status: Active

The exam is connected to the course, questions and the student.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Exam_ID	int	Identity / Auto increment
▦		Exam_Desc	nvarchar(50)	Nullable
▦		Exam_Date	date	Nullable
▦		Course_ID	int	References: dbo.Course

Links to

	Table	Join	Title / Name / Description
≻—	dbo.Course	dbo.Exam.Course_ID = dbo.Course.Course_ID	FK_Exams_Courses

Linked from

	Table	Join	Title / Name / Description
—≺	dbo.Exam_Questions	dbo.Exam.Exam_ID = dbo.Exam_Questions.Exam_ID	FK_Exam_Questions_Exam
—≺	dbo.Std_Answers	dbo.Exam.Exam_ID = dbo.Std_Answers.Exam_ID	FK_Std_Answers_Exam
—≺	dbo.Std_Exam_Result	dbo.Exam.Exam_ID = dbo.Std_Exam_Result.Exam_ID	FK_Std_Exam_Result_Exam

Unique keys

	Columns	Name / Description
🔑	Exam_ID	PK_Exams

Uses

Name
▦ dbo.Exam
≻— dbo.Course

Used By

Name
▦ dbo.Exam
⚙ dbo.SP_DELETEEXAM
⚙ dbo.Sp_GenerateExam
⚙ dbo.SP_INSERTINTOEXAM
⚙ dbo.SP_RP_SELECTEXAMQUESTIONWITHCHOICES
⚙ dbo.SP_RP_SELECTSTDANSWERS
⚙ dbo.SP_SELECTEXAMQUESTION
⚙ dbo.SP_SELECTFROMEXAM

Name
dbo.SP_SELECTSTDANSWERS
dbo.SP_SELECTSTUDENTRESULTS
dbo.SP_UPDATEXAMDATE
dbo.Exam_Questions
dbo.Std_Answers
dbo.Std_Exam_Result

2.1.5. Table: dbo.Exam_Questions

Status: Active

This relation illustrates the connection between exam and the question.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Exam_ID	int	**References**: dbo.Exam
▦	🔑	Q_ID	int	
▦		Exam_Q	nvarchar(250)	
▦		Q_rank	int	**Nullable**

Links to

	Table	Join	Title / Name / Description
≻—	dbo.Exam	**dbo.Exam_Questions**.Exam_ID = dbo.Exam.Exam_ID	FK_Exam_Questions_Exam

Unique keys

	Columns	Name / Description
🔑	Exam_ID, Q_ID	PK_Exam_Questions

Uses

Name
▦ **dbo.Exam_Questions**
≻— dbo.Exam

Used By

Name
▦ **dbo.Exam_Questions**
⚙ dbo.Sp_GenerateExam
⚙ dbo.SP_RP_SELECTEXAMQUESTIONWITHCHOICES
⚙ dbo.SP_SELECTEXAMQUESTION
⚙ dbo.SP_UPDATEEXAMQUESTION

2.1.6. Table: dbo.Ins_Course

Status: Active

This relation illustrates the connection between the instructor and the course.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Course_ID	int	**References**: dbo.Course
▦	🔑	Ins_ID	int	**References**: dbo.Instructor

Links to

	Table	Join	Title / Name / Description
≻	dbo.Course	**dbo.Ins_Course**.Course_ID = dbo.Course.Course_ID	FK_Ins_Course_Course
≻	dbo.Instructor	**dbo.Ins_Course**.Ins_ID = dbo.Instructor.Ins_ID	FK_Ins_Course_Instructor

Unique keys

	Columns	Name / Description
🔑	Course_ID, Ins_ID	PK_Ins_Course

Uses

Name
▦ dbo.Ins_Course
≻ dbo.Course
≻ dbo.Instructor

Used By

Name
▦ dbo.Ins_Course
⚙ dbo.SP_DELETEINSCOURSE
⚙ dbo.SP_INSERTINSCOURSES
⚙ dbo.SP_RP_INSTRUCTORCOURSESWITHSTUDENT
⚙ dbo.SP_SELECTINSCOURSE
⚙ dbo.SP_UPDATEINSCOURSE

2.1.7. Table: dbo.Instructor

Status: Active

The instructor has relation between the department and the course.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Ins_ID	int	Identity / Auto increment
▦		Ins_FName	nvarchar(50)	Nullable
▦		Ins_LName	nvarchar(50)	Nullable
▦		Ins_Address	nvarchar(50)	Nullable
▦		Ins_DOB	date	Nullable
▦		Ins_Email	nvarchar(50)	Nullable
▦		Ins_Salary	int	Nullable
▦		Dept_ID	int	Nullable References: dbo.Department

Links to

	Table	Join	Title / Name / Description
→	dbo.Department	dbo.Instructor.Dept_ID = dbo.Department.Dept_ID	FK_Instructor_Department

Linked from

	Table	Join	Title / Name / Description
←	dbo.Ins_Course	dbo.Instructor.Ins_ID = dbo.Ins_Course.Ins_ID	FK_Ins_Course_Instructor

Unique keys

	Columns	Name / Description
🔑	Ins_ID	PK_Instructor

Uses

Name
▦ dbo.Instructor
→ dbo.Department

Used By

Name
▦ dbo.Instructor
⚙ dbo.INSERTINTODEP
⚙ dbo.SP_DELETEINSTRUCTOR
⚙ dbo.SP_INSERTINSCOURSES
⚙ dbo.SP_INSERTINTOINST
⚙ dbo.SP_RP_INSTRUCTORCOURSESWITHSTUDENT

Name
dbo.SP_SELECTFROMINST
dbo.SP_SELECTINSCOURSE
dbo.SP_UPDATEDEPMANAGER
dbo.SP_UPDATEINSCOURSE
dbo.SP_UPDATEINSTRACTOR
dbo.Ins_Course

2.1.8. Table: dbo.Question

Status: Active

The question inside each exam has 2 types either MCQ or true/false, they are genereated depending on the course and the exam number.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Q_ID	int	Identity / Auto increment
▦		Q_Desc	nvarchar(250)	Nullable
▦		Q_Model_Ans	char(1)	Nullable
▦		Type_ID	int	References: dbo.Type
▦		Course_ID	int	References: dbo.Course
▦		Choice_1	char(250)	Nullable
▦		Choice_2	char(250)	Nullable
▦		Choice_3	char(250)	Nullable
▦		Choice_4	char(250)	Nullable

Links to

	Table	Join	Title / Name / Description
≻—	dbo.Course	dbo.Question.Course_ID = dbo.Course.Course_ID	FK_Question_Course
≻—	dbo.Type	dbo.Question.Type_ID = dbo.Type.T_ID	FK_Question_Type

Linked from

	Table	Join	Title / Name / Description
—≺	dbo.Std_Answers	dbo.Question.Q_ID = dbo.Std_Answers.Q_ID	FK_Std_Answers_Question

Unique keys

	Columns	Name / Description
🔑	Q_ID	PK_Question

Uses

Name
▦ dbo.Question
≻— dbo.Course
≻— dbo.Type

Used By

Name
▦ dbo.Question
⚙ dbo.SP_DELETEFROMQUESTION
⚙ dbo.Sp_GenerateExam

Name
dbo.SP_INSERTQUESTION
dbo.SP_RP_SELECTEXAMQUESTIONWITHCHOICES
dbo.SP_RP_SELECTSTDANSWERS
dbo.SP_SELECTEXAMQUESTION
dbo.SP_SELECTQUESTION
dbo.SP_SELECTSTDANSWERS
dbo.SP_UBDATEQUESTIONTYPE_OR_QUESTION
dbo.SP_UPDATEEXAMQUESTION
dbo.Std_Answers

2.1.9. Table: dbo.Std_Answers

Status: Active

This table illustrates the relation between the student, the exam and the questions inside the axam.

Columns

		Name	Data type	Description / Attributes
▦	🔑	Std_ID	int	**References**: dbo.Student
▦	🔑	Exam_ID	int	**References**: dbo.Exam
▦	🔑	Q_ID	int	**References**: dbo.Question
▦		Std_Ans	char(1)	

Links to

	Table	Join	Title / Name / Description
≻	dbo.Exam	**dbo.Std_Answers.**Exam_ID = dbo.Exam.Exam_ID	FK_Std_Answers_Exam
≻	dbo.Question	**dbo.Std_Answers.**Q_ID = dbo.Question.Q_ID	FK_Std_Answers_Question
≻	dbo.Student	**dbo.Std_Answers.**Std_ID = dbo.Student.Std_ID	FK_Std_Answers_Student

Unique keys

	Columns	Name / Description
🔑	Std_ID, Exam_ID, Q_ID	PK_Std_Answers

Uses

Name
▦ dbo.Std_Answers
≻ dbo.Exam
≻ dbo.Question
≻ dbo.Student

Used By

Name
▦ dbo.Std_Answers
⚙ dbo.SP_RP_SELECTSTDANSWERS
⚙ dbo.SP_SELECTSTDANSWERS

2.1.10. Table: dbo.Std_Exam_Result

Status: Active

This table illustrates the relation between the student, the exam and the results of that axam.

Columns

Name			Data type	Description / Attributes
▦	🔑	Std_ID	int	**References**: dbo.Student
▦	🔑	Exam_ID	int	**References**: dbo.Exam
▦		Std_Result	int	

Links to

Table		Join	Title / Name / Description
≻	dbo.Exam	**dbo.Std_Exam_Result**.Exam_ID = dbo.Exam.Exam_ID	FK_Std_Exam_Result_Exam
≻	dbo.Student	**dbo.Std_Exam_Result**.Std_ID = dbo.Student.Std_ID	FK_Std_Exam_Result_Student

Unique keys

Columns		Name / Description
🔑	Std_ID, Exam_ID	PK_Std_Exam_Result

Uses

Name
▦ **dbo.Std_Exam_Result**
≻ dbo.Exam
≻ dbo.Student

Used By

Name
▦ **dbo.Std_Exam_Result**
⚙ dbo.SP_SELECTSTUDENTRESULTS

2.1.11. Table: dbo.Student

Status: Active

This table illustrates the relation between the student, the answers, the exam results and the department.

Columns

		Name	Data type	Description / Attributes
	🔑	Std_ID	int	Identity / Auto increment
		Std_FName	nvarchar(50)	Nullable
		Std_LName	nvarchar(50)	Nullable
		Std_Address	nvarchar(80)	Nullable
		Std_Email	nvarchar(50)	Nullable
		Std_DOB	date	Nullable
		Dept_id	int	Nullable **References**: dbo.Department

Links to

	Table	Join	Title / Name / Description
>-	dbo.Department	dbo.Student.Dept_id = dbo.Department.Dept_ID	FK_Student_Department

Linked from

	Table	Join	Title / Name / Description
-<	dbo.Std_Answers	dbo.Student.Std_ID = dbo.Std_Answers.Std_ID	FK_Std_Answers_Student
-<	dbo.Std_Exam_Result	dbo.Student.Std_ID = dbo.Std_Exam_Result.Std_ID	FK_Std_Exam_Result_Student

Unique keys

	Columns	Name / Description
🔑	Std_ID	PK_Student

Uses

Name
dbo.Student
>- dbo.Department

Used By

Name
dbo.Student
dbo.SP_DELETESTUDENT
dbo.SP_INSERTINTOSTD
dbo.SP_RP_INSTRUCTORCOURSESWITHSTUDENT
dbo.SP_RP_SELECTSTDANSWERS
dbo.SP_RP_STDINF

Name
dbo.SP_SELECTFROMSTD
dbo.SP_SELECTSTDANSWERS
dbo.SP_SELECTSTUDENTRESULTS
dbo.SP_UPDATESTDADRESS
dbo.SP_UPDATESTDEMAIL
dbo.Std_Answers
dbo.Std_Exam_Result

2.1.12. Table: dbo.Topic

Status: Active

This table illustrates the relation between each course and the topics of it.

Columns

		Name	Data type	Description / Attributes
▦	🔑	ID	int	Identity / Auto increment
▦		Name	nvarchar(50)	Nullable

Linked from

	Table	Join	Title / Name / Description
—<	dbo.Course	dbo.Topic.ID = dbo.Course.Topic_ID	FK_Courses_Topic

Unique keys

	Columns	Name / Description
🔑	ID	PK_Topic

Used By

Name
▦ dbo.Topic
⚙ dbo.INSERTINTOTOPIC
⚙ dbo.SELECTFROMTOPIC
⚙ dbo.SP_DELETETOPIC
⚙ dbo.SP_RP_COURSETOPIC
⚙ dbo.SP_UPDATEINTOTOPIC
⚙ dbo.SP_UPDATETOPIC
—< dbo.Course

2.1.13. Table: dbo.Type

Status: Active

This table illustrates the relation between each question and its type.

Columns

		Name	Data type	Description / Attributes
▦	🔑	T_ID	int	Identity / Auto increment
▦		T_Desc	nvarchar(50)	Nullable

Linked from

	Table	Join	Title / Name / Description
⤙	dbo.Question	dbo.Type.T_ID = dbo.Question.Type_ID	FK_Question_Type

Unique keys

	Columns	Name / Description
🔑	T_ID	PK_Type

Used By

Name
▦ dbo.Type
⚙ dbo.INSERTINTOTYPE
⚙ dbo.SELECTFROMTYPE
⚙ dbo.SP_DELETETYPE
⚙ dbo.SP_INSERTQUESTION
⚙ dbo.SP_UBDATEQUESTIONTYPE_OR_QUESTION
⚙ dbo.SP_UPDATETYPE
⤙ dbo.Question

2.2. Procedures

2.2.1. Procedure: dbo.INSERTINTOCOURSES

Status: Active

This stored procedure is created to insert into the courses table the name and the ID.

Input/Output

Name		Data type	Description
→@	C_name	nvarchar(50)	
→@	T_ID	int	

Uses

Name
⚙ dbo.INSERTINTOCOURSES
▦ dbo.Course

Script

```
CREATE proc [dbo].[INSERTINTOCOURSES]  (@C_name nvarchar(50) , @T_ID int )
as
insert into Course (Course_Name,Topic_ID)
values (@C_name , @T_ID)
```

2.2.2. Procedure: dbo.INSERTINTODEP

Status: Active

This stored procedure is created to insert into the department table the name, description, ID and hiring date of the managers.

Input/Output

	Name	Data type	Description
→@	name	nvarchar(50)	
→@	Desc	nvarchar(50)	
→@	id	int	
→@	Hire_D	date	

Uses

Name
⚙ dbo.INSERTINTODEP
▦ dbo.Department
▦ dbo.Instructor

Script

```
CREATE proc [dbo].[INSERTINTODEP]  (@name nvarchar(50),@Desc nvarchar(50)  , @id int = null  , @Hire_D date =  null  )
as
if @id is null
    begin
             insert into Department values (@name ,@Desc ,@id ,@Hire_D )
    end
   else
     begin
        if exists(select ins.Ins_ID from Instructor as ins where ins.Ins_ID=@id)
                        insert into Department values (@name ,@Desc ,@id ,@Hire_D )
                        else
                        begin
                          select 'There is no instructor with this id'
                        end
             end
```

2.2.3. Procedure: dbo.INSERTINTOTOPIC

Status: Active

This stored procedure is created to insert into the topic, its name.

Input/Output

Name		Data type	Description
→@	name	nvarchar(50)	

Uses

Name
⚙ dbo.INSERTINTOTOPIC
▦ dbo.Topic

Script

```
create proc [dbo].[INSERTINTOTOPIC] (@name nvarchar (50))
as
insert into Topic (Name)
values (@name)
```

2.2.4. Procedure: dbo.INSERTINTOTYPE

Status: Active

This stored procedure is created to insert into the question the type f it.

Input/Output

Name		Data type	Description
→@	name	nvarchar(50)	

Uses

Name
⚙ dbo.INSERTINTOTYPE
▦ dbo.Type

Script

```
create proc [dbo].[INSERTINTOTYPE] (@name nvarchar (50))
as
insert into Type(T_Desc)
values (@name)
```

2.2.5. Procedure: dbo.SELECTFROMTOPIC

Status: Active

Input/Output

	Name	Data type	Description
→@	SEARCH	nvarchar(50)	

Uses

Name
dbo.SELECTFROMTOPIC
dbo.Topic

Script

```
CREATE proc [dbo].[SELECTFROMTOPIC] (@SEARCH nvarchar(50))
as
if (ISNUMERIC(@SEARCH)=1)
begin
declare @id int;
select @id =  CONVERT(int,@SEARCH)
end
select * from Topic  as T
where t.name =@SEARCH or t.ID=@id
```

2.2.6. Procedure: dbo.SELECTFROMTYPE

Status: Active

Input/Output

	Name	Data type	Description
→@	SEARCH	nvarchar(50)	

Uses

Name
⚙ dbo.SELECTFROMTYPE
▦ dbo.Type

Script

```
create proc [dbo].[SELECTFROMTYPE] (@SEARCH nvarchar(50))
as
if (ISNUMERIC(@SEARCH)=1)
begin
declare @id int;
select @id =  CONVERT(int,@SEARCH)
end
select * from Type as T
where t.T_Desc=@SEARCH or T_ID=@id
```

2.2.7. Procedure: dbo.SP_DELETECOURSE

Status: Active

This stored procedure is created to delete from the course name and the ID.

Input/Output

	Name	Data type	Description
→@	Selector	nvarchar(50)	

Uses

Name
⚙ dbo.SP_DELETECOURSE
▦ dbo.Course

Script

```
create proc [dbo].[SP_DELETECOURSE] (@Selector nvarchar(50))
as
if(ISNUMERIC(@Selector) = 1)
begin
declare @id int
select @id = CONVERT(int,@Selector)
delete from Course where Course_ID=@id
end
else
begin
delete from Course where Course_Name=@Selector
end
```

2.2.8. Procedure: dbo.SP_DELETEDEPARTMENTCOURSE

Status: Active

Input/Output

	Name	Data type	Description
→@	deo_ID	nvarchar(50)	
→@	course_ID	nvarchar(50)	

Uses

Name
⚙ dbo.SP_DELETEDEPARTMENTCOURSE
▦ dbo.Dept_Course

Script

```
CREATE proc [dbo].[SP_DELETEDEPARTMENTCOURSE] (@deo_ID nvarchar(50) , @course_ID nvarchar(50))

as
           if (ISNUMERIC(@deo_ID)=1)
                      begin
                                 if(ISNUMERIC(@course_ID)=1)
                                            begin
                                                       if exists (select * from Dept_Course as dc where dc.Dept_ID=@deo_ID
and dc.Course_ID=@course_ID)
                                                                  begin
                                                                             delete from Dept_Course where Dept_ID=@deo_ID
and Course_ID=@course_ID
                                                                  end
                                                       else
                                                                  begin
                                                                             select 'check you data, no data with your
Inputs'
                                                                  end
                                            end
                      end
           else
                      begin
                                 select 'enter a valid data to continue'
                      end
```

2.2.9. Procedure: dbo.SP_DELETEEXAM

Status: Active

Input/Output

	Name	Data type	Description
→@	Selector	nvarchar(50)	

Uses

Name
⚙ dbo.SP_DELETEEXAM
▦ dbo.Exam

Script

```
create proc [dbo].[SP_DELETEEXAM] (@Selector nvarchar(50))

as
           if(ISNUMERIC(@Selector) =1)
                      begin
                                 declare @id int;
                                 select @id = convert(int , @selector)
                                            if exists (select * from Exam where Exam_ID=@id)
                                                       begin
                                                                  delete from Exam where Exam_ID=@id
                                                                  select 'Done, The Exam with id :'+'
'+convert(nvarchar(50),@id)+' '+'is deleted'
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no Exam with this ID : ' +
convert(nvarchar(50),@id)
                                                       end
                      end
```

2.2.10. Procedure: dbo.SP_DELETEFROMQUESTION

Status: Active

Input/Output

	Name	Data type	Description
→@	QUESTION	nvarchar(50)	

Uses

Name
dbo.SP_DELETEFROMQUESTION
dbo.Question

Script

```
CREATE proc [dbo].[SP_DELETEFROMQUESTION](@QUESTION nvarchar(50))
           as
                      if(ISNUMERIC(@QUESTION)=0)
                                 begin
                                            if exists(select * from Question where Q_Desc=@QUESTION)
                                                       begin
                                                                  delete from Question where Q_Desc=@QUESTION
                                                                  select 'This question is deleted :'+@QUESTION
                                                       end
                                            else
                                                       begin
                                                                  select 'No Question with this Entry : '+@QUESTION
                                                       end
                                 end
                      else
                                 BEGIN
                                            SELECT     'Check your data !!'
                                 END
```

2.2.11. Procedure: dbo.SP_DELETEINSCOURSE

Status: Active

Input/Output

	Name	Data type	Description
→@	isn_ID	nvarchar(50)	
→@	course_ID	nvarchar(50)	

Uses

Name
dbo.SP_DELETEINSCOURSE
dbo.Ins_Course

Script

```
create proc [dbo].[SP_DELETEINSCOURSE](@isn_ID nvarchar(50), @course_ID nvarchar(50))
           as
                      if(ISNUMERIC(@isn_ID)=1)
                                 begin
                                            if(ISNUMERIC(@course_ID)=1)
                                                       begin
                                                                  if exists (select * from Ins_Course as ic where
ic.Course_ID=@course_ID and ic.Ins_ID=@isn_ID)
                                                                             begin
                                                                                        delete from Ins_Course where
Course_ID=@course_ID and Ins_ID=@isn_ID
                                                                                        select 'These data are deleted :
course_ID -> '+@course_ID+'  ins_ID  -> '+@isn_ID
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'There is no data with this
selectors !!'
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'Check the data !!'
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check the data !!'
                                 end
```

2.2.12. Procedure: dbo.SP_DELETEINSTRUCTOR

Status: Active

Input/Output

	Name	Data type	Description
→@	Selector	nvarchar(50)	

Uses

Name
⚙ dbo.SP_DELETEINSTRUCTOR
▦ dbo.Instructor

Script

```
create proc [dbo].[SP_DELETEINSTRUCTOR] (@Selector nvarchar(50))
as
    if(ISNUMERIC(@Selector) =1)
        begin
            declare @id int;
            select @id = convert(int , @selector)
                if exists (select * from Instructor where Ins_ID=@id)
                    begin
                        delete from Instructor where Ins_ID=@id
                        select 'Done, The Instructor with id :'+'
'+convert(nvarchar(50),@id)+' '+'is deleted'
                    end
                else
                    begin
                        select 'There is no Instructor with this ID : ' +
convert(nvarchar(50),@id)
                    end
        end
    else
        begin
            if exists (select * from Instructor where (Ins_FName+' '+Ins_LName) =@Selector)
                begin
                    delete from Instructor where (Ins_FName+' '+Ins_LName) =@Selector
                    select 'Done, The Instructor with name :'+' '+@Selector+' '+'is
deleted'
                end
            else
                begin
                    select 'There is no Instructor with this name : ' + @Selector
                end
        end
```

2.2.13. Procedure: dbo.SP_DELETESTUDENT

Status: Active

Input/Output

	Name	Data type	Description
→@	Selector	nvarchar(50)	

Uses

Name
dbo.SP_DELETESTUDENT
dbo.Student

Script

```
CREATE proc [dbo].[SP_DELETESTUDENT] (@Selector nvarchar(50))
as
           if(ISNUMERIC(@Selector) =1)
                      begin
                                 declare @id int;
                                 select @id = convert(int , @selector)
                                            if exists (select * from Student where Std_ID=@id)
                                                       begin
                                                                  delete from Student where Std_ID=@id
                                                                  select 'Done, The student with id :'+'
'+convert(nvarchar(50),@id)+' '+'is deleted'
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no Student with this ID : ' +
convert(nvarchar(50),@id)
                                                       end
                      end
           else
                      begin
                                 if exists (select * from student where (Std_FName+' '+Std_LName) =@Selector)
                                            begin
                                                       delete from Student where (Std_FName+' '+Std_LName) =@Selector
                                                       select 'Done, The student with name :'+' '+@Selector+' '+'is deleted'
                                            end
                                 else
                                            begin
                                                       select 'There is no student with this name : ' + @Selector
                                            end
                      end
```

2.2.14. Procedure: dbo.SP_DELETETOPIC

Status: Active

Input/Output

	Name	Data type	Description
→@	Selector	nvarchar(50)	

Uses

Name
dbo.SP_DELETETOPIC
dbo.Topic

Script

```
create proc [dbo].[SP_DELETETOPIC] (@Selector nvarchar(50))
as
if(ISNUMERIC(@Selector) = 1)
begin
declare @id int
select @id = CONVERT(int,@Selector)
delete from Topic where ID=@id
end
else
begin
delete from Topic where Name=@Selector
end
```

2.2.15. Procedure: dbo.SP_DELETETYPE

Status: Active

Input/Output

	Name	Data type	Description
→@	Selector	nvarchar(50)	

Uses

Name
dbo.SP_DELETETYPE
dbo.Type

Script

```
create proc [dbo].[SP_DELETETYPE] (@Selector nvarchar(50))
as
if(ISNUMERIC(@Selector) = 1)
begin
declare @id int
select @id = CONVERT(int,@Selector)
delete from Type where T_ID=@id
end
else
begin
delete from Type where T_Desc=@Selector
end
```

2.2.16. Procedure: dbo.Sp_GenerateExam

Status: Active

Input/Output

Name		Data type	Description
@	e_ID	int	
@	c_ID	int	

Uses

Name
dbo.Sp_GenerateExam
dbo.Course
dbo.Exam
dbo.Exam_Questions
dbo.Question

Script

```
create proc [dbo].[Sp_GenerateExam] (@e_ID int,@c_ID int)
as
insert into Exam_Questions (Exam_ID,Q_ID,Exam_Q,Q_rank)

select top(2)  e.Exam_ID,qu.Q_ID,qu.Q_Desc  , Dense_rank()over(partition by qu.Course_ID order by newid())
from dbo.Question as qu
inner join    dbo.Course as co
on co.Course_ID = qu.Course_ID and qu.Course_ID=@c_ID and qu.Type_ID=2
inner join Exam as e
on e.Course_ID = co.Course_ID and e.Exam_ID=@e_ID

union all

select top(2)e.Exam_ID,qu.Q_ID,qu.Q_Desc , Dense_rank()over(partition by qu.Course_ID order by newid())
from dbo.Question as qu
inner join  dbo.Course as co
on co.Course_ID = qu.Course_ID and qu.Course_ID=@c_ID and qu.Type_ID=1
inner join Exam as e
on e.Course_ID = co.Course_ID and e.Exam_ID=@e_ID
```

2.2.17. Procedure: dbo.SP_INSERTDEPARTMENTCOURSE

Status: Active

Input/Output

	Name	Data type	Description
→@	dep_ID	nvarchar(50)	
→@	Course_ID	nvarchar(50)	
→@	duration	nvarchar(50)	

Uses

Name
dbo.SP_INSERTDEPARTMENTCOURSE
dbo.Course
dbo.Department
dbo.Dept_Course

Script

```
CREATE proc [dbo].[SP_INSERTDEPARTMENTCOURSE] ( @dep_ID nvarchar(50) , @Course_ID nvarchar(50) , @duration nvarchar(50)=NULL
)

           as
                      if(ISNUMERIC(@dep_ID)=1)
                                 begin
                                            if(ISNUMERIC(@Course_ID)=1)
                                                       begin
                                                                  if exists (select * from Department as d where
d.Dept_ID=@dep_ID)
                                                                             begin
                                                                                        if exists(select * from Course as c
where c.Course_ID=@Course_ID)
                                                                                                   begin
                                                                                                              insert into
Dept_Course Values (@dep_ID,@Course_ID,@duration)
                                                                                                   end
                                                                                        else
                                                                                                   begin
                                                                                                              select 'Check
the course Number'
                                                                                                   end
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'Check the deprtment Number'
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'Enter a valid Data'
                                                       end

                                 end
                      else
                                 begin
                                            select 'Enter a Valid Data !!'
                                 end
```

2.2.18. Procedure: dbo.SP_INSERTINSCOURSES

Status: Active

Input/Output

	Name	Data type	Description
→@	ins_Selector	nvarchar(50)	
→@	course_Selector	nvarchar(50)	

Uses

Name
dbo.SP_INSERTINSCOURSES
dbo.Course
dbo.Ins_Course
dbo.Instructor

Script

```
CREATE proc [dbo].[SP_INSERTINSCOURSES] (@ins_Selector nvarchar(50) , @course_Selector nvarchar(50))
           as
                      if(ISNUMERIC(@ins_Selector)=1)
                                 begin
                                            if(ISNUMERIC(@course_Selector)=1)
                                                       begin
                                                                  if exists (select * from Instructor where 
Ins_ID=@ins_Selector)
                                                                             begin
                                                                                        if exists (select * from Course where 
Course_ID=@course_Selector)
                                                                                                   begin
                                                                                                              if exists 
(select * from Ins_Course as c where c.Course_ID=@course_Selector and c.Ins_ID=@ins_Selector)

           begin

                                                                                                                         end
                                                                                                              else

           begin


                                                                                                                         end
                                                                                                   end
                                                                                        else
                                                                                                   begin
                                                                                                              select 'Course 
id not valid'
                                                                                                   end
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'Instructor id not valid'
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'Check your data !! the both entries must be names 
or ids'
                                                       end
                                 end
                      else
                                 begin
                                            declare @ins_ID int;
                                            declare @course_ID int;
                                            if exists (select i.Ins_ID from Instructor as i where (Ins_FName+' 
'+Ins_LName)=@ins_Selector)
                                                       begin
                                                                  if exists (select * from Course as c where 
c.Course_Name=@course_Selector)
                                                                             begin
                                                                                        select @ins_ID = i.Ins_ID from 
Instructor as i where (Ins_FName+' '+Ins_LName)=@ins_Selector
                                                                                        select @course_ID = c.Course_ID  from 
Course as c where c.Course_Name=@course_Selector

                                                                                                              if exists 
(select * from Ins_Course as c where c.Course_ID=@course_ID and c.Ins_ID=@ins_ID)

           begin

                                                                                                                         end
                                                                                                              else

           begin


                                                                                                                         end
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'Course name not valid'
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'Instructor name not valid'
                                                       end
                                 end
```

2.2.19. Procedure: dbo.SP_INSERTINTOEXAM

Status: Active

Input/Output

	Name	Data type	Description
→@	Desc	nvarchar(50)	
→@	Date	date	
→@	id	int	

Uses

Name
⚙ dbo.SP_INSERTINTOEXAM
▦ dbo.course
▦ dbo.Exam

Script

```
create proc [dbo].[SP_INSERTINTOEXAM] (@Desc nvarchar(50), @Date date ,@id int)
as
if exists (select c.Course_ID from course as c where c.Course_ID=@id)
  begin
    insert into Exam (Exam_Desc ,Exam_Date ,Course_ID )
    values (@Desc , @Date , @id)
  end
   else
     begin
       select 'There is no Course with this ID '
    end
```

2.2.20. Procedure: dbo.SP_INSERTINTOINST

Status: Active

Input/Output

Name	Data type	Description
→@ Fname	nvarchar(50)	
→@ Lname	nvarchar(50)	
→@ Address	nvarchar(50)	
→@ BOD	date	
→@ Email	nvarchar(50)	
→@ salary	int	
→@ id	int	

Uses

Name
⚙ dbo.SP_INSERTINTOINST
▦ dbo.Department
▦ dbo.Instructor

Script

```
create proc [dbo].[SP_INSERTINTOINST] (@Fname nvarchar(50),@Lname nvarchar(50),@Address nvarchar(50) ,@BOD date,@Email 
nvarchar(50),@salary int , @id int)
as
if exists (select d.Dept_ID from Department as d where d.Dept_ID=@id)
  begin
    insert into dbo.Instructor (Ins_FName , Ins_LName, Ins_Address, Ins_DOB  , Ins_Email , Ins_Salary , Dept_ID)
    values (@Fname,@Lname,@Address,@BOD,@Email ,@salary,@id)
  end
   else
     begin
       select 'There is no Department with this ID '
    end
```

2.2.21. Procedure: dbo.SP_INSERTINTOSTD

Status: Active

Input/Output

	Name	Data type	Description
→@	Fname	nvarchar(50)	
→@	Lname	nvarchar(50)	
→@	Address	nvarchar(80)	
→@	Email	nvarchar(50)	
→@	BOD	date	
→@	id	int	

Uses

Name
dbo.SP_INSERTINTOSTD
dbo.Department
dbo.Student

Script

```
CREATE proc [dbo].[SP_INSERTINTOSTD] (@Fname nvarchar(50),@Lname nvarchar(50),@Address nvarchar(80) ,@Email nvarchar(50),
@BOD date ,@id int)
as
if exists (select d.Dept_ID from Department as d where d.Dept_ID=@id)
  begin
    insert into Student(Std_FName , Std_LName , Std_Address, Std_Email, Std_DOB , Dept_ID)
    values (@Fname,@Lname,@Address,@Email ,@BOD,@id)
  end
   else
     begin
       select 'There is no Department with this ID '
    end
```

2.2.22. Procedure: dbo.SP_INSERTQUESTION

Status: Active

Input/Output

Name		Data type	Description
→@	Q_desc	nvarchar(50)	
→@	Q_Model	nvarchar(50)	
→@	Q_type	nvarchar(50)	
→@	course_ID	nvarchar(50)	
→@	ch1	nvarchar(50)	
→@	ch2	nvarchar(50)	
→@	ch3	nvarchar(50)	
→@	ch4	nvarchar(50)	

Uses

Name
⚙ dbo.SP_INSERTQUESTION
▦ dbo.course
▦ dbo.Question
▦ dbo.Type

Script

```
CREATE proc [dbo].[SP_INSERTQUESTION]
(
@Q_desc nvarchar(50),@Q_Model nvarchar(50),@Q_type nvarchar(50),@course_ID nvarchar(50),
@ch1 nvarchar(50)=null,@ch2 nvarchar(50)=null,@ch3 nvarchar(50)=null,@ch4 nvarchar(50)=null
)
           as
                      if exists (select * from Type where T_ID=@Q_type)
                                 begin
                                             if exists (select * from course where course_ID=@course_ID)
                                                       begin
                                                                  insert into Question
(Q_Desc,Q_Model_Ans,Type_ID,Course_ID,Choice_1,Choice_2,Choice_3,Choice_4)
                                                                  values
(@Q_desc,@Q_Model,@Q_type,@course_ID,@ch1,@ch2,@ch3,@ch4)
                                                                  select 'Data inserted...'
                                                       end
                                            else
                                                       begin
                                                                  select 'check course id !!'
                                                       end
                                 end
                      else
                                 begin
                                            select 'check the type !!'
                                 end
```

2.2.23. Procedure: dbo.SP_RP_COURSETOPIC

Status: Active

Input/Output

	Name	Data type	Description
→@	co_ID	nvarchar(50)	

Uses

Name
dbo.SP_RP_COURSETOPIC
dbo.Course
dbo.Topic

Script

```
create proc [dbo].[SP_RP_COURSETOPIC](@co_ID nvarchar(50))
           as
                      if(ISNUMERIC(@co_ID)=1)
                                 begin
                                            if exists (select * from Course as c where c.Course_ID=@co_ID)
                                                       begin
                                                                  select  c.Course_ID , c.Course_Name , t.Name from Course as
c
                                                                  inner join Topic as t on c.Topic_ID = t.ID
                                                                  where c.Course_ID=@co_ID
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no course with this ID : '+@co_ID
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check the data !!'
                                 end
```

2.2.24. Procedure: dbo.SP_RP_INSTRUCTORCOURSESWITHSTUDENT

Status: Active

Input/Output

	Name	Data type	Description
→@	ins_ID	nvarchar(50)	

Uses

Name
⚙ dbo.SP_RP_INSTRUCTORCOURSESWITHSTUDENT
dbo.Course
dbo.Department
dbo.Ins_Course
dbo.Instructor
dbo.Student

Script

```
create proc [dbo].[SP_RP_INSTRUCTORCOURSESWITHSTUDENT] (@ins_ID nvarchar(50))
           as
                      if(ISNUMERIC(@ins_ID)=1)
                                 begin
                                            if exists (select * from Instructor as i where i.Ins_ID=@ins_ID)
                                                       begin
                                                                  select COUNT(s.Std_ID) as
NO_OF_STUDENTS,i.Ins_ID,i.Ins_FName,ic.Course_ID,c.Course_Name
                                                                  from Student as s
                                                                  inner join Department as d on d.Dept_ID=s.Dept_id
                                                                  inner join Instructor as i on i.Dept_ID=d.Dept_ID
                                                                  inner join Ins_Course as ic on ic.Ins_ID=i.Ins_ID and
ic.Ins_ID=@ins_ID
                                                                  inner join Course     as c  on c.Course_ID=ic.Course_ID
                                                                  group by s.Dept_id , i.Ins_ID , ic.Course_ID , i.Ins_FName
,c.Course_Name
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no instructor with this id : '+@ins_ID
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check your data !!'
                                 end
```

2.2.25. Procedure: dbo.SP_RP_SELECTEXAMQUESTIONWITHCHOICES

Status: Active

Input/Output

Name		Data type	Description
→@	examNumber	nvarchar(50)	

Uses

Name
dbo.SP_RP_SELECTEXAMQUESTIONWITHCHOICES
dbo.Exam
dbo.Exam_Questions
dbo.Question

Script

```
create proc [dbo].[SP_RP_SELECTEXAMQUESTIONWITHCHOICES] (@examNumber nvarchar(50))
           as
                      if(ISNUMERIC(@examNumber)=1)
                                 begin
                                            if exists(select * from Exam as e where e.Exam_ID=@examNumber)
                                                       begin
                                                                  if exists (select * from Exam_Questions as eq where 
eq.Exam_ID=@examNumber)
                                                                             begin
                                                                                        select eq.Exam_ID , q.Q_ID , 
q.Q_Desc,q.Choice_1,q.Choice_2,q.Choice_3,q.Choice_4  from Exam_Questions as eq
                                                                                        inner join Question as q on 
eq.Q_ID=q.Q_ID
                                                                                        where eq.Exam_ID=@examNumber
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'There is no Questions for 
this exam : '+@examNumber
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no exam with this id :'+@examNumber
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check your data the input must an exam ID !!!'
                                 end
```

2.2.26. Procedure: dbo.SP_RP_SELECTSTDANSWERS

Status: Active

Input/Output

	Name	Data type	Description
→@	STD_ID	nvarchar(50)	
→@	EXAM_ID	nvarchar(50)	

Uses

Name
dbo.SP_RP_SELECTSTDANSWERS
dbo.Exam
dbo.Question
dbo.Std_Answers
dbo.Student

Script

```
create proc [dbo].[SP_RP_SELECTSTDANSWERS](@STD_ID nvarchar(50),@EXAM_ID nvarchar(50))
           as
                      if(ISNUMERIC(@STD_ID)=1)
                                 begin
                                            if(ISNUMERIC(@EXAM_ID)=1)
                                                       begin
                                                                  if exists (select * from Student as s where
s.Std_ID=@STD_ID)
                                                                             begin
                                                                                        if exists (select * from Exam as ex
where ex.Exam_ID=@EXAM_ID)
                                                                                                   begin
                                                                                                              if exists
(select * from Std_Answers as s where s.Exam_ID=@EXAM_ID and s.Std_ID=@STD_ID)

           begin


                                                                                                                         end
                                                                                                              else

           begin

                                                                                                                         end
                                                                                                   end
                                                                                        else
                                                                                                   begin
                                                                                                              select 'There
is no exam with this id : '+@EXAM_ID
                                                                                                   end
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'There is no student with this
id : '+@STD_ID
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'Check the data !!'
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check the data !!'
                                 end
```

2.2.27. Procedure: dbo.SP_RP_STDINF

Status: Active

Input/Output

	Name	Data type	Description
→@	dep_ID	nvarchar(50)	

Uses

Name
dbo.SP_RP_STDINF
dbo.Department
dbo.Student

Script

```
create proc [dbo].[SP_RP_STDINF] (@dep_ID nvarchar(50))
           as
                      if(ISNUMERIC(@dep_ID)=1)
                                 begin
                                            if exists (select * from Department where Dept_ID=@dep_ID)
                                                       begin
                                                                  select * from Student where Dept_id=@dep_ID
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no department with this id : '+@dep_ID
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check the data !!'
                                 end
```

2.2.28. Procedure: dbo.SP_SELECTALLCOURSESINDEPARTMENT

Status: Active

Input/Output

	Name	Data type	Description
→@	dep_ID	nvarchar(50)	

Uses

Name
⚙ dbo.SP_SELECTALLCOURSESINDEPARTMENT
▦ dbo.Course
▦ dbo.Department
▦ dbo.Dept_Course

Script

```
CREATE proc [dbo].[SP_SELECTALLCOURSESINDEPARTMENT] (@dep_ID nvarchar(50))
           as
                      if(ISNUMERIC(@dep_ID)=1)
                                 begin
                                            if exists (select * from Dept_Course where  Dept_ID=@dep_ID)
                                                       begin
                                                                  select
dc.Dept_ID,d.Dept_Name,dc.Course_ID,c.Course_Name,dc.Course_Duration from Dept_Course as dc
                                                                  inner join Course as c on dc.Course_ID = c.Course_ID
                                                                  inner join Department as d on dc.Dept_ID = d.Dept_ID
                                                                  where  dc.Dept_ID=@dep_ID
                                                       end
                                            else
                                                       begin
                                                                  select 'No Courses with this Dept_ID'
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check Your Data !!'
                                 end
```

2.2.29. Procedure: dbo.SP_SELECTEXAMQUESTION

Status: Active

Input/Output

	Name	Data type	Description
→@	examNumber	nvarchar(50)	

Uses

Name
dbo.SP_SELECTEXAMQUESTION
dbo.Exam
dbo.Exam_Questions
dbo.Question

Script

```
CREATE proc [dbo].[SP_SELECTEXAMQUESTION] (@examNumber nvarchar(50))
           as
                      if(ISNUMERIC(@examNumber)=1)
                                 begin
                                            if exists(select * from Exam as e where e.Exam_ID=@examNumber)
                                                       begin
                                                                  if exists (select * from Exam_Questions as eq where
eq.Exam_ID=@examNumber)
                                                                             begin
                                                                                        select eq.Exam_ID , q.Q_ID , q.Q_Desc
from Exam_Questions as eq
                                                                                        inner join Question as q on
eq.Q_ID=q.Q_ID
                                                                                        where eq.Exam_ID=@examNumber
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'There is no Questions for
this exam : '+@examNumber
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no exam with this id :'+@examNumber
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check your data the input must an exam ID !!!'
                                 end
```

2.2.30. Procedure: dbo.SP_SELECTFROMCOURSE

Status: Active

Input/Output

	Name	Data type	Description
→@	SEARCH	nvarchar(50)	

Uses

Name
dbo.SP_SELECTFROMCOURSE
dbo.Course

Script

```
CREATE proc [dbo].[SP_SELECTFROMCOURSE] (@SEARCH nvarchar(50))
as
if (ISNUMERIC(@SEARCH)=1)
begin
declare @id int;
select @id =  CONVERT(int,@SEARCH)
end
select * from Course as c
where c.Course_Name=@SEARCH or c.Course_ID=@id
```

2.2.31. Procedure: dbo.SP_SELECTFROMDEP

Status: Active

Input/Output

	Name	Data type	Description
→@	SEARCH	nvarchar(50)	

Uses

Name
dbo.SP_SELECTFROMDEP
dbo.Department

Script

```
create proc [dbo].[SP_SELECTFROMDEP] (@SEARCH nvarchar(50))
as
if (ISNUMERIC(@SEARCH)=1)
begin
declare @id int;
select @id =  CONVERT(int,@SEARCH)
end
select * from Department as D
where D.Dept_Name=@SEARCH or D.Dept_ID=@id
```

2.2.32. Procedure: dbo.SP_SELECTFROMEXAM

Status: Active

Input/Output

	Name	Data type	Description
→@	SEARCH	nvarchar(50)	

Uses

Name
⚙ dbo.SP_SELECTFROMEXAM
▦ dbo.Exam

Script

```
CREATE proc [dbo].[SP_SELECTFROMEXAM] (@SEARCH nvarchar(50))
as
if (ISNUMERIC(@SEARCH)=1)
begin
declare @id int;
select @id =  CONVERT(int,@SEARCH)
end
select * from Exam as E
where E.Exam_Desc=@SEARCH
or E.Exam_Date=@SEARCH or E.Exam_ID=@id
```

2.2.33. Procedure: dbo.SP_SELECTFROMINST

Status: Active

Input/Output

	Name	Data type	Description
→@	SEARCH	nvarchar(50)	

Uses

Name
dbo.SP_SELECTFROMINST
dbo.Instructor

Script

```
create proc [dbo].[SP_SELECTFROMINST] (@SEARCH nvarchar(50))
as
if (ISNUMERIC(@SEARCH)=1)
begin
declare @id int , @date date ;
select @id =  CONVERT(int,@SEARCH)
select @date = CONVERT(date ,@SEARCH )
end
select * from Instructor as I
where I.Ins_FName=@SEARCH or I.Ins_LName=@SEARCH or
i.Ins_Address=@SEARCH or I.Ins_Salary= @id or
i.Ins_DOB=@date or i.Ins_ID=@id
```

2.2.34. Procedure: dbo.SP_SELECTFROMSTD

Status: Active

Input/Output

	Name	Data type	Description
→@	SEARCH	nvarchar(50)	

Uses

Name
dbo.SP_SELECTFROMSTD
dbo.Student

Script

```
create proc [dbo].[SP_SELECTFROMSTD] (@SEARCH nvarchar(50))
as
if (ISNUMERIC(@SEARCH)=1)
begin
declare @id int , @date date ;
select @id =  CONVERT(int,@SEARCH)
select @date = CONVERT(date ,@SEARCH )
end
select * from Student as S
where s.Std_FName=@SEARCH or S.std_LName=@SEARCH or
S.Std_Address=@SEARCH or s.Std_Email= @SEARCH or
S.Std_DOB=@date or S.Std_ID=@id
```

2.2.35. Procedure: dbo.SP_SELECTINSCOURSE

Status: Active

Input/Output

	Name	Data type	Description
→@	ins_selctor	nvarchar(50)	

Uses

Name
dbo.SP_SELECTINSCOURSE
dbo.Course
dbo.Ins_Course
dbo.Instructor

Script

```
CREATE proc [dbo].[SP_SELECTINSCOURSE](@ins_selctor nvarchar(50))
           as

                      if(ISNUMERIC(@ins_selctor)=1)
                                 begin
                                            if exists (select * from Instructor as i where i.Ins_ID=@ins_selctor)
                                                       begin
                                                                  if exists (select * from Ins_Course as ic where
ic.Ins_ID=@ins_selctor)

                                                                             begin
                                                                                        select ic.Ins_ID,(i.Ins_FName+'

'+i.Ins_LName) , ic.Course_ID, c.Course_Name

                                                                                        from Ins_Course as ic
                                                                                        inner join Course as c on

c.Course_ID=ic.Course_ID

                                                                                        inner join Instructor as i on
i.Ins_ID = ic.Ins_ID

                                                                                        where ic.Ins_ID=@ins_selctor
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'This instructor not assigned
to any courses...'

                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no instructor with this ID :
'+@ins_selctor

                                                       end
                                 end
                      else
                                 begin
                                            if exists (select * from Instructor as i where (i.Ins_FName+'
'+i.Ins_LName)=@ins_selctor)

                                                       begin
                                                                  declare @ins_ID nvarchar(50);
                                                                  select @ins_ID=i.Ins_ID from Instructor as i where
(i.Ins_FName+' '+i.Ins_LName)=@ins_selctor

                                                                  select ic.Ins_ID,(i.Ins_FName+' '+i.Ins_LName) ,
ic.Course_ID, c.Course_Name

                                                                                        from Ins_Course as ic
                                                                                        inner join Course as c on

c.Course_ID=ic.Course_ID

                                                                                        inner join Instructor as i on
i.Ins_ID = ic.Ins_ID

                                                                                        where ic.Ins_ID=@ins_ID
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no instructor with this name :
'+@ins_selctor

                                                       end

                                 end
```

2.2.36. Procedure: dbo.SP_SELECTQUESTION

Status: Active

Input/Output

	Name	Data type	Description
→@	selector	nvarchar(50)	

Uses

Name
dbo.SP_SELECTQUESTION
dbo.Course
dbo.Question

Script

```
create proc [dbo].[SP_SELECTQUESTION] (@selector nvarchar(50))
           as
                      if(ISNUMERIC(@selector)=1)
                                 begin

                                            if exists (select * from Course as c where c.Course_ID=@selector)
                                                       begin
                                                                  select * from Question as Q where Q.Course_ID=@selector
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no course with this ID'
                                                       end
                                 end
                      else
                                 begin
                                            if exists (select * from Question where Q_Desc=@selector)
                                                       begin
                                                                  select * from Question where Q_Desc=@selector
                                                       end
                                            else
                                                       begin
                                                                  select 'There is no Question matched'
                                                       end
                                 end
```

2.2.37. Procedure: dbo.SP_SELECTSTDANSWERS

Status: Active

Input/Output

	Name	Data type	Description
→@	STD_ID	nvarchar(50)	
→@	EXAM_ID	nvarchar(50)	

Uses

Name
dbo.SP_SELECTSTDANSWERS
dbo.Exam
dbo.Question
dbo.Std_Answers
dbo.Student

Script

```
create proc [dbo].[SP_SELECTSTDANSWERS](@STD_ID nvarchar(50),@EXAM_ID nvarchar(50))
           as
                      if(ISNUMERIC(@STD_ID)=1)
                                 begin
                                           if(ISNUMERIC(@EXAM_ID)=1)
                                                      begin
                                                                 if exists (select * from Student as s where
s.Std_ID=@STD_ID)
                                                                            begin
                                                                                       if exists (select * from Exam as ex
where ex.Exam_ID=@EXAM_ID)
                                                                                                  begin
                                                                                                             if exists
(select * from Std_Answers as s where s.Exam_ID=@EXAM_ID and s.Std_ID=@STD_ID)

           begin


                                                                                                                       end
                                                                                                             else

           begin

                                                                                                                       end
                                                                                                  end
                                                                                       else
                                                                                                  begin
                                                                                                             select 'There
is no exam with this id : '+@EXAM_ID
                                                                                                  end
                                                                            end
                                                                 else
                                                                            begin
                                                                                       select 'There is no student with this
id : '+@STD_ID
                                                                            end
                                                      end
                                           else
                                                      begin
                                                                 select 'Check the data !!'
                                                      end
                                 end
                      else
                                 begin
                                           select 'Check the data !!'
                                 end
```

2.2.38. Procedure: dbo.SP_SELECTSTUDENTRESULTS

Status: Active

Input/Output

	Name	Data type	Description
→@	std_ID	nvarchar(50)	
→@	course_name	nvarchar(50)	

Uses

Name
dbo.SP_SELECTSTUDENTRESULTS
dbo.Course
dbo.Exam
dbo.Std_Exam_Result
dbo.Student

Script

```
CREATE proc [dbo].[SP_SELECTSTUDENTRESULTS] (@std_ID nvarchar(50) , @course_name nvarchar(50)=null)
    as
        if(ISNUMERIC(@std_ID)=1)
            begin
                if(ISNUMERIC(@course_name)=0)
                    begin
                        if(@course_name is not null)
                            begin

                                if exists (select * from Student
where Std_ID=@std_ID)

                                    begin
                                        if exists

(select * from Course as c where c.Course_Name=@course_name)

    begin




                                            end
                                        else

    begin

                                            end
                                    end
                                else
                                    begin
                                        select 'There
is no student with this id : '+@std_ID

                                    end
                            end
                        else
                            begin


                                 if(@course_name is null)
                                    begin
                                        select 'ana
null'

                                        select * from

Std_Exam_Result as er where Std_ID=@std_ID

                                    end
                                else
                                    begin
                                        select 'Check
the data 11 !!'

                                    end

                            end
                    end
                else
                    begin
                        select 'Check the data 22 !!'
                    end
            end
        else
            begin
                select 'Check the data 33 !!'
            end
```

2.2.39. Procedure: dbo.SP_UBDATEQUESTIONTYPE_OR_QUESTION

Status: Active

Input/Output

Name	Data type	Description
→@ Question	nvarchar(50)	
→@ newdata	nvarchar(50)	

Uses

Name
dbo.SP_UBDATEQUESTIONTYPE_OR_QUESTION
dbo.Question
dbo.Type

Script

```sql
CREATE proc [dbo].[SP_UBDATEQUESTIONTYPE_OR_QUESTION] (@Question nvarchar(50) , @newdata nvarchar(50))
           as
                      if(ISNUMERIC(@newdata)=1)
                                 begin
                                            if exists (select * from Question where Q_Desc = @Question)
                                                       begin
                                                                  if exists (select * from dbo.Type where T_ID = @newdata)
                                                                             begin
                                                                                        update Question set Type_ID=@newdata
where Q_Desc=@Question
                                                                                        select 'Type Updated'
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'Check the type, There is no
Type with this id : '+@newdata
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'Can not update type, There is no Question with this
description : '+@Question
                                                       end
                                 end
                      else
                                 begin
                                            if exists (select * from Question where Q_Desc = @Question)
                                                       begin
                                                                  update Question set Q_Desc=@newdata where Q_Desc=@Question
                                                                  select 'Question Updated'
                                                       end
                                            else
                                                       begin
                                                                  select 'Can not update Question, There is no Question with
this description : '+@Question
                                                       end
                                 end
```

2.2.40. Procedure: dbo.SP_UPDATECOURSE

Status: Active

Input/Output

	Name	Data type	Description
→@	Old_Data	nvarchar(50)	
→@	New_Data	nvarchar(50)	

Uses

Name
⚙ dbo.SP_UPDATECOURSE
▦ dbo.Course

Script

```
create proc [dbo].[SP_UPDATECOURSE] (@Old_Data nvarchar(50) , @New_Data nvarchar(50))
as
if(ISNUMERIC(@Old_Data)=1)
begin
declare @id int;
select @id = CONVERT(int,@Old_Data)
update Course set Course_Name=@New_Data
where Course.Course_ID=@id
end
else
begin
update Course set Course_Name=@New_Data
where Course.Course_Name=@Old_Data
end
```

2.2.41. Procedure: dbo.SP_UPDATEDEP

Status: Active

Input/Output

Name		Data type	Description
→@	id	int	
→@	Dept_name	nvarchar(50)	
→@	Dept_desc	nvarchar(50)	

Uses

Name
⚙ dbo.SP_UPDATEDEP
▦ dbo.Department

Script

```
CREATE proc [dbo].[SP_UPDATEDEP] (@id int , @Dept_name nvarchar(50),@Dept_desc nvarchar(50))
as
if exists (select d.Dept_ID from Department as d where d.Dept_ID=@id)
  begin
    update Department set Dept_Name=@Dept_name , Dept_Desc=@Dept_desc where Dept_ID=@id
  end
  else
    begin
       select 'There is no Department with this ID to update'
    end
```

2.2.42. Procedure: dbo.SP_UPDATEDEPMANAGER

Status: Active

Input/Output

	Name	Data type	Description
→@	id	int	
→@	MANAGER_Dept_id	int	
→@	HIRE_DATE	date	

Uses

Name
dbo.SP_UPDATEDEPMANAGER
dbo.Department
dbo.Instructor

Script

```
CREATE proc [dbo].[SP_UPDATEDEPMANAGER] (@id int , @MANAGER_Dept_id int , @HIRE_DATE date)
as
if exists (select d.Dept_ID from Department as d where d.Dept_ID=@id)
  begin
      if exists(select ins.Dept_ID from Instructor as ins where ins.Ins_ID=@MANAGER_Dept_id)
         begin
                          update Department set Manager_ID=@MANAGER_Dept_id , Mng_hire_Date=@HIRE_DATE where Dept_ID=@id
                                 select 'Manager Involved'
                       end
      else
                begin
                         select 'There is no instructor with this id'
         end
  end
  else
    begin
       select 'There is no Department with this ID to update'
    end
```

2.2.43. Procedure: dbo.SP_UPDATEDEPTCOURSESDURATION

Status: Active

Input/Output

Name	Data type	Description
dept_ID	nvarchar(50)	
course_ID	nvarchar(50)	
duration	nvarchar(50)	

Uses

Name
dbo.SP_UPDATEDEPTCOURSESDURATION
dbo.Dept_Course

Script

```
CREATE proc [dbo].[SP_UPDATEDEPTCOURSESDURATION] (@dept_ID nvarchar(50) , @course_ID nvarchar(50),@duration nvarchar(50))
           as
                      if(ISNUMERIC(@dept_ID)=1)
                                 begin
                                            if(ISNUMERIC(@course_ID)=1)
                                                       begin
                                                                  if exists (select * from Dept_Course as dc where
dc.Dept_ID=@dept_ID and dc.Course_ID=@course_ID)
                                                                             begin
                                                                                        update Dept_Course set
Course_Duration=@duration where Dept_ID=@dept_ID and Course_ID=@course_ID
                                                                                        select 'Done, data updated'
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'There is no data with these
inputs'
                                                                             end
                                                       end
                                            else
                                                       begin
                                                                  select 'check the course ID'
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check the department ID'
                                 end
```

2.2.44. Procedure: dbo.SP_UPDATEEXAMQUESTION

Status: Active

Input/Output

	Name	Data type	Description
→@	exam_ID	nvarchar(50)	
→@	old_Q	nvarchar(50)	
→@	new_Q	nvarchar(50)	

Uses

Name
dbo.SP_UPDATEEXAMQUESTION
dbo.Exam_Questions
dbo.Question

Script

```
CREATE proc [dbo].[SP_UPDATEEXAMQUESTION] (@exam_ID nvarchar(50) , @old_Q nvarchar(50) , @new_Q nvarchar(50))
           as
                      if(ISNUMERIC(@new_Q)=0)
                                 begin
                                            if exists(select * from Question as q where q.Q_Desc=@new_Q)
                                                       begin
                                                                  declare @new_Q_ID nvarchar(50);
                                                                  select @new_Q_ID = q.Q_ID from Question AS q where
q.Q_Desc=@new_Q
                                                                  if(ISNUMERIC(@exam_ID)=1)
                                                                             begin
                                                                                        if(ISNUMERIC(@old_Q)=0)
                                                                                                   begin
                                                                                                              if exists
(select * from Exam_Questions as eq where eq.Exam_ID=@exam_ID and eq.Exam_Q=@old_Q)

           begin









                                                                                                                         end
                                                                                                              else

           begin

                                                                                                                         end
                                                                                                   end
                                                                                        else
                                                                                                   begin
                                                                                                              select 'Check
the data !!'
                                                                                                   end
                                                                             end
                                                                  else
                                                                             begin
                                                                                        select 'Check the data  !!'
                                                                             end

                                                       end
                                            else
                                                       begin
                                                                  select 'There is no question with this description'
                                                       end
                                 end
                      else
                                 begin
                                            select 'Check the data !!'
                                 end
```

2.2.45. Procedure: dbo.SP_UPDATEINSCOURSE

Status: Active

Input/Output

Name	Data type	Description
ins_ID	nvarchar(50)	
Course_ID	nvarchar(50)	
NEW_INS_ID	nvarchar(50)	

Uses

Name
dbo.SP_UPDATEINSCOURSE
dbo.Ins_Course
dbo.Instructor

Script

```
create proc [dbo].[SP_UPDATEINSCOURSE] (@ins_ID nvarchar(50),@Course_ID nvarchar(50),@NEW_INS_ID nvarchar(50))
    as
        if(ISNUMERIC(@NEW_INS_ID)=1)
            begin
                if exists (select * from Instructor as i where i.Ins_ID=@NEW_INS_ID)
                    begin
                        if(ISNUMERIC(@ins_ID)=1)
                            begin
                                if(ISNUMERIC(@Course_ID)=1)
                                    begin
                                        if exists
(select * from Ins_Course as ic where ic.Course_ID=@Course_ID and ic.Ins_ID=@ins_ID)

    begin


                                            end
                                        else

    begin

                                            end
                                    end
                                else
                                    begin
                                        select 'Check
the data !!'
                                    end
                            end
                        else
                            begin
                                select 'Check the data !!'
                            end

                    end
                else
                    begin
                        select 'This instructor id not involved !!'
                    end
            end
        else
            begin
                select 'Check the data !!'
            end
```

2.2.46. Procedure: dbo.SP_UPDATEINSTRACTOR

Status: Active

Input/Output

Name		Data type	Description
→@	id	int	
→@	DATA	nvarchar(50)	
→@	Email	nvarchar(50)	
→@	Salary	int	

Uses

Name
⚙ dbo.SP_UPDATEINSTRACTOR
▦ dbo.Instructor

Script

```
CREATE proc [dbo].[SP_UPDATEINSTRACTOR] ( @id int ,@DATA nvarchar (50) = null ,@Email nvarchar(50)=null, @Salary int = null
)
as
if exists (select i.Ins_ID from Instructor as i  where i.Ins_ID=@id)
  begin
         update Instructor set Ins_Address=@Data,Ins_Email=@Email,Ins_Salary=@Salary where Ins_ID=@id
                select 'Manager Involved'
                      end
      else
                begin
                         select 'There is no instructor with this id'
         end
```

2.2.47. Procedure: dbo.SP_UPDATEINTOTOPIC

Status: Active

Input/Output

	Name	Data type	Description
→@	id	int	
→@	Name	nvarchar(50)	

Uses

Name
dbo.SP_UPDATEINTOTOPIC
dbo.topic

Script

```
create proc [dbo].[SP_UPDATEINTOTOPIC] (@id int , @Name nvarchar(50))
as
if exists (select ID from  topic where ID= @id)
 begin
  update Topic set Name=@Name where ID= @id
 end
  else
   begin
   select 'invalid id !'
   end
```

2.2.48. Procedure: dbo.SP_UPDATESTDADRESS

Status: Active

Input/Output

	Name	Data type	Description
→@	id	int	
→@	Address	nvarchar(50)	

Uses

Name
⚙ dbo.SP_UPDATESTDADRESS
▦ dbo.Student

Script

```
Create proc [dbo].[SP_UPDATESTDADRESS] (@id int , @Address nvarchar(50))

as
if exists (select s.Std_ID from Student as s  where s.Std_ID=@id)
 begin
         update Student set Std_Address=@Address where Std_ID=@id
                select 'Colum Updated '
                      end
      else
                begin
                         select 'There is no Student with this id !'
         end
```

2.2.49. Procedure: dbo.SP_UPDATESTDEMAIL

Status: Active

Input/Output

	Name	Data type	Description
→@	id	int	
→@	Email	nvarchar(50)	

Uses

Name
⚙ dbo.SP_UPDATESTDEMAIL
▦ dbo.Student

Script

```
create proc [dbo].[SP_UPDATESTDEMAIL] (@id int , @Email nvarchar(50))

as
if exists (select s.Std_ID from Student as s  where s.Std_ID=@id)
 begin
        update Student set Std_Email=@Email where Std_ID=@id
                select 'Colum Updated '
                      end
      else
                begin
                         select 'There is no student with this id !'
         end
```

2.2.50. Procedure: dbo.SP_UPDATETOPIC

Status: Active

Input/Output

	Name	Data type	Description
→@	Old_Data	nvarchar(50)	
→@	New_Data	nvarchar(50)	

Uses

Name
dbo.SP_UPDATETOPIC
dbo.Topic

Script

```
create proc [dbo].[SP_UPDATETOPIC] (@Old_Data nvarchar(50) , @New_Data nvarchar(50))
as
if(ISNUMERIC(@Old_Data)=1)
begin
declare @id int;
select @id = CONVERT(int,@Old_Data)
update Topic set Name=@New_Data
where Topic.ID=@id
end
else
begin
update Topic set Name=@New_Data
where topic.Name=@Old_Data
end
```

2.2.51. Procedure: dbo.SP_UPDATETYPE

Status: Active

Input/Output

Name		Data type	Description
→@	Old_Data	nvarchar(50)	
→@	New_Data	nvarchar(50)	

Uses

Name
⚙ dbo.SP_UPDATETYPE
▦ dbo.Type

Script

```
create proc [dbo].[SP_UPDATETYPE] (@Old_Data nvarchar(50) , @New_Data nvarchar(50))
as
if(ISNUMERIC(@Old_Data)=1)
begin
declare @id int;
select @id = CONVERT(int,@Old_Data)
update Type set T_Desc=@New_Data
where T_ID=@id
end
else
begin
update Type set T_Desc=@New_Data
where T_Desc=@Old_Data
end
```

2.2.52. Procedure: dbo.SP_UPDATEXAMDATE

Status: Active

Input/Output

	Name	Data type	Description
→@	id	int	
→@	Date	date	

Uses

Name
⚙ dbo.SP_UPDATEXAMDATE
▦ dbo.Exam

Script

```
Create proc [dbo].[SP_UPDATEXAMDATE](@id int , @Date date)

as
if exists (select E.Exam_ID from Exam as E  where E.Exam_ID=@id)
 begin
         update Exam set Exam_Date=@Date where Exam_ID=@id
                select 'Colum Updated '
                      end
      else
                begin
                         select 'There is no Exam with this id !'
         end
```