Online Examination System.

OnlineExamDB

Team members:
1-Amany Mohamed Ahmed.
2-Hend Shaaban Ali.
3-Mina Medhat Mamdouh.
4-Nagwa Ahmed Mahmoud.
5-Nevine Ahmed Mahmoud Seddik.

Table of Contents

(local)

Databases (1)

- OnlineExamDB

Server Properties

Property	Value
Product	Microsoft SQL Server
Version	12.0.5000.0
Language	English (United States)
Platform	NT x64
Edition	Enterprise Edition (64-bit)
Engine Edition	3 (Enterprise)
Processors	4
OS Version	6.3 (18362)
Physical Memory	7365
Is Clustered	False
Root Directory	C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL
Collation	SQL_Latin1_General_CP1_CI_AS

Server Settings

Property	Value
Default data file path	C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\DATA\
Default backup file path	C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\Backup
Default log file path	C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\DATA\
Recovery Interval (minutes)	0
Default index fill factor	0
Default backup media retention	0
Compress Backup	False

Advanced Server Settings

Property	Value
Full text upgrade option	2

Locks	0
Nested triggers enabled	True
Allow triggers to fire others	True
Default language	English
Network packet size	4096
Default fulltext language LCID	1033
Two-digit year cutoff	2049
Remote login timeout	10
Cursor threshold	-1
Max text replication size	65536
Parallelism cost threshold	5
Max degree of parallelism	0
Min server memory	16
Max server memory	2147483647
Blocked process threshold	0

⬜ User databases

Databases (1)

- 🗄 OnlineExamDB

▤ OnlineExamDB Database

MS_Description

Online Examination System

Database Properties

Property	Value
SQL Server Version	SQL Server 2014
Compatibility Level	SQL Server 2012
Last backup time	-
Last log backup time	-
Creation date	Mar 1 2020
Users	4
Database Encryption Enabled	False
Database Encryption Algorithm	None
Database size	10.75 MB
Unallocated space	0.92 MB

Database Options

Property	Value
Compatibility Level	110
Database collation	SQL_Latin1_General_CP1_CI_AS
Restrict access	MULTI_USER
Auto shrink	False
Database status	ONLINE
Cleanly shutdown	False
Snapshot isolation state	OFF
Recovery model	FULL
Page verify option	CHECKSUM
Auto create statistics	True
Auto update statistics	True
Auto update statistics asynchronously	False
ANSI NULL default	False
Fulltext enabled	True
Trustworthy	False
Database chaining	False

Service broker GUID	b8e3a917-3928-4f88-85c3-fa7811e19b1d
Log reuse wait	LOG_BACKUP
Encrypted	False
Default language	English
Default fulltext language LCID	1033
Nested triggers enabled	True
Two-digit year cutoff	2049
Containment	NONE
Target recovery time	0
Database owner	DESKTOP-65N36O4\amany

Files

Name	Type	Size	Maxsize	Autogrowth	File Name
OnlineExamDB	Data	4.00 MB	unlimited	1.00 MB	C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\DATA\OnlineExamDB.mdf
OnlineExamDB_log	Log	6.75 MB	2048.00 GB	10.00 percent	C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER\MSSQL\DATA\OnlineExam-DB_log.ldf

---

📰 *Tables*

Objects

Name
ExamSystem.Course *Show Course Name & Id of Instructor who Teach This Course*
ExamSystem.Exam *Show Exam Id Duration ,Time,Student who take this Exam,Course id that student take Exam for and Total grade of This student in This Exam*
ExamSystem.Exam_Questions *Each Exam has many Question & a Question comes In Many Exams ,* *So WE Store Each Exam with Its Questions embedded in.*
ExamSystem.Instructor *Store Instructor Name & his ID*
ExamSystem.Question *Store Question ID , Question Itself , Course id that Belongs to ,Model Answer of This Question ,& its choices.* *(Choices used in Exam to Show to Student To Choose from it)*
ExamSystem.Stud_Exam_Quest *This Table Used To Store Student Id with his Exam That He Take ,Questions in this Exam , Student Answer on these Question , and Grade in this Question.*
ExamSystem.Student *Store Data of Student(ID,First Name,Last Name ,Address,Age ,& Track ID)*
ExamSystem.Student_Courses *This Table Show Relation between Student &Courses That He Study.* *It Store Each Student With Course He Studies.*
ExamSystem.Topic *Each Course Has Many Topics In It So THis Table Stores Data About Topic Like (id , Name of Topic & Course That Is Belong to It)*
ExamSystem.Track *ITI has Many Tracks That a Student register in it.* *Data About Track Like ..(Id ,Name , Manager Name , Duration of this Track &number of Studet In It).*
ExamSystem.Track_Courses *A Track has Many Courses In it & a Course Exists in Many Tracks so this Table Store Each Track With Its Courses.*

🖾 [ExamSystem].[Course]

MS_Description

Show Course Name & Id of Instructor who Teach This Course

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	2
Created	11:59:33 PM Saturday, February 29, 2020
Last Modified	12:11:40 AM Sunday, March 1, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity
PK C	Course_ID	int	4	NOT NULL	1 - 1
	Course_Name	nvarchar(50)	100	NULL allowed	
FK	Instructor_ID	int	4	NULL allowed	

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Course	Course_ID	True

Foreign Keys

Name	Update	Delete	Columns
FK_Course_Instructor	SetNull	SetNull	Instructor_ID->[ExamSystem].[Instructor].[Instructor_ID]

SQL Script

```
CREATE TABLE [ExamSystem].[Course]
(
[Course_ID] [int] NOT NULL IDENTITY(1, 1),
[Course_Name] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Instructor_ID] [int] NULL
) ON [PRIMARY]
```

```
GO
ALTER TABLE [ExamSystem].[Course] ADD CONSTRAINT [PK_Course] PRIMARY KEY CLUSTERED
([Course_ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Course] ADD CONSTRAINT [FK_Course_Instructor] FOREIGN KEY
([Instructor_ID]) REFERENCES [ExamSystem].[Instructor] ([Instructor_ID]) ON DELETE SET
NULL ON UPDATE SET NULL
GO
EXEC sp_addextendedproperty N'MS_Description', N'Show Course Name & Id of Instructor
who Teach This Course', 'SCHEMA', N'ExamSystem', 'TABLE', N'Course', NULL, NULL
GO
```

Uses

[ExamSystem].[Instructor]
ExamSystem

Used By

[ExamSystem].[Exam]
[ExamSystem].[Question]
[ExamSystem].[Student_Courses]
[ExamSystem].[Topic]
[ExamSystem].[Track_Courses]
[dbo].[Course_With_Its_Students]
[dbo].[Delete_Course]
[dbo].[generate_Exam_check]
[dbo].[Insert_Course]
[dbo].[Select_Course]
[dbo].[Student_Grades_Report]
[dbo].[Update_Course]

ALTER TABLE [ExamSystem].[Course] ADD CONSTRAINT [PK_Course] PRIMARY KEY CLUSTERED

[ExamSystem].[Exam]

MS_Description

Show Exam Id Duration ,Time,Student who take this Exam,Course id that student take Exam for and Total grade of This student in This Exam

Properties

Property	Value
Row Count (~)	5
Created	3:38:38 PM Thursday, February 27, 2020
Last Modified	12:08:37 AM Sunday, March 1, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability
PK C	Exam_ID	int	4	NOT NULL
	Exam_Duration	int	4	NULL allowed
	Exam_DateTime	datetime	8	NULL allowed
FK	Std_ID	int	4	NULL allowed
FK	Course_ID	int	4	NULL allowed
	Total_Grade	int	4	NULL allowed

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Exam	Exam_ID	True

Foreign Keys

Name	Update	Delete	Columns
FK_Exam_Course	SetNull	SetNull	Course_ID->[ExamSystem].[Course].[Course_ID]
FK_Exam_Student	SetNull	SetNull	Std_ID->[ExamSystem].[Student].[Std_ID]

SQL Script

```
CREATE TABLE [ExamSystem].[Exam]
(
[Exam_ID] [int] NOT NULL,
[Exam_Duration] [int] NULL,
[Exam_DateTime] [datetime] NULL,
[Std_ID] [int] NULL,
[Course_ID] [int] NULL,
[Total_Grade] [int] NULL
) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Exam] ADD CONSTRAINT [PK_Exam] PRIMARY KEY CLUSTERED  ([Exam_-
ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Exam] ADD CONSTRAINT [FK_Exam_Course] FOREIGN KEY ([Course_-
ID]) REFERENCES [ExamSystem].[Course] ([Course_ID]) ON DELETE SET NULL ON UPDATE SET
NULL
GO
ALTER TABLE [ExamSystem].[Exam] ADD CONSTRAINT [FK_Exam_Student] FOREIGN KEY ([Std_ID])
REFERENCES [ExamSystem].[Student] ([Std_ID]) ON DELETE SET NULL ON UPDATE SET NULL
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Show Exam Id Duration ,Time,Student who take this Exam,Course id that student take Exam
for and Total grade of
This student in This Exam', 'SCHEMA', N'ExamSystem', 'TABLE', N'Exam', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]
[ExamSystem].[Student]
ExamSystem

Used By

[ExamSystem].[Exam_Questions]
[ExamSystem].[Stud_Exam_Quest]
[dbo].[Delete_Exam]
[dbo].[Exam_Correction]
[dbo].[generate_Exam_check]
[dbo].[Insert_Exam]
[dbo].[Select_Exam]
[dbo].[Student_Grades_Report]
[dbo].[Update_Exam]

▦ [ExamSystem].[Exam_Questions]

MS_Description

Each Exam has many Question & a Question comes In Many Exams ,
So WE Store Each Exam with Its Questions embedded in.

Properties

Property	Value
Row Count (~)	50
Created	2:15:55 AM Sunday, February 23, 2020
Last Modified	3:38:39 PM Thursday, February 27, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability
PK C FK	Exam_ID	int	4	NOT NULL
PK C FK	Question_ID	int	4	NOT NULL

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Exam_Questions	Exam_ID, Question_ID	True

Foreign Keys

Name	Columns
FK_Exam_Questions_Exam	Exam_ID->[ExamSystem].[Exam].[Exam_ID]
FK_Exam_Questions_Question	Question_ID->[ExamSystem].[Question].[Question_ID]

SQL Script

```
CREATE TABLE [ExamSystem].[Exam_Questions]
(
[Exam_ID] [int] NOT NULL,
```

```
[Question_ID] [int] NOT NULL
) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Exam_Questions] ADD CONSTRAINT [PK_Exam_Questions] PRIMARY
KEY CLUSTERED  ([Exam_ID], [Question_ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Exam_Questions] ADD CONSTRAINT [FK_Exam_Questions_Exam]
FOREIGN KEY ([Exam_ID]) REFERENCES [ExamSystem].[Exam] ([Exam_ID])
GO
ALTER TABLE [ExamSystem].[Exam_Questions] ADD CONSTRAINT [FK_Exam_Questions_Question]
FOREIGN KEY ([Question_ID]) REFERENCES [ExamSystem].[Question] ([Question_ID])
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Each Exam has many Question & a Question comes In Many Exams ,
So WE Store Each Exam with Its Questions embedded in.', 'SCHEMA', N'ExamSystem',
'TABLE', N'Exam_Questions', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam]
[ExamSystem].[Question]
ExamSystem

Used By

[dbo].[Exam_Answers]
[dbo].[Exam_With_Questions_Report]
[dbo].[generate_Exam_check]

📑 [ExamSystem].[Instructor]

MS_Description

Store Instructor Name & his ID

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	12
Created	12:55:32 AM Sunday, February 23, 2020
Last Modified	11:59:33 PM Saturday, February 29, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity
PK C	Instructor_ID	int	4	NOT NULL	1 - 1
	Instructor_Name	nvarchar(50)	100	NULL allowed	

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Instructor	Instructor_ID	True

SQL Script

```sql
CREATE TABLE [ExamSystem].[Instructor]
(
[Instructor_ID] [int] NOT NULL IDENTITY(1, 1),
[Instructor_Name] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL
) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Instructor] ADD CONSTRAINT [PK_Instructor] PRIMARY KEY
CLUSTERED  ([Instructor_ID]) ON [PRIMARY]
GO
EXEC sp_addextendedproperty N'MS_Description', N'Store Instructor Name & his ID',
'SCHEMA', N'ExamSystem', 'TABLE', N'Instructor', NULL, NULL
GO
```

Uses

ExamSystem

Used By

[ExamSystem].[Course]
[dbo].[Course_With_Its_Students]
[dbo].[Delete_Instructor]
[dbo].[Insert_Instructor]
[dbo].[Select_Instructor]
[dbo].[Update_Instructor]

📋 [ExamSystem].[Question]

MS_Description

Store Question ID , Question Itself , Course id that Belongs to ,Model Answer of This Question ,& its choices.
(Choices used in Exam to Show to Student To Choose from it)

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	41
Created	2:11:56 AM Sunday, February 23, 2020
Last Modified	12:08:37 AM Sunday, March 1, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity
PK C	Question_ID	int	4	NOT NULL	1 - 1
	Question_Text	nvarchar(max)	max	NULL allowed	
	Model_Answer	nvarchar(max)	max	NULL allowed	
	Choise1	nvarchar(max)	max	NOT NULL	
	Choise2	nvarchar(max)	max	NOT NULL	
	Choise3	nvarchar(max)	max	NULL allowed	
	Choise4	nvarchar(max)	max	NULL allowed	
	Question_Type	nvarchar(50)	100	NULL allowed	
FK	Course_ID	int	4	NULL allowed	

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Question	Question_ID	True

Foreign Keys

Name	Update	Delete	Columns
FK_Question_Course	SetNull	SetNull	Course_ID->[ExamSystem].[Course].[Course_ID]

SQL Script

```sql
CREATE TABLE [ExamSystem].[Question]
(
[Question_ID] [int] NOT NULL IDENTITY(1, 1),
[Question_Text] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Model_Answer] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Choise1] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
[Choise2] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
[Choise3] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Choise4] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Question_Type] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Course_ID] [int] NULL
) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Question] ADD CONSTRAINT [PK_Question] PRIMARY KEY CLUSTERED
([Question_ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Question] ADD CONSTRAINT [FK_Question_Course] FOREIGN KEY
([Course_ID]) REFERENCES [ExamSystem].[Course] ([Course_ID]) ON DELETE SET NULL ON
UPDATE SET NULL
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Store Question ID , Question Itself , Course id that Belongs to ,Model Answer of This
Question ,& its choices.
(Choices used in Exam to Show to Student To Choose from it )', 'SCHEMA', N'ExamSystem',
'TABLE', N'Question', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]
ExamSystem

Used By

[ExamSystem].[Exam_Questions]
[ExamSystem].[Stud_Exam_Quest]
[dbo].[Delete_Question]
[dbo].[Exam_Answers]
[dbo].[Exam_Correction]
[dbo].[Exam_With_Questions_Report]
[dbo].[generate_Exam_check]
[dbo].[Insert_Question]
[dbo].[Select_Question]
[dbo].[Update_Question]

Project > (local) > User databases > OnlineExamDB > Tables > ExamSystem.Stud_Exam_Quest   Project SQL & BI.

ITI Alex.

Intake 40.

📇 [ExamSystem].[Stud_Exam_Quest]

MS_Description

This Table Used To Store Student Id with his Exam That He Take ,Questions in this Exam , Student Answer on these Question , and Grade in this Question.

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	50
Created	2:20:28 AM Sunday, February 23, 2020
Last Modified	3:38:39 PM Thursday, February 27, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Default
PK C FK	Std_ID	int	4	NOT NULL	
PK C FK	Exam_ID	int	4	NOT NULL	
PK C FK	Question_ID	int	4	NOT NULL	
	Question_Answer	nvarchar(max)	max	NULL allowed	
	Question_Grade	int	4	NULL allowed	((0))

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Stud_Exam_Quest	Std_ID, Exam_ID, Question_ID	True

Foreign Keys

Name	Columns
FK_Stud_Exam_Quest_Exam	Exam_ID->[ExamSystem].[Exam].[Exam_ID]
FK_Stud_Exam_Quest_Question	Question_ID->[ExamSystem].[Question].[Question_ID]

FK_Stud_Exam_Quest_Student	Std_ID->[ExamSystem].[Student].[Std_ID]

SQL Script

```sql
CREATE TABLE [ExamSystem].[Stud_Exam_Quest]
(
[Std_ID] [int] NOT NULL,
[Exam_ID] [int] NOT NULL,
[Question_ID] [int] NOT NULL,
[Question_Answer] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Question_Grade] [int] NULL CONSTRAINT [DF_Stud_Exam_Quest_Question_Grade] DEFAULT
((0))
) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Stud_Exam_Quest] ADD CONSTRAINT [PK_Stud_Exam_Quest] PRIMARY
KEY CLUSTERED  ([Std_ID], [Exam_ID], [Question_ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Stud_Exam_Quest] ADD CONSTRAINT [FK_Stud_Exam_Quest_Exam]
FOREIGN KEY ([Exam_ID]) REFERENCES [ExamSystem].[Exam] ([Exam_ID])
GO
ALTER TABLE [ExamSystem].[Stud_Exam_Quest] ADD CONSTRAINT [FK_Stud_Exam_Quest_Question]
FOREIGN KEY ([Question_ID]) REFERENCES [ExamSystem].[Question] ([Question_ID])
GO
ALTER TABLE [ExamSystem].[Stud_Exam_Quest] ADD CONSTRAINT [FK_Stud_Exam_Quest_Student]
FOREIGN KEY ([Std_ID]) REFERENCES [ExamSystem].[Student] ([Std_ID])
GO
EXEC sp_addextendedproperty N'MS_Description', N'
This Table Used To Store Student Id with his Exam That He Take ,Questions in this Exam
, Student Answer on these
Question , and Grade in this Question.', 'SCHEMA', N'ExamSystem', 'TABLE', N'Stud_Exam_-
Quest', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam]
[ExamSystem].[Question]
[ExamSystem].[Student]
ExamSystem

Used By

[dbo].[Delete_Stud_Exam_Quest]
[dbo].[Exam_Answer_For_Student]
[dbo].[Exam_Answers]
[dbo].[Exam_Correction]
[dbo].[generate_Exam_check]
[dbo].[Insert_Stud_Exam_Quest]
[dbo].[Select_Stud_Exam_Quest]

[dbo].[Update_Stud_Exam_Quest]

📇 [ExamSystem].[Student]

MS_Description

Store Data of Student(ID,First Name,Last Name ,Address,Age ,& Track ID)

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	18
Created	12:49:01 AM Sunday, February 23, 2020
Last Modified	3:38:38 PM Thursday, February 27, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity
PK C	Std_ID	int	4	NOT NULL	1 - 1
	Std_FName	nvarchar(50)	100	NULL allowed	
	Std_LName	nvarchar(50)	100	NULL allowed	
	Std_Address	nvarchar(max)	max	NULL allowed	
	Std_Age	int	4	NULL allowed	
	Track_ID	int	4	NULL allowed	

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Student	Std_ID	True

SQL Script

```
CREATE TABLE [ExamSystem].[Student]
(
[Std_ID] [int] NOT NULL IDENTITY(1, 1),
[Std_FName] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Std_LName] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Std_Address] [nvarchar] (max) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Std_Age] [int] NULL,
[Track_ID] [int] NULL
) ON [PRIMARY]
```

```
GO
ALTER TABLE [ExamSystem].[Student] ADD CONSTRAINT [PK_Student] PRIMARY KEY CLUSTERED
([Std_ID]) ON [PRIMARY]
GO
EXEC sp_addextendedproperty N'MS_Description', N'Store Data of Student(ID,First
Name,Last Name ,Address,Age ,& Track ID)', 'SCHEMA', N'ExamSystem', 'TABLE',
N'Student', NULL, NULL
GO
```

Uses

ExamSystem

Used By

[ExamSystem].[Exam]
[ExamSystem].[Stud_Exam_Quest]
[ExamSystem].[Student_Courses]
[dbo].[Delete_Student]
[dbo].[Insert_Student]
[dbo].[Select_Student]
[dbo].[Student_Info_Report]
[dbo].[Update_Student]

▦ [ExamSystem].[Student_Courses]

MS_Description

This Table Show Relation between Student &Courses That He Study.
It Store Each Student With Course He Studies.

Properties

Property	Value
Row Count (~)	4
Created	7:25:26 PM Tuesday, February 25, 2020
Last Modified	12:11:40 AM Sunday, March 1, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability
PK C FK	Student_ID	int	4	NOT NULL
PK C FK	Course_ID	int	4	NOT NULL

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Student_Courses	Student_ID, Course_ID	True

Foreign Keys

Name	Columns
FK_Student_Courses_Course	Course_ID->[ExamSystem].[Course].[Course_ID]
FK_Student_Courses_Student	Student_ID->[ExamSystem].[Student].[Std_ID]

SQL Script

```
CREATE TABLE [ExamSystem].[Student_Courses]
(
[Student_ID] [int] NOT NULL,
```

```
[Course_ID] [int] NOT NULL
) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Student_Courses] ADD CONSTRAINT [PK_Student_Courses] PRIMARY
KEY CLUSTERED  ([Student_ID], [Course_ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Student_Courses] ADD CONSTRAINT [FK_Student_Courses_Course]
FOREIGN KEY ([Course_ID]) REFERENCES [ExamSystem].[Course] ([Course_ID])
GO
ALTER TABLE [ExamSystem].[Student_Courses] ADD CONSTRAINT [FK_Student_Courses_Student]
FOREIGN KEY ([Student_ID]) REFERENCES [ExamSystem].[Student] ([Std_ID])
GO
EXEC sp_addextendedproperty N'MS_Description', N'
This Table Show Relation between Student &Courses That He Study.
It Store Each Student With Course He Studies.', 'SCHEMA', N'ExamSystem', 'TABLE',
N'Student_Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]
[ExamSystem].[Student]
ExamSystem

Used By

[dbo].[Course_With_Its_Students]
[dbo].[Delete_Student_Courses]
[dbo].[Insert_Student_Courses]
[dbo].[Select_Student_Courses]
[dbo].[Update_Student_Courses]

📄 [ExamSystem].[Topic]

MS_Description

Each Course Has Many Topics In It So THis Table Stores Data About Topic Like (id , Name of Topic & Course That Is Belong to It)

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	17
Created	12:52:18 AM Sunday, February 23, 2020
Last Modified	12:08:37 AM Sunday, March 1, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity
PK C	Topic_ID	int	4	NOT NULL	1 - 1
	Topic_Name	nvarchar(50)	100	NULL allowed	
FK	Course_ID	int	4	NULL allowed	

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Topic	Topic_ID	True

Foreign Keys

Name	Update	Delete	Columns
FK_Topic_Course	SetNull	SetNull	Course_ID->[ExamSystem].[Course].[Course_ID]

SQL Script

```
CREATE TABLE [ExamSystem].[Topic]
(
[Topic_ID] [int] NOT NULL IDENTITY(1, 1),
[Topic_Name] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
```

Project > (local) > User databases > OnlineExamDB > Tables > ExamSystem.Topic   Project SQL & BI.

ITI Alex.

Intake 40.

```
[Course_ID] [int] NULL
) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Topic] ADD CONSTRAINT [PK_Topic] PRIMARY KEY CLUSTERED
([Topic_ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Topic] ADD CONSTRAINT [FK_Topic_Course] FOREIGN KEY ([Course_-
ID]) REFERENCES [ExamSystem].[Course] ([Course_ID]) ON DELETE SET NULL ON UPDATE SET
NULL
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Each Course Has Many Topics In It So THis Table Stores Data About Topic Like (id , Name
of Topic & Course That Is
Belong to It )', 'SCHEMA', N'ExamSystem', 'TABLE', N'Topic', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]
ExamSystem

Used By

[dbo].[Course_With_Topics_Report]
[dbo].[Delete_Topic]
[dbo].[Insert_Topic]
[dbo].[Select_Topic]
[dbo].[Update_Topic]

🖽 [ExamSystem].[Track]

MS_Description

ITI has Many Tracks That a Student register in it.
Data About Track Like ..(Id ,Name , Manager Name , Duration of this Track &number of Studet In It).

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	9
Created	12:24:21 AM Sunday, February 23, 2020
Last Modified	12:10:05 AM Sunday, March 1, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity
PK C	Track_ID	int	4	NOT NULL	1 - 1
	Track_Name	nvarchar(50)	100	NULL allowed	
	Track_Manager	nvarchar(50)	100	NULL allowed	
	Track_Duration	int	4	NULL allowed	
	No_Of_Stud	int	4	NULL allowed	

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Track	Track_ID	True

SQL Script

```
CREATE TABLE [ExamSystem].[Track]
(
[Track_ID] [int] NOT NULL IDENTITY(1, 1),
[Track_Name] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Track_Manager] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
[Track_Duration] [int] NULL,
[No_Of_Stud] [int] NULL
) ON [PRIMARY]
GO
```

```
ALTER TABLE [ExamSystem].[Track] ADD CONSTRAINT [PK_Track] PRIMARY KEY CLUSTERED
([Track_ID]) ON [PRIMARY]
GO
EXEC sp_addextendedproperty N'MS_Description', N'
ITI has Many Tracks That a Student register in it.
Data About Track Like ..( Id ,Name , Manager Name , Duration of this Track &number of
Studet In It).', 'SCHEMA', N'ExamSystem', 'TABLE', N'Track', NULL, NULL
GO
```

Uses

ExamSystem

Used By

[ExamSystem].[Track_Courses]
[dbo].[Delete_Student]
[dbo].[Delete_Track]
[dbo].[Insert_Student]
[dbo].[Insert_Track]
[dbo].[Select_Track]
[dbo].[Update_Student]
[dbo].[Update_Track]

🖽 [ExamSystem].[Track_Courses]

MS_Description

A Track has Many Courses In it & a Course Exists in Many Tracks so this Table Store Each Track With Its Courses.

Properties

Property	Value
Row Count (~)	2
Created	2:05:41 AM Sunday, February 23, 2020
Last Modified	12:10:05 AM Sunday, March 1, 2020

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability
PK C FK	Track_ID	int	4	NOT NULL
PK C FK	Course_ID	int	4	NOT NULL

Indexes

Key	Name	Key Columns	Unique
PK C	PK_Track_Courses	Track_ID, Course_ID	True

Foreign Keys

Name	Columns
FK_Track_Courses_Course	Course_ID->[ExamSystem].[Course].[Course_ID]
FK_Track_Courses_Track	Track_ID->[ExamSystem].[Track].[Track_ID]

SQL Script

```
CREATE TABLE [ExamSystem].[Track_Courses]
(
[Track_ID] [int] NOT NULL,
[Course_ID] [int] NOT NULL
) ON [PRIMARY]
```

```
GO
ALTER TABLE [ExamSystem].[Track_Courses] ADD CONSTRAINT [PK_Track_Courses] PRIMARY KEY
CLUSTERED  ([Track_ID], [Course_ID]) ON [PRIMARY]
GO
ALTER TABLE [ExamSystem].[Track_Courses] ADD CONSTRAINT [FK_Track_Courses_Course]
FOREIGN KEY ([Course_ID]) REFERENCES [ExamSystem].[Course] ([Course_ID])
GO
ALTER TABLE [ExamSystem].[Track_Courses] ADD CONSTRAINT [FK_Track_Courses_Track]
FOREIGN KEY ([Track_ID]) REFERENCES [ExamSystem].[Track] ([Track_ID])
GO
EXEC sp_addextendedproperty N'MS_Description', N'A Track has Many Courses In it & a
Course Exists in Many Tracks so this Table Store Each Track With Its Courses.',
'SCHEMA', N'ExamSystem', 'TABLE', N'Track_Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]
[ExamSystem].[Track]
ExamSystem

Used By

[dbo].[Insert_Track_Courses]
[dbo].[Select_Track_Courses]
[dbo].[Update_Track_Courses]

🖺 *Stored Procedures*

Objects

Name
dbo.Course_With_Its_Students *Stored Procedure used to Make A Report , it takes Instructor _ID to show Courses he Teach and Number Of Student In these Courses.*
dbo.Course_With_Topics_Report *Stored Procedure Used To Show Topics At Specific Course*
dbo.Delete_Course *Stored Procedure take Course ID That You Want To Delete.*
dbo.Delete_Exam *Stored Procedure take Exam ID That You Want To Delete.*
dbo.Delete_Instructor *Stored Procedure take Instructor ID That You Want To Delete.*
dbo.Delete_Question *Stored Procedure take Question ID That You Want To Delete.*
dbo.Delete_Stud_Exam_Quest *Stored Procedure take Student ID , Exam ID & Question ID That You Want To Delete.*
dbo.Delete_Student *Stored Procedure take Student ID That You Want To Delete.* *And Update Number Of Student In This Track (IN Track Table).*
dbo.Delete_Student_Courses *Stored Procedure take Student ID That You Want To Delete & Delete All Courses He Had Registered In Them.*
dbo.Delete_Topic *Stored Procedure take Topic hat You Want To Delete*
dbo.Delete_Track *Stored Procedure take Track ID That You Want To Delete.*
dbo.Delete_Track_Courses *Stored Procedure take Track ID That You Want To Delete Its Relation With Course Table (delete All courses in Specific Track).*
dbo.Exam_Answer_For_Student *Stored Procedure for Student To Enters Answers of Exam .*
dbo.Exam_Answers *Stored Procedure take Exam ID & Student ID That You Want To Show His Answers In Exam .*
dbo.Exam_Correction *Stored Procedure take Exam ID & Student ID That You Want To Correct His Answers In Exam .* *IT Compare Student Answers With Correct Answer In Question Table .* *It Update Grade Of Question Either 0 or 1 in table Student_Exam_Question* *And* *Update Total Grade At Exam Table.*
dbo.Exam_With_Questions_Report

Stored Procedure take Exam ID To Show Exam Questions & Its Choices AS A Report .
dbo.generate_Exam_check *Stored Procedure take Exam ID , Course_id , Student_id , Date of Exam , Duration , Number of T/F Questions & Number of MCQ Questions.* *it Generate Exam With Random Question Related To Course That Has been Chosen.*
dbo.Insert_Course *Stored Procedure take Name of Course & Instructor ID That You Want To Add.*
dbo.Insert_Exam *Stored Procedure take Course_id , Student_id , Date of Exam , Duration & Total Grade.*
dbo.Insert_Instructor *Stored Procedure take Name Of Instructor You Want To Add.*
dbo.Insert_Question *Stored Procedure To Add New Question it Take Question_Text , Model_Answer(correct) ,First_choice , Second_-Choice ,* *Third_Choice , Fourth_Choice , Question_Type(T/F-MCQ), Course_ID)*
dbo.Insert_Stud_Exam_Quest *Stored Procedure To store Student , Each Exam & Questions In It .* *It Takes Student ID , Exam ID & Questions ID , Answer Of This Question , & Question Grade .* *First Time We Could Add First 3 Parameters & Fill It In Table by Wizard.* *After That Correction Procedure Will Take Theses Answers & Compare Them With Correct &change The Grade.*
dbo.Insert_Student *Stored Procedure To Add New Student It Take Student Data like..(First Name , Last Name , Address , Age , Track_id)*
dbo.Insert_Student_Courses *Stored Procedure To Store Student With Courses He is Studied.* *It Takes Parameter..(Student_ID , Course_ID)*
dbo.Insert_Topic *Stored Procedure To Add New Topic It Take Topic Data like..(Name ,Course_id)*
dbo.Insert_Track *Stored Procedure To Add New Track It Takes Track Data like..(Name , Duration , Manager Name , & Number Of Student In It)*
dbo.Insert_Track_Courses *Stored Procedure To Store Tracks with its Courses , Takes Track ID & Course ID.*
dbo.Select_Course *Stored Procedure To select Course by ID.*
dbo.Select_Exam *Stored Procedure To select Exam by ID.*
dbo.Select_Instructor *Stored Procedure To select Instructor by ID.*
dbo.Select_Question *Stored Procedure To select Question by ID.*
dbo.Select_Stud_Exam_Quest *Stored Procedure To select Specific Student With Specific Exam & Questions in This Exam.*
dbo.Select_Student *Stored Procedure To select Student by ID.*
dbo.Select_Student_Courses

Project > (local) > User databases > OnlineExamDB > Programmability > Stored Procedures   Project SQL & BI.

ITI Alex.

Intake 40.

Stored Procedure To select All Courses to Specific Student by his ID.
dbo.Select_Topic *Stored Procedure To select Topic by ID.*
dbo.Select_Track *Stored Procedure To select Track by ID.*
dbo.Select_Track_Courses *Stored Procedure To select All Courses in Specific Tack by Track_ID.*
dbo.Student_Grades_Report *Stored Procedure that Show Each Student With his Last Grade .*
dbo.Student_Info_Report *Stored Procedure To select Students in Specific Track Takes Track_ID used in Report.*
dbo.Update_Course *Stored Procedure to Update Course And Instructor who Teach This Course.*
dbo.Update_Exam *Stored Procedure to Update Exam Its Takes All Parameter like(Exam_ID , Std_ID , Course_ID , Exam_Duration , Date of Exam , Total_Grade)*
dbo.Update_Instructor *Stored Procedure to Update Instructor Data Its Takes his ID &Name.*
dbo.Update_Question *Stored Procedure to Update Question , its Model Answer , choices , Question Type & Course of this Question.*
dbo.Update_Stud_Exam_Quest *Stored Procedure to Update Answer & Grade in Table 'Stud_Exam_Quest ' by Student_ID ,Exam_ID & Question_ID.*
dbo.Update_Student *Stored Procedure to Update Student Data Depending on Student_ID .* *It Also Update Number of Student in Table Track Depending On* *Track of This Student.(increase new track he join & decrease old track he left)*
dbo.Update_Student_Courses *Stored Procedure to Update Course That Student Study by Taking Student_ID.*
dbo.Update_Topic *Stored Procedure to Update Topic Data by Taking ID , Name & Course_ID.*
dbo.Update_Track *Stored Procedure to Update Track Name It Takes Track_ID & New Name of Track To change .*
dbo.Update_Track_Courses *Stored Procedure to Update Courses Data That Belong To Specific Track.*

🗒 [dbo].[Course_With_Its_Students]

MS_Description

Stored Procedure used to Make A Report , it takes Instructor _ID to show Courses he Teach and Number Of Student In these Courses.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@instructor_ID	int	4

SQL Script

```
CREATE proc [dbo].[Course_With_Its_Students] @instructor_ID integer
as
begin
select c.Course_Name,count(sc.Student_ID) as No_Of_Students
from [ExamSystem].[Course] c,[ExamSystem].[Student_Courses] sc,[Exam-
System].[Instructor] i
where c.Instructor_ID=i.Instructor_ID and sc.Course_ID=c.Course_ID and i.Instructor_-
ID=@instructor_ID
group by c.Course_Name
end
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure used to Make A Report , it takes Instructor _ID to show Courses he
Teach and Number Of Student
In these Courses.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Course_With_Its_Students', NULL,
NULL
GO
```

Uses

[ExamSystem].[Course]
[ExamSystem].[Instructor]
[ExamSystem].[Student_Courses]

🖹 [dbo].[Course_With_Topics_Report]

MS_Description

Stored Procedure Used To Show Topics At Specific Course

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Course_ID	int	4

SQL Script

```
create proc [dbo].[Course_With_Topics_Report]  @Course_ID INTEGER
AS
BEGIN
    select  Topic_Name from ExamSystem.Topic where Course_ID=@Course_ID
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure Used To Show Topics
At Specific Course', 'SCHEMA', N'dbo', 'PROCEDURE', N'Course_With_Topics_Report', NULL,
NULL
GO
```

Uses

[ExamSystem].[Topic]

[dbo].[Delete_Course]

MS_Description

Stored Procedure take Course ID That You Want To Delete.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Delete_Course]  @id int
AS
BEGIN
    delete from ExamSystem.Course where Course_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Course ID That
You Want To Delete.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Delete_Course', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]

[dbo].[Delete_Exam]

MS_Description

Stored Procedure take Exam ID That You Want To Delete.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Exam_ID	int	4

SQL Script

```
create   procedure [dbo].[Delete_Exam] ( @Exam_ID INTEGER)
        as
        BEGIN
            DELETE FROM ExamSystem.Exam
            WHERE  Exam_ID = @Exam_ID
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Exam ID That You
Want To Delete.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Delete_Exam', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam]

📄 [dbo].[Delete_Instructor]

MS_Description

Stored Procedure take Instructor ID That You Want To Delete.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Delete_Instructor]  @id int
AS
BEGIN
    delete from ExamSystem.Instructor where Instructor_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Instructor ID
That You Want To Delete.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Delete_Instructor', NULL,
NULL
GO
```

Uses

[ExamSystem].[Instructor]

📄 [dbo].[Delete_Question]

MS_Description

Stored Procedure take Question ID That You Want To Delete.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Question_ID	int	4

SQL Script

```
create   procedure [dbo].[Delete_Question] ( @Question_ID INTEGER)
        as
        BEGIN
            DELETE FROM ExamSystem.Question
            WHERE  Question_ID = @Question_ID
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Question ID That
You Want To Delete.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Delete_Question', NULL, NULL
GO
```

Uses

[ExamSystem].[Question]

🖹 [dbo].[Delete_Stud_Exam_Quest]

MS_Description

Stored Procedure take Student ID , Exam ID & Question ID That You Want To Delete.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Std_ID	int	4
@Exam_ID	int	4
@Question_ID	int	4

SQL Script

```
    create  procedure [dbo].[Delete_Stud_Exam_Quest] ( @Std_ID INTEGER ,@Exam_ID
INTEGER, @Question_ID INTEGER)
        as
        BEGIN
            DELETE FROM ExamSystem.Stud_Exam_Quest
            WHERE  Std_ID = @Std_ID
            AND    Exam_ID=@Exam_ID
            AND    Question_ID=@Question_ID
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Student ID ,
Exam ID & Question ID That You Want To Delete.', 'SCHEMA', N'dbo', 'PROCEDURE',
N'Delete_Stud_Exam_Quest', NULL, NULL
GO
```

Uses

[ExamSystem].[Stud_Exam_Quest]

📄 [dbo].[Delete_Student]

MS_Description

Stored Procedure take Student ID That You Want To Delete.
And Update Number Of Student In This Track (IN Track Table).

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
----------------------------------
--delete student
CREATE PROCEDURE [dbo].[Delete_Student]  @id int
AS
BEGIN
    delete from ExamSystem.Student where Std_ID=@id
END
BEGIN
    update ExamSystem.Track set  No_Of_Stud-=1    where Track_ID=1(select s.Track_ID
from ExamSystem.Student s where s.Std_ID=17)
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure take Student ID That You Want To Delete.
And Update Number Of Student In This Track ( IN Track Table).', 'SCHEMA', N'dbo',
'PROCEDURE', N'Delete_Student', NULL, NULL
GO
```

Uses

[ExamSystem].[Student]
[ExamSystem].[Track]

🖺 [dbo].[Delete_Student_Courses]

MS_Description

Stored Procedure take Student ID That You Want To Delete & Delete All Courses He Had Registered In Them.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Delete_Student_Courses]  @id int
AS
BEGIN
    delete from ExamSystem.Student_Courses where Student_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Student ID That
You Want To Delete & Delete All Courses He Had Registered In Them.', 'SCHEMA', N'dbo',
'PROCEDURE', N'Delete_Student_Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Student_Courses]

[dbo].[Delete_Topic]

MS_Description

Stored Procedure take Topic hat You Want To Delete

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Topic_ID	int	4

SQL Script

```sql
create  procedure [dbo].[Delete_Topic] ( @Topic_ID INTEGER)
        as
        BEGIN
            DELETE FROM ExamSystem.Topic
            WHERE  Topic_ID = @Topic_ID
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Topic hat You
Want To Delete', 'SCHEMA', N'dbo', 'PROCEDURE', N'Delete_Topic', NULL, NULL
GO
```

Uses

[ExamSystem].[Topic]

📄 [dbo].[Delete_Track]

MS_Description

Stored Procedure take Track ID That You Want To Delete.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Delete_Track]  @id int
AS
BEGIN
    delete from ExamSystem.Track where Track_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Track ID That
You Want To Delete.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Delete_Track', NULL, NULL
GO
```

Uses

[ExamSystem].[Track]

📄 [dbo].[Delete_Track_Courses]

MS_Description

Stored Procedure take Track ID That You Want To Delete Its Relation With Course Table (delete All courses in Specific Track).

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Delete_Track_Courses]  @id int
AS
BEGIN
    delete from SystemExam.Track_Courses where Track_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure take Track ID That You Want To Delete Its Relation With Course Table (delete All courses in
Specific Track).', 'SCHEMA', N'dbo', 'PROCEDURE', N'Delete_Track_Courses', NULL, NULL
GO
```

🖹 [dbo].[Exam_Answer_For_Student]

MS_Description

Stored Procedure for Student To Enters Answers of Exam .

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Std_Id	int	4
@Exam_Id	int	4
@Quest_Id	int	4
@quest_Ans	nvarchar(200)	400

SQL Script

```sql
create proc [dbo].[Exam_Answer_For_Student] @Std_Id int ,@Exam_Id int,@Quest_Id
int,@quest_Ans nvarchar(200)
as
begin
update [ExamSystem].[Stud_Exam_Quest]
set [Question_Answer]= @quest_Ans
where [ExamSystem].[Stud_Exam_Quest].[Std_ID]=@Std_Id and [ExamSystem].[Stud_Exam_-
Quest].[Exam_ID]=@Exam_Id and  [ExamSystem].[Stud_Exam_Quest].[Question_ID]=@Quest_Id
end
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure for Student To Enters
Answers of Exam .', 'SCHEMA', N'dbo', 'PROCEDURE', N'Exam_Answer_For_Student', NULL,
NULL
GO
```

Uses

[ExamSystem].[Stud_Exam_Quest]

📄 [dbo].[Exam_Answers]

MS_Description

Stored Procedure take Exam ID & Student ID That You Want To Show His Answers In Exam .

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Exam_ID	int	4
@Student_ID	int	4

SQL Script

```
CREATE proc [dbo].[Exam_Answers] @Exam_ID int, @Student_ID int
As
    select Q.Question_Text,Q.Model_Answer,STQ.Question_Answer
    from ExamSystem.Question Q,ExamSystem.Exam_Questions EQ,ExamSystem.Stud_Exam_Quest
STQ
    where EQ.Exam_ID = @Exam_ID
    And Q.Question_ID = EQ.Question_ID
    And STQ.Std_ID = @Student_ID
    And STQ.Question_ID = Q.Question_ID
    And STQ.Exam_ID = @Exam_ID
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Exam ID &
Student ID That You Want To Show His Answers In Exam .', 'SCHEMA', N'dbo', 'PROCEDURE',
N'Exam_Answers', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam_Questions]
[ExamSystem].[Question]
[ExamSystem].[Stud_Exam_Quest]

📄 [dbo].[Exam_Correction]

MS_Description

Stored Procedure take Exam ID & Student ID That You Want To Correct His Answers In Exam .

IT Compare Student Answers With Correct Answer In Question Table .

It Update Grade Of Question Either 0 or 1 in table Student_Exam_Question

And

Update Total Grade At Exam Table.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Exam_ID	int	4
@Studnet_ID	int	4

SQL Script

```
CREATE proc [dbo].[Exam_Correction] @Exam_ID int,@Studnet_ID int
AS
begin
    if(@Exam_ID is null or @Studnet_ID is null)
        select 'Exam or Student not Exist'
    else
        begin
            Declare c1 cursor
            for
                select top(10) STQ.Question_ID,STQ.Question_Answer,Q.Model_Answer
                from ExamSystem.Stud_Exam_Quest STQ,ExamSystem.Question Q
                where STQ.Exam_ID=@Exam_ID
                And STQ.Std_ID = @Studnet_ID
                And q.Question_ID=STQ.Question_ID

            for update
                Declare @Ques_ID int,@Stud_Ans nvarchar(max),@Ques_Ans nvarchar(max)
                open c1
                fetch c1 into @Ques_ID,@Stud_Ans,@Ques_Ans
                while @@FETCH_STATUS=0
```

```
                begin
                    if @Stud_Ans = @Ques_Ans
                        update ExamSystem.Stud_Exam_Quest
                        set Question_Grade = 1
                        where Exam_ID=@Exam_ID
                        And Std_ID=@Studnet_ID
                        And Question_ID=@Ques_ID
                    else
                        update ExamSystem.Stud_Exam_Quest
                        set Question_Grade = 0
                        where Exam_ID=@Exam_ID
                        And Std_ID=@Studnet_ID
                        And Question_ID=@Ques_ID
                    fetch c1 into @Ques_ID,@Stud_Ans,@Ques_Ans
                    end
            End
            Close c1
            Deallocate c1
        update ExamSystem.Exam
        set Total_Grade = (select sum(Question_Grade)
                            from ExamSystem.Stud_Exam_Quest
                            where Exam_ID = @Exam_ID
                            And Std_ID = @Studnet_ID)
                            where Exam.Exam_ID=@Exam_ID
                            And Exam.Std_ID = @Studnet_ID
    End
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure take Exam ID & Student ID That You Want To Correct His Answers In Exam
.
IT Compare Student Answers With Correct Answer In Question Table .
It Update Grade Of Question Either 0 or 1 in table Student_Exam_Question
And
Update Total Grade At Exam Table.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Exam_Correction',
NULL, NULL
GO
```

Uses

[ExamSystem].[Exam]
[ExamSystem].[Question]
[ExamSystem].[Stud_Exam_Quest]

Project > (local) > User databases > OnlineExamDB > Programmability > Stored Procedures > dbo.Exam_With_Questions_Report

Project SQL & BI.

ITI Alex.

Intake 40.

🖹 [dbo].[Exam_With_Questions_Report]

MS_Description

Stored Procedure take Exam ID To Show Exam Questions & Its Choices AS A Report .

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@exam_id	int	4

SQL Script

```sql
create proc [dbo].[Exam_With_Questions_Report] @exam_id integer
as
begin
select q.Question_Text , q.Choise1 ,q.Choise2,q.Choise3,q.Choise4
from ExamSystem.Question q ,ExamSystem.Exam_Questions eq
where q.Question_ID=eq.Question_ID and eq.Exam_ID=@exam_id
end
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Exam ID To Show
Exam Questions & Its Choices AS A Report .', 'SCHEMA', N'dbo', 'PROCEDURE', N'Exam_-
With_Questions_Report', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam_Questions]
[ExamSystem].[Question]

📑 [dbo].[generate_Exam_check]

MS_Description

Stored Procedure take Exam ID , Course_id , Student_id , Date of Exam , Duration , Number of T/F Questions & Number of MCQ Questions.
it Generate Exam With Random Question Related To Course That Has been Chosen.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Ex_ID	int	4
@course_id	int	4
@Student_id	int	4
@Exam_DateTime	datetime	8
@Exam_Duration	int	4
@MCQ	int	4
@TF	int	4

SQL Script

```sql
CREATE procedure [dbo].[generate_Exam_check] @Ex_ID INTEGER,@course_id
INTEGER,@Student_id INTEGER,@Exam_DateTime  Datetime,@Exam_Duration INTEGER,@MCQ
INTEGER,@TF INTEGER
as
if(@course_id is null or @student_id is null)
select 'please enter course_number and sudent_number'
else
INSERT  INTO ExamSystem.Exam
                    ( Exam_ID,
                      Std_ID   ,
                                Course_ID,
                                Exam_Duration,
                                Exam_DateTime ,
                                Total_Grade )

        VALUES      (       @Ex_ID,
```

```sql
                                            @Student_id   ,
                                                @course_id,
                                                @Exam_Duration,
                                                @Exam_DateTime ,
                                                null)

Declare @course_name nvarchar(50)
SET
@course_name=( select Course_Name from ExamSystem.Course c where c.Course_-
ID=@course_id)

 --Declare @last_Exam_id INTEGER
--SET
--@last_Exam_id=(select max(Exam_ID) from ExamSystem.Exam)+1
 --Declare @last_Exam_id INTEGER
 --SET
 --@last_Exam_id=0;

--declare @n  INTEGER
--set @n=0;

--declare @m   INTEGER
--set @m=0;
---------------------------------------------------------------------------
begin


DECLARE @TF_question_cursor CURSOR

SET
@TF_question_cursor = CURSOR FOR    --course _id

SELECT Top(@TF) Question_ID ,
Question_Text ,
Choise1 ,
Choise2 ,
Choise3 ,
Choise4,
Question_Type
 FROM ExamSystem.Question Q
 where Question_Type='T/F'
 AND Course_ID=@course_id
 ORDER BY RAND()                              --select random rows from questions



 Declare @Question_Text nvarchar(MAX),@Question_Type nvarchar(50),@Choise1
nvarchar(MAX),@Choise2 nvarchar(MAX) ,@Choise3 nvarchar(MAX),
@Choise4 nvarchar(MAX),@Question_ID INTEGER

open @TF_question_cursor
```

```sql
fetch next from @TF_question_cursor into @Question_ID ,
@Question_Text,
@Choise1 ,
@Choise2 ,
@Choise3 ,
@Choise4 ,
@Question_Type

print 'Exam: '+cast(@course_id as nvarchar)+' '+@course_name      --اسم الكورس والرقم
لول فادر هيبلتيلا

WHILE @@FETCH_STATUS = 0
BEGIN
PRINT cast(@Question_ID as nvarchar) + '-' + @Question_Text + CHAR(13)  + 'a-
'+@Choise1+CHAR(9)+'b-'+@Choise2


INSERT INTO ExamSystem.Exam_Questions  (Exam_ID,Question_ID)

            VALUES       (@Ex_ID,@Question_ID )

Insert into ExamSystem.Stud_Exam_Quest (Std_ID,Exam_ID,Question_ID)
values(@Student_id,@Ex_ID,@Question_ID)

fetch next from @TF_question_cursor into @Question_ID ,
@Question_Text,
@Choise1 ,
@Choise2 ,
@Choise3 ,
@Choise4 ,
@Question_Type




--set @last_Exam_id =@last_Exam_id+1;




end
print 'TF end'


end

CLOSE @TF_question_cursor
DEALLOCATE @TF_question_cursor

-------------------------------------------------------------------------------
-
begin
DECLARE @MCQ_question_cursor CURSOR
```

```
SET
@MCQ_question_cursor = CURSOR FOR    --course _id


SELECT Top(@MCQ) Question_ID ,
Question_Text ,
Choise1 ,
Choise2 ,
Choise3 ,
Choise4,
Question_Type
 FROM ExamSystem.Question Q
 where Question_Type='MCQ'
 AND Course_ID=@course_id
 ORDER BY RAND()                            --select random rows from questions



open @MCQ_question_cursor
fetch next from @MCQ_question_cursor into @Question_ID ,
@Question_Text,
@Choise1 ,
@Choise2 ,
@Choise3 ,
@Choise4 ,
@Question_Type

print 'Exam: '+cast(@course_id as nvarchar)+' '+@course_name    --اسم الكورس والرقم
اول فاول هيدر التيبل

WHILE @@FETCH_STATUS = 0
BEGIN

PRINT cast(@Question_ID as nvarchar) + '-' + @Question_Text + CHAR(13)  + 'a-'+@Choise1
+CHAR(9)+'b-'+@Choise2 +CHAR(9)+'c-'+@Choise3+CHAR(9)+'d-'+@Choise4




INSERT INTO ExamSystem.Exam_Questions  (Exam_ID,Question_ID)

          VALUES      (@Ex_ID,@Question_ID )

Insert into ExamSystem.Stud_Exam_Quest (Std_ID,Exam_ID,Question_ID)
values(@Student_id,@Ex_ID,@Question_ID)

fetch next from @MCQ_question_cursor into @Question_ID ,
@Question_Text,
@Choise1 ,
@Choise2 ,
@Choise3 ,
@Choise4 ,
```

```
@Question_Type



--set @last_Exam_id =@last_Exam_id+1;




end
print 'MCQ end'
print 'Exam page end'



end

CLOSE @MCQ_question_cursor
DEALLOCATE @MCQ_question_cursor
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure take Exam ID , Course_id , Student_id , Date of Exam , Duration ,
Number of T/F Questions &
Number of MCQ Questions.
it Generate Exam With Random Question Related To Course That Has been Chosen.',
'SCHEMA', N'dbo', 'PROCEDURE', N'generate_Exam_check', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]
[ExamSystem].[Exam]
[ExamSystem].[Exam_Questions]
[ExamSystem].[Question]
[ExamSystem].[Stud_Exam_Quest]

📄 [dbo].[Insert_Course]

MS_Description

Stored Procedure take Name of Course & Instructor ID That You Want To Add.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@name	nvarchar(50)	100
@instructor_id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Insert_Course]  @name nvarchar(50),@instructor_id int=1
AS
BEGIN
    insert into ExamSystem.Course values(@name,@instructor_id)
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Name of Course &
Instructor ID That You Want To Add.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Insert_Course',
NULL, NULL
GO
```

Uses

[ExamSystem].[Course]

📄 [dbo].[Insert_Exam]

MS_Description

Stored Procedure take Course_id , Student_id , Date of Exam , Duration & Total Grade.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Std_ID	int	4
@Course_ID	int	4
@Exam_Duration	int	4
@Exam_DateTime	datetime	8
@Total_Grade	int	4

SQL Script

```
create procedure [dbo].[Insert_Exam]           @Std_ID          INTEGER,
                                    @Course_ID       INTEGER,
                                    @Exam_Duration    INTEGER,
                                    @Exam_DateTime     datetime,
                                    @Total_Grade     INTEGER


AS

     BEGIN
       INSERT INTO ExamSystem.Exam
                 (             Std_ID   ,
                               Course_ID,
                               Exam_Duration,
                               Exam_DateTime ,
                               Total_Grade )

         VALUES    (           @Std_ID   ,
                               @Course_ID,
                               @Exam_Duration,
                               @Exam_DateTime ,
                               @Total_Grade )
```

```
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Course_id ,
Student_id , Date of Exam , Duration & Total Grade.', 'SCHEMA', N'dbo', 'PROCEDURE',
N'Insert_Exam', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam]

📄 [dbo].[Insert_Instructor]

MS_Description

Stored Procedure take Name Of Instructor You Want To Add.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@name	nvarchar(50)	100

SQL Script

```sql
CREATE PROCEDURE [dbo].[Insert_Instructor]  @name nvarchar(50)
AS
BEGIN
    insert into ExamSystem.Instructor values(@name)
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure take Name Of
Instructor You Want To Add.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Insert_Instructor',
NULL, NULL
GO
```

Uses

[ExamSystem].[Instructor]

🖹 [dbo].[Insert_Question]

MS_Description

Stored Procedure To Add New Question it Take Question_Text , Model_Answer(correct) ,First_choice , Second_-Choice ,
Third_Choice , Fourth_Choice , Question_Type(T/F-MCQ), Course_ID)

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Question_Text	nvarchar(max)	max
@Model_Answer	nvarchar(max)	max
@Choise1	nvarchar(max)	max
@Choise2	nvarchar(max)	max
@Choise3	nvarchar(max)	max
@Choise4	nvarchar(max)	max
@Question_Type	nvarchar(50)	100
@Course_ID	int	4

SQL Script

```
create procedure [dbo].[Insert_Question]
                            @Question_Text      nvarchar(Max),
                            @Model_Answer       nvarchar(Max),
                            @Choise1            nvarchar(Max),
                            @Choise2            nvarchar(Max),
                            @Choise3            nvarchar(Max),
                            @Choise4            nvarchar(Max),
                            @Question_Type      nvarchar(50),
                            @Course_ID      INTEGER

AS

        BEGIN
            INSERT INTO ExamSystem.Question
```

```sql
                        (
                                Question_Text ,
                                Model_Answer ,
                                Choise1       ,
                                Choise2       ,
                                Choise3       ,
                                Choise4       ,
                                Question_Type,
                                Course_ID)

        VALUES      (           @Question_Text ,
                                @Model_Answer ,
                                @Choise1       ,
                                @Choise2       ,
                                @Choise3       ,
                                @Choise4       ,
                                @Question_Type,
                                @Course_ID)
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure To Add New Question it Take Question_Text , Model_Answer(correct)
,First_choice , Second_-
Choice ,
Third_Choice , Fourth_Choice , Question_Type(T/F-MCQ), Course_ID)', 'SCHEMA', N'dbo',
'PROCEDURE', N'Insert_Question', NULL, NULL
GO
```

Uses

[ExamSystem].[Question]

📄 [dbo].[Insert_Stud_Exam_Quest]

MS_Description

Stored Procedure To store Student , Each Exam & Questions In It .

It Takes Student ID , Exam ID & Questions ID , Answer Of This Question , & Question Grade .

First Time We Could Add First 3 Parameters & Fill It In Table by Wizard.

After That Correction Procedure Will Take Theses Answers & Compare Them With Correct &change The Grade.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Std_ID	int	4
@Exam_ID	int	4
@Question_ID	int	4
@Question_Answer	nvarchar(max)	max
@Question_grade	int	4

SQL Script

```
CREATE procedure [dbo].[Insert_Stud_Exam_Quest]
                              @Std_ID      INTEGER,
                              @Exam_ID     INTEGER,
                              @Question_ID    INTEGER,
                              @Question_Answer nvarchar(max),
                              @Question_grade  INTEGER


AS

      BEGIN
          INSERT INTO ExamSystem.Stud_Exam_Quest
                      (      Std_ID,
                        Exam_ID,
                        Question_ID,
                              Question_Answer ,
                              Question_grade   )
```

```sql
            VALUES      (          @Std_ID,
                                   @Exam_ID,
                                   @Question_ID,
                                   @Question_Answer ,
                                      @Question_grade  )
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure To store Student , Each Exam & Questions In It .
It Takes Student ID , Exam ID & Questions ID , Answer Of This Question , & Question
Grade .
First Time We Could Add First 3 Parameters & Fill It In Table by Wizard.
After That Correction Procedure Will Take Theses Answers & Compare Them With Correct
&change The Grade.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Insert_Stud_Exam_Quest', NULL,
NULL
GO
```

Uses

[ExamSystem].[Stud_Exam_Quest]

🖺 [dbo].[Insert_Student]

MS_Description

Stored Procedure To Add New Student It Take Student Data like..(First Name , Last Name , Address , Age , Track_id)

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@fname	nvarchar(50)	100
@lname	nvarchar(50)	100
@address	nvarchar(max)	max
@age	int	4
@track_id	int	4

SQL Script

```sql
CREATE PROCEDURE [dbo].[Insert_Student]  @fname nvarchar(50),@lname nvarchar(50),
                         @address nvarchar(max)='unknown'  ,@age int=22 ,
@track_id int=1
AS
BEGIN
    insert into ExamSystem.Student values(@fname,@lname,@address,@age,@track_id)
    --update num of student in this track
    update ExamSystem.Track
    set No_Of_Stud+=1
    where ExamSystem.Track.Track_ID=@track_id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To Add New Student It
Take Student Data like..(First Name , Last Name , Address , Age , Track_id)', 'SCHEMA',
N'dbo', 'PROCEDURE', N'Insert_Student', NULL, NULL
GO
```

Uses

[ExamSystem].[Student]

[ExamSystem].[Track]

🖹 [dbo].[Insert_Student_Courses]

MS_Description

Stored Procedure To Store Student With Courses He is Studied.
It Takes Parameter..(Student_ID , Course_ID)

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Student_id	int	4
@course_id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Insert_Student_Courses]  @Student_id int ,@course_id int
AS
BEGIN
    insert into ExamSystem.Student_Courses(Student_ID,Course_ID)
values(@Student_id,@course_id)
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure To Store Student With Courses He is Studied.
It Takes Parameter..(Student_ID , Course_ID)', 'SCHEMA', N'dbo', 'PROCEDURE', N'Insert_-
Student_Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Student_Courses]

🖹 [dbo].[Insert_Topic]

MS_Description

Stored Procedure To Add New Topic It Take Topic Data like..(Name ,Course_id)

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Topic_Name	nvarchar(50)	100
@Course_ID	int	4

SQL Script

```
CREATE procedure [dbo].[Insert_Topic]

                              @Topic_Name     nvarchar(50),
                              @Course_ID     INTEGER

AS

     BEGIN
         INSERT INTO ExamSystem.Topic
                    (

                              Topic_Name      ,
                              Course_ID  )

         VALUES      (

                              @Topic_Name     ,
                              @Course_ID  )

     END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To Add New Topic It
Take Topic Data like..(Name ,Course_id)', 'SCHEMA', N'dbo', 'PROCEDURE', N'Insert_-
Topic', NULL, NULL
GO
```

Uses

[ExamSystem].[Topic]

[dbo].[Insert_Track]

MS_Description

Stored Procedure To Add New Track It Takes Track Data like..(Name , Duration , Manager Name , & Number Of Student In It)

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@name	nvarchar(50)	100
@manager	nvarchar(50)	100
@duration	int	4
@num	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Insert_Track]  @name nvarchar(50),@manager
nvarchar(50)='ANON',@duration int=3,@num int=0
AS
BEGIN
    insert into ExamSystem.Track values(@name,@manager,@duration,@num)
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure To Add New Track It Takes Track Data like..(Name , Duration , Manager
Name , & Number Of
Student In It )', 'SCHEMA', N'dbo', 'PROCEDURE', N'Insert_Track', NULL, NULL
GO
```

Uses

[ExamSystem].[Track]

🖾 [dbo].[Insert_Track_Courses]

MS_Description

Stored Procedure To Store Tracks with its Courses , Takes Track ID & Course ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@track_id	int	4
@course_id	int	4

SQL Script

```
---------------------------------
--insert track_course
CREATE PROCEDURE [dbo].[Insert_Track_Courses]  @track_id int=1 ,@course_id int=5
AS
BEGIN
    insert into ExamSystem.Track_Courses(Track_ID,Course_ID)
values(@track_id,@course_id)
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To Store Tracks with
its Courses , Takes Track ID & Course ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Insert_-
Track_Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Track_Courses]

🖺 [dbo].[Select_Course]

MS_Description

Stored Procedure To select Course by ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Select_Course]  @id int
AS
BEGIN
    select * from ExamSystem.Course where Course_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Course by ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Course', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]

🖹 [dbo].[Select_Exam]

MS_Description

Stored Procedure To select Exam by ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE procedure [dbo].[Select_Exam]  @id int
AS
SELECT * FROM ExamSystem.Exam  WHERE Exam_ID=@id
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Exam by
ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Exam', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam]

🖹 [dbo].[Select_Instructor]

MS_Description

Stored Procedure To select Instructor by ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Select_Instructor]  @id int
AS
BEGIN
    select * from ExamSystem.Instructor where Instructor_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Instructor
by ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Instructor', NULL, NULL
GO
```

Uses

[ExamSystem].[Instructor]

🖹 [dbo].[Select_Question]

MS_Description

Stored Procedure To select Question by ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE procedure [dbo].[Select_Question] @id int
AS
SELECT * FROM ExamSystem.Question  WHERE Question_ID=@id
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Question by
ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Question', NULL, NULL
GO
```

Uses

[ExamSystem].[Question]

📄 [dbo].[Select_Stud_Exam_Quest]

MS_Description

Stored Procedure To select Specific Student With Specific Exam & Questions in This Exam.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@stu_id	int	4
@exam_id	int	4

SQL Script

```
CREATE procedure [dbo].[Select_Stud_Exam_Quest] @stu_id INT ,@exam_id INT
AS
SELECT * FROM ExamSystem.Stud_Exam_Quest WHERE Std_ID=@stu_id AND Exam_ID=@exam_id
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Specific
Student With Specific Exam & Questions in This Exam.', 'SCHEMA', N'dbo', 'PROCEDURE',
N'Select_Stud_Exam_Quest', NULL, NULL
GO
```

Uses

[ExamSystem].[Stud_Exam_Quest]

🖺 [dbo].[Select_Student]

MS_Description

Stored Procedure To select Student by ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE procedure [dbo].[Select_Student] @id int
AS
SELECT * FROM ExamSystem.Student WHERE Std_ID=@id
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Student by
ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Student', NULL, NULL
GO
```

Uses

[ExamSystem].[Student]

📄 [dbo].[Select_Student_Courses]

MS_Description

Stored Procedure To select All Courses to Specific Student by his ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Select_Student_Courses]  @id int
AS
BEGIN
    select * from ExamSystem.Student_Courses where Student_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select All Courses
to Specific Student by his ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Student_-
Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Student_Courses]

📄 [dbo].[Select_Topic]

MS_Description

Stored Procedure To select Topic by ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE procedure [dbo].[Select_Topic] @id INT
AS
SELECT * FROM ExamSystem.Topic WHERE Topic_ID=@id
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Topic by
ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Topic', NULL, NULL
GO
```

Uses

[ExamSystem].[Topic]

📄 [dbo].[Select_Track]

MS_Description

Stored Procedure To select Track by ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
CREATE PROCEDURE [dbo].[Select_Track]  @id int
AS
BEGIN
    select * from ExamSystem.Track where Track_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Track by
ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Track', NULL, NULL
GO
```

Uses

[ExamSystem].[Track]

🗒 [dbo].[Select_Track_Courses]

MS_Description

Stored Procedure To select All Courses in Specific Tack by Track_ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4

SQL Script

```
--------------------------------------------------
--create proc to select track_course
CREATE PROCEDURE [dbo].[Select_Track_Courses]  @id int
AS
BEGIN
    select * from ExamSystem.Track_Courses where Track_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select All Courses
in Specific Tack by Track_ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Select_Track_Courses',
NULL, NULL
GO
```

Uses

[ExamSystem].[Track_Courses]

📄 [dbo].[Student_Grades_Report]

MS_Description

Stored Procedure that Show Each Student With his Last Grade .

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Student_ID	int	4

SQL Script

```
CREATE proc [dbo].[Student_Grades_Report] @Student_ID integer
as
begin
select * from
(select  c.Course_Name,e.Total_Grade,row_number() over (partition by e.Course_ID order
by e.Exam_DateTime  desc )  as r
from ExamSystem.exam e , ExamSystem.Course c
where e.Course_ID = c.Course_ID and e.Std_ID=@student_ID
 ) as newtable
where r=1
end


GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure that Show Each
Student With his Last Grade .', 'SCHEMA', N'dbo', 'PROCEDURE', N'Student_Grades_-
Report', NULL, NULL
GO
```

Uses

[ExamSystem].[Course]
[ExamSystem].[Exam]

Project > (local) > User databases > OnlineExamDB > Programmability > Stored Procedures >
dbo.Student_Info_Report

Project SQL
& BI.
ITI Alex.
   Intake 40.

🖹 [dbo].[Student_Info_Report]

MS_Description

Stored Procedure To select Students in Specific Track Takes Track_ID used in Report.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@track_No	int	4

SQL Script

```sql
create proc [dbo].[Student_Info_Report] @track_No integer
as
begin
  select * from ExamSystem.Student
  where Track_ID=@track_No
end
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure To select Students in
Specific Track Takes Track_ID used in Report.', 'SCHEMA', N'dbo', 'PROCEDURE',
N'Student_Info_Report', NULL, NULL
GO
```

Uses

[ExamSystem].[Student]

🗒 [dbo].[Update_Course]

MS_Description

Stored Procedure to Update Course And Instructor who Teach This Course.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4
@name	nvarchar(50)	100
@instructor_id	int	4

SQL Script

```sql
CREATE PROCEDURE [dbo].[Update_Course]  @id int,@name nvarchar(50),@instructor_id int
=1
AS
BEGIN
    update ExamSystem.Course
    set Course_Name=@name ,Instructor_ID=@instructor_id
    where Course_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Course And
Instructor who Teach This Course.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Update_Course',
NULL, NULL
GO
```

Uses

[ExamSystem].[Course]

📄 [dbo].[Update_Exam]

MS_Description

Stored Procedure to Update Exam Its Takes All Parameter like(Exam_ID , Std_ID , Course_ID , Exam_Duration , Date of Exam , Total_Grade)

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Exam_ID	int	4
@Std_ID	int	4
@Course_ID	int	4
@Exam_Duration	int	4
@Exam_DateTime	datetime	8
@Total_Grade	int	4

SQL Script

```
create procedure [dbo].[Update_Exam]    @Exam_ID        INTEGER,
                            @Std_ID        INTEGER,
                            @Course_ID      INTEGER,
                            @Exam_Duration    INTEGER,
                            @Exam_DateTime    datetime,
                             @Total_Grade    INTEGER


As
       BEGIN
          UPDATE ExamSystem.Exam
          SET    Std_ID = @Std_ID,
                 Course_ID=@Course_ID,
                 Exam_Duration = @Exam_Duration ,
                 Exam_DateTime=@Exam_DateTime,
                 Total_Grade = @Total_Grade
          WHERE  Exam_ID = @Exam_ID
       END
```

```
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure to Update Exam Its Takes All Parameter like(Exam_ID , Std_ID , Course_-
ID , Exam_Duration , Date of Exam , Total_Grade)', 'SCHEMA', N'dbo', 'PROCEDURE',
N'Update_Exam', NULL, NULL
GO
```

Uses

[ExamSystem].[Exam]

📄 [dbo].[Update_Instructor]

MS_Description

Stored Procedure to Update Instructor Data Its Takes his ID &Name.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4
@name	nvarchar(50)	100

SQL Script

```
CREATE PROCEDURE [dbo].[Update_Instructor]  @id int,@name nvarchar(50)
AS
BEGIN
    update ExamSystem.Instructor
    set Instructor_Name=@name
    where Instructor_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Instructor
Data Its Takes his ID &Name.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Update_Instructor',
NULL, NULL
GO
```

Uses

[ExamSystem].[Instructor]

📄 [dbo].[Update_Question]

MS_Description

Stored Procedure to Update Question , its Model Answer , choices , Question Type & Course of this Question.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Question_ID	int	4
@Question_Text	nvarchar(max)	max
@Model_Answer	nvarchar(max)	max
@Choise1	nvarchar(max)	max
@Choise2	nvarchar(max)	max
@Choise3	nvarchar(max)	max
@Choise4	nvarchar(max)	max
@Question_Type	nvarchar(50)	100
@Course_ID	int	4

SQL Script

```sql
create procedure [dbo].[Update_Question]    @Question_ID        INTEGER,
                                @Question_Text     nvarchar(Max),
                                @Model_Answer      nvarchar(Max),
                                @Choise1           nvarchar(Max),
                                @Choise2           nvarchar(Max),
                                @Choise3           nvarchar(Max),
                                @Choise4           nvarchar(Max),
                                @Question_Type     nvarchar(50),
                                @Course_ID    INTEGER


As
      BEGIN
          UPDATE ExamSystem.Question
          SET Question_Text =@Question_Text,
              Model_Answer=@Model_Answer,
```

```
                Choise1=@Choise1,
                Choise2=@Choise2,
                Choise3=@Choise3,
                Choise4=@Choise4,
                Question_Type=@Question_Type,
                Course_ID=@Course_ID
            WHERE  Question_ID = @Question_ID
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Question ,
its Model Answer , choices , Question Type & Course of this Question.', 'SCHEMA',
N'dbo', 'PROCEDURE', N'Update_Question', NULL, NULL
GO
```

Uses

[ExamSystem].[Question]

🖹 [dbo].[Update_Stud_Exam_Quest]

MS_Description

Stored Procedure to Update Answer & Grade in Table 'Stud_Exam_Quest ' by Student_ID ,Exam_ID & Question_ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Std_ID	int	4
@Exam_ID	int	4
@Question_ID	int	4
@Question_Answer	nvarchar(max)	max
@Question_grade	int	4

SQL Script

```sql
create procedure [dbo].[Update_Stud_Exam_Quest]  @Std_ID        INTEGER,
                                      @Exam_ID     INTEGER,
                                      @Question_ID   INTEGER,
                                      @Question_Answer nvarchar(max),
                                      @Question_grade  INTEGER


As
        BEGIN
            UPDATE   ExamSystem.Stud_Exam_Quest
            SET
                                    Question_Answer=@Question_Answer ,
                                    Question_grade=@Question_grade
            WHERE  Std_ID = @Std_ID
            AND    Exam_ID=@Exam_ID
            AND    Question_ID=@Question_ID
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Answer &
Grade in Table ''Stud_Exam_Quest '' by Student_ID ,Exam_ID & Question_ID.', 'SCHEMA',
N'dbo', 'PROCEDURE', N'Update_Stud_Exam_Quest', NULL, NULL
```

```
GO
```

Uses

[ExamSystem].[Stud_Exam_Quest]

▣ [dbo].[Update_Student]

MS_Description

Stored Procedure to Update Student Data Depending on Student_ID .
It Also Update Number of Student in Table Track Depending On
Track of This Student.(increase new track he join & decrease old track he left)

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4
@fname	nvarchar(50)	100
@lname	nvarchar(50)	100
@address	nvarchar(max)	max
@age	int	4
@track_id	int	4

SQL Script

```
--update student
CREATE PROCEDURE [dbo].[Update_Student]  @id int , @fname nvarchar(50)='ANON',@lname
nvarchar(50)='ANON',

                                @address nvarchar(max)='unknown'  ,@age int=22 ,
@track_id int=1

AS

BEGIN
    --select current track_id of this student who will update his data
    declare @current_track_id int
    set @current_track_id=(
                    select s.Track_ID
                    from ExamSystem.Student s
                    where s.Std_ID=@id
                    )
```

```
END
BEGIN
    update ExamSystem.Student
    set Std_FName= @fname , Std_LName=@lname , Std_Address=@address , Std_Age=@age ,
Track_ID=@track_id
    where Std_ID=@id
END

    --//if track_id != current track_is of student
 if(@track_id!=@current_track_id )
  BEGIN
        update ExamSystem.Track set No_Of_Stud+=1 where Track.Track_ID=@track_id
        update ExamSystem.Track set No_Of_Stud-=1 where Track.Track_-
ID=@current_track_id

  END
GO
EXEC sp_addextendedproperty N'MS_Description', N'
Stored Procedure to Update Student Data Depending on Student_ID .
It Also Update Number of Student in Table Track Depending On
Track of This Student.(increase new track he join & decrease old track he left)',
'SCHEMA', N'dbo', 'PROCEDURE', N'Update_Student', NULL, NULL
GO
```

Uses

[ExamSystem].[Student]
[ExamSystem].[Track]

📄 [dbo].[Update_Student_Courses]

MS_Description

Stored Procedure to Update Course That Student Study by Taking Student_ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4
@course_id	int	4

SQL Script

```sql
CREATE PROCEDURE [dbo].[Update_Student_Courses]  @id int,@course_id int
AS
BEGIN
    update ExamSystem.Student_Courses
    set Course_ID=@course_id
    where Student_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Course That
Student Study by Taking Student_ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Update_Student_-
Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Student_Courses]

🖺 [dbo].[Update_Topic]

MS_Description

Stored Procedure to Update Topic Data by Taking ID , Name & Course_ID.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@Topic_ID	int	4
@Topic_Name	nvarchar(50)	100
@Course_ID	int	4

SQL Script

```
create procedure [dbo].[Update_Topic]          @Topic_ID        INTEGER,
                                      @Topic_Name     nvarchar(50),
                                      @Course_ID     INTEGER


As
        BEGIN
            UPDATE ExamSystem.Topic
            set
                Topic_Name= @Topic_Name   ,
                Course_ID=  @Course_ID
            WHERE   Topic_ID = @Topic_ID
        END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Topic Data
by Taking ID , Name & Course_ID.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Update_Topic',
NULL, NULL
GO
```

Uses

[ExamSystem].[Topic]

[dbo].[Update_Track]

MS_Description

Stored Procedure to Update Track Name It Takes Track_ID & New Name of Track To change .

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4
@name	nvarchar(50)	100

SQL Script

```
CREATE PROCEDURE [dbo].[Update_Track]  @id int,@name nvarchar(50)--,@manager
nvarchar(50)='Anon'
AS
BEGIN
    update ExamSystem.Track
    set Track_Name=@name
    where Track_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Track Name
It Takes Track_ID & New Name of Track To change .', 'SCHEMA', N'dbo', 'PROCEDURE',
N'Update_Track', NULL, NULL
GO
```

Uses

[ExamSystem].[Track]

📄 [dbo].[Update_Track_Courses]

MS_Description

Stored Procedure to Update Courses Data That Belong To Specific Track.

Properties

Property	Value
ANSI Nulls On	True

Parameters

Name	Data Type	Max Length (Bytes)
@id	int	4
@course_id	int	4

SQL Script

```
-----------------------------------
--update track_course
CREATE PROCEDURE [dbo].[Update_Track_Courses]  @id int,@course_id int
AS
BEGIN
    update ExamSystem.Track_Courses
    set Course_ID=@course_id
    where Track_ID=@id
END
GO
EXEC sp_addextendedproperty N'MS_Description', N'Stored Procedure to Update Courses
Data That Belong To Specific Track.', 'SCHEMA', N'dbo', 'PROCEDURE', N'Update_Track_-
Courses', NULL, NULL
GO
```

Uses

[ExamSystem].[Track_Courses]

Project > (local) > User databases > OnlineExamDB > Security > Schemas   Project SQL & BI.

ITI Alex.

Intake 40.

⋀ *Schemas*

Objects

Name
ExamSystem

⋀ ExamSystem

Properties

Property	Value
Owner	dbo

SQL Script

```
CREATE SCHEMA [ExamSystem]
AUTHORIZATION [dbo]
GO
```

Used By

[ExamSystem].[Course]
[ExamSystem].[Exam]
[ExamSystem].[Exam_Questions]
[ExamSystem].[Instructor]
[ExamSystem].[Question]
[ExamSystem].[Stud_Exam_Quest]
[ExamSystem].[Student]
[ExamSystem].[Student_Courses]
[ExamSystem].[Topic]
[ExamSystem].[Track]
[ExamSystem].[Track_Courses]