Database documentation

Server name: 192.168.1.8

Database name: ExamDB

Documentation date: 3/1/2022

Database size: 16.00 MB

Database description:

Table summary

	<u>Name</u>	<u>Data size</u>	<u>Index size</u>	Row size (*)	Row Count
Ħ	dbo.answers	8 KB	8 KB	185	2
Ħ	dbo.course	8 KB	8 KB	123	16
Ħ	dbo.course_dept	8 KB	8 KB	19	9
Ħ	dbo.department	8 KB	8 KB	115	11
Ħ	dbo.exam	8 KB	8 KB	103	1
Ħ	dbo.instructor	8 KB	8 KB	95	1
Ħ	dbo.instructor_student	0 KB	0 KB	43	0
Ħ	dbo.knex_migrations	8 KB	8 KB	537	1
Ħ	dbo.knex_migrations_lock	8 KB	8 KB	15	1
Ħ	dbo.mainUser	8 KB	8 KB	833	1
Ħ	dbo.questions	8 KB	8 KB	259	1
Ħ	dbo.student	0 KB	0 KB	87	0
	dbo.student_exam_question	8 KB	8 KB	139	2
Ħ	dbo.topic	8 KB	8 KB	115	6

14 table(s)

* Estimated row size. It assumes that all variable-length are 100% full, except those set to MAX.



dbo.answers

<u>l F</u>	<u> P</u>	Column name	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pac .	P	id	int	NO		
		answer_one	nvarchar (20)	NO		
		answer_two	nvarchar (20)	NO		
		answer_three	nvarchar (20)	NO		
		answer_four	nvarchar (20)	NO		
P	<	question_id	int	NO		

🐍 Indexes

	<u>Index name</u>		<u>Column name</u>	Sort direction	<u>Is unique</u>	<u>Index type</u>
PK	answers 3213E83F3495FDB8	id		ASC	Yes	CLUSTERED

🗣 Foreign keys

Constraint name Column name		<u>Reference</u>	<u>Description</u>
answers_question_id_foreign	question_id	PK_question_3213E83FEC6D7F3	

Objects that depend on dbo.answers

dbo.deleteAnswerByld (Stored procedure) dbo.insertAnswer (Stored procedure) dbo.selectAllAnswers (Stored procedure) dbo.updateAnswerByld (Stored procedure)

dbo.course

<u>IFP</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox 📍	id	int	NO		
	name	nvarchar (50)	NO		
	total_hours	int	NO		
₽ĸ	topic_id	int	NO		

🖟 Indexes

	<u>Index name</u>		<u>Column name</u>	Sort direction	<u>ls unique</u>	<u>Index type</u>
PK	course 3213E83F576DD4F3	id		ASC	Yes	CLUSTERED

👫 Foreign keys

Constraint name	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
course_topic_id_foreign	topic_id	PK_topic_3213E83F6B289CAD (dbo topic)	

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint
dbo.course_dept	course_dept_crs_id_foreign	PKcourse3213E83F576DD4F3
dbo.instructor	instructor_course_id_foreign	PKcourse3213E83F576DD4F3

Objects that depend on dbo.course

dbo.getAllInstructorsProc (Stored procedure)
dbo.insertCourse (Stored procedure)
dbo.selectAllCourses (Stored procedure)
dbo.selectAllCoursesDept (Stored procedure)

dbo.course_dept

<u>IFP</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox 💡	id	int	NO		
₽ĸ	crs_id	int	NO		
₽ĸ	dept_id	int	NO		

🐍 Indexes

<u>Index name</u>		Column name	Sort direction	<u>ls unique</u>	<u>Index type</u>
PK course d 3213E83F7E08238B	id		ASC	Yes	CLUSTERED

♀k Foreign keys

Constraint name	Column name	<u>Reference</u>	<u>Description</u>
course_dept_crs_id_foreign	crs_id	PKcourse3213E83F576DD4F3 (dbo.course)	
course_dept_dept_id_foreign	dept_id	PKdepartme3213E83F185FD5 CF (dbo.department)	

Objects that depend on dbo.course_dept

dbo.insertCourseDept (Stored procedure) dbo.selectAllCoursesDept (Stored procedure)

dbo.department

<u>IF</u>	P	<u>Column name</u>	Data type	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox.	P	id	int	NO		
		name	nvarchar (50)	NO		

Indexes

	<u>Index name</u>		<u>Column name</u>	Sort direction	<u>Is unique</u>	<u>Index type</u>
PK	departme 3213E83E185ED5CE	id		ASC	Yes	CLUSTERED

保 Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint
dbo.course_dept	course_dept_dept_id_foreign	PKdepartme3213E83F185FD5CF
dbo.instructor	instructor_dept_id_foreign	PKdepartme3213E83F185FD5CF
dbo.student	student_dept_id_foreign	PKdepartme3213E83F185FD5CF

Objects that depend on dbo.department

dbo.getAllInstructorsProc (Stored procedure) dbo.selectAllCoursesDept (Stored procedure)

dbo.exam

<u>IFP</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox 📍	id	int	NO		
	exam_name	nvarchar (30)	NO		
	duration	bigint	NO		
_	total_marks	int	NO		
₽ĸ	instructor_id	uniqueidentifier	NO		

ዬ Indexes

	<u>Index name</u>		Column name	Sort direction	<u>ls unique</u>	<u>Index type</u>
PKexam_	_3213E83F990DE12D	id		ASC	Yes	CLUSTERED

🗣 Foreign keys

Constraint name	<u>Column name</u>	<u>Reference</u>	<u>Description</u>
exam instructor id foreign	instructor id	instructor pkey (dbo instructor)	

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint		
dbo.student_exam_question	student_exam_question_exam_id_foreign	PKexam3213E83F990DE12D		

Objects that depend on dbo.exam

dbo.insertExam (Stored procedure)
dbo.insertStudentExamQuestion (Stored procedure)

dbo.selectAllExams (Stored procedure)



dbo.instructor

<u>IFP</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pax 📍	id	uniqueidentifier	NO	(newid())	
	phone_number	nvarchar (20)	NO		
	salary	int	NO		
₽ĸ	course_id	int	NO		
₽ĸ	dept_id	int	NO		
₽ĸ	user_id	uniqueidentifier	NO		

🐍 Indexes

<u>Index name</u>		Column name	Sort direction	<u>ls unique</u>	<u>Index type</u>
instructor pkey	id		ASC	Yes	CLUSTERED

👫 Foreign keys

Constraint name	Column name	<u>Reference</u>	<u>Description</u>
instructor_course_id_foreign	course_id	PKcourse3213E83F576DD4F3 (dbo.course)	
instructor_dept_id_foreign	dept_id	PKdepartme3213E83F185FD5 CF (dbo.department)	
instructor_user_id_foreign	user_id	mainUser_pkey (dbo.mainUser)	

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint		
dbo.exam	exam_instructor_id_foreign	instructor_pkey		
dbo.instructor_student	instructor_student_instructor_id_foreign	instructor_pkey		

Objects that depend on dbo.instructor

dbo.getAllInstructorsProc (Stored procedure)
dbo.insertExam (Stored procedure)
dbo.insertInstructorUser (Stored procedure)

dbo.instructor_student

<u>IFP</u>	<u>Column name</u>	Data type	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pax 💡	id	int	NO		
₽ĸ	student_id	uniqueidentifier	NO		
₽ĸ	instructor_id	uniqueidentifier	NO		

🐍 Indexes

	<u>Index name</u>		<u>Column name</u>	Sort direction	<u>ls unique</u>	<u>Index type</u>
Ī	PK instruct 3213E83FDFE138F4	id		ASC	Yes	CLUSTERED

🗣 Foreign keys

Constraint name	Column name	<u>Reference</u>	<u>Description</u>
instructor_student_instructor_id_ foreign	instructor_id	instructor_pkey (dbo.instructor)	
instructor_student_student_id_fo	student_id	student_pkey (dbo.student)	

dbo.knex_migrations

<u> 11</u>	F P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox.	P	id	int	NO		
		name	nvarchar (255)	YES		
		batch	int	YES		
		migration_time	datetime2	YES		

% Indexes

	<u>Index name</u>		<u>Column name</u>	Sort direction	<u>ls unique</u>	<u>Index type</u>
PK	knex mig 3213F83FCB6AA60D	id		ASC	Yes	CLUSTERED

dbo.knex_migrations_lock

<u>IFP</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pax 💡	index	int	NO		
	is_locked	int	YES		
Q. Ind	dovoe				

🐍 Indexes

<u>Index name</u>	<u>Column name</u>	Sort direction	<u>ls unique</u>	<u>Index type</u>
PK_knex_mig1D0A33482D6ACFCA	index	ASC	Yes	CLUSTERED

dbo.mainUser

<u>IFP</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox 📍	id	uniqueidentifier	NO	(newid())	
	first_name	nvarchar (50)	NO		
	last_name	nvarchar (50)	NO		
	email	nvarchar (150)	NO		
	password	nvarchar (150)	NO		

Indexes

<u>Index name</u>		<u>Column name</u>	Sort direction	<u>Is unique</u>	Index type
mainUser pkev	id		ASC	Yes	CLUSTERED

윢 Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint
dbo.instructor	instructor_user_id_foreign	mainUser_pkey
dbo.student	student_user_id_foreign	mainUser_pkey

Objects that depend on dbo.mainUser

dbo.getAllInstructorsProc (Stored procedure) dbo.insertInstructorUser (Stored procedure)

dbo.questions

<u>IFP</u>	<u>Column name</u>	Data type	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox 📍	id	int	NO		
	name	nvarchar (50)	NO		
	question_type_id	nvarchar (50)	NO		
	correct_answer	nvarchar (20)	NO		

🐍 Indexes

	<u>Index name</u>		<u>Column name</u>	Sort direction	<u>ls unique</u>	<u>Index type</u>
PK	guestion 3213F83FFC6D7F3B	id		ASC	Yes	CLUSTERED

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint
dbo.answers	answers_question_id_foreign	PKquestion3213E83FEC6D7F3B
dbo.student_exam_question	student_exam_question_question_id_foreig	PKquestion3213E83FEC6D7F3B

Check constraints

	<u>Check name</u>		<u>1e</u>	<u>Column name</u>	<u>Expression</u>		
CK	questions	quest	1D9B5BB	question type id	([question type id]='TRUEorFALSE' OR [question type id]		

Objects that depend on dbo.questions

dbo.deleteQuestionById (Stored procedure)

dbo.insertAnswer (Stored procedure)

dbo.insertQuestion (Stored procedure)

dbo.insertStudentExamQuestion (Stored procedure)

dbo.selectAllAnswers (Stored procedure)

dbo.selectAllQuestions (Stored procedure)

dbo.updateAnswerByld (Stored procedure)

dbo.updateQuestionById (Stored procedure)

dbo.student

<u>IFP</u>	Column name	Data type	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox 📍	id	uniqueidentifier	NO	(newid())	
	phone_number	nvarchar (20)	NO		
₽ĸ	user_id	uniqueidentifier	NO		
₽ĸ	dept_id	int	NO		

🖟 Indexes

<u>Index name</u>		Column name	Sort direction	<u>ls unique</u>	Index type
student pkev	id		ASC	Yes	CLUSTERED

🧚 Foreign keys

Constraint name	Column name	<u>Reference</u>	<u>Description</u>
student_dept_id_foreign	dept_id	PKdepartme3213E83F185FD5 CF (dbo.department)	
student_user_id_foreign	user_id	mainUser_pkey (dbo.mainUser)	

Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint
dbo.instructor_student	instructor_student_student_id_foreign	student_pkey
dbo.student exam question	student exam question student id foreign	student pkey

Objects that depend on dbo.student

dbo.insertStudentExamQuestion (Stored procedure)

dbo.student_exam_question

<u>IFP</u>	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
🏗 📍	id	int	NO		
₽ĸ	exam_id	int	YES		
₽ĸ	student_id	uniqueidentifier	YES		
₽ĸ	question_id	int	YES		
	answer	nvarchar (50)	YES		

🐍 Indexes

	<u>lı</u>	<u>ndex name</u>		<u>Column name</u>	Sort direction	<u>Is unique</u>	Index type
PK	student	3213F83FADF99D08	id		ASC	Yes	CLUSTERED

🗣 Foreign keys

Constraint name	Column name	<u>Reference</u>	<u>Description</u>
student_exam_question_exam_i d_foreign	exam_id	PKexam3213E83F990DE12D (dbo.exam)	
student_exam_question_questio n_id_foreign	question_id	PKquestion3213E83FEC6D7F3 B (dbo.questions)	
student_exam_question_student _id_foreign	student_id	student_pkey (dbo.student)	

Objects that depend on dbo.student_exam_question

dbo.insertStudentExamQuestion (Stored procedure) dbo.selectAllStudentExamQuestions (Stored procedure)

dbo.topic

<u> 1 </u>	F P	<u>Column name</u>	<u>Data type</u>	<u>Nulls</u>	<u>Default</u>	<u>Description</u>
Pox.	P	id	int	NO		
		name	nvarchar (50)	NO		

🐍 Indexes

		<u>Index name</u>		Column name	Sort direction	<u>ls unique</u>	<u>Index type</u>
PK	topic	3213E83F6B289CAD	id		ASC	Yes	CLUSTERED

保 Referencing tables

<u>Table name</u>	<u>Foreign key</u>	Primary key or unique constraint
dho course	course tonic id foreign	PK tonic 3213E83E6B289CAD

Objects that depend on dbo.topic

dbo.deleteTopicByID (Stored procedure)
dbo.insertTopic (Stored procedure)
dbo.selectAllCourses (Stored procedure)
dbo.selectAllTopics (Stored procedure)
dbo.updateTopicByID (Stored procedure)



☑ dbo.deleteAnswerByld

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	<u>DAT</u>	A TYPE MODE	<u>DESCRIPTION</u>	
@id	int	IN		
CREATE PROC deleteAnswerByI	d			
	@id AS int			
AS				
BEGIN delete Answers where id = @id				
END;				

Related objects

dbo.answers(TABLE)

☑ dbo.deleteExamByld

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TY	PE MODE	<u>DESCRIPTION</u>
@id	int	IN	
CREATE PROC deleteExamById			
	@id AS int		
AS BEGIN delete exam where id = @id			
END;			

☑ dbo.deleteQuestionByld

Created: 18 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@id	int	IN	

Related objects

dbo.questions(TABLE)

dbo.deleteStudentExamQuestionByld

Created: 19 Feb 2022 Last updated: 19 Feb 2022

a Parameters

PARAMETER NAME	<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
@id	int	IN	
CREATE PROC deleteStudentExamQuest	ionById		
(
@id	AS int		
)			
AS			
BEGIN			
delete student_exam_question			
where id = @id			
END;			

☑ dbo.deleteTopicByID

Created: 19 Feb 2022 Last updated: 19 Feb 2022

a Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	<u>DESCRIPTION</u>	
@topic_id	int	IN		
CREATE PROC deleteTopicByID				
((@top	ic id AS in t			
)	_			
AS BEGIN				
<pre>delete topic where id = @topic id</pre>				
END;				

Related objects

dbo.topic(TABLE)

☑ dbo.getAllInstructorsProc

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC getAllInstructorsProc
AS
BEGIN
SELECT mainUser.id as userId,
                         mainUser.first_name as userFirstName,
                         mainUser.last_name as userLastName,
mainUser.email as userEmail,
                         instructor.id as ins_id,
                         instructor.phone_number as insPhone,
                         instructor.salary as insSalary,
                         (select name as courseName from course where id = instructor.course_id) as courses,
(select name as departmentName from department where id = instructor.dept_id) as
department
FROM mainUser
INNER JOIN instructor
ON
                         mainUser.id = instructor.user id
END
```

Related objects

dbo.mainUser(TABLE) dbo.course(TABLE) dbo.instructor(TABLE) dbo.department(TABLE)

☑ dbo.getAllStudentsProc

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC getAllStudentsProc

AS

BEGIN

SELECT mainUser.id as userId,

mainUser.first_name as userFirstName,

mainUser.last_name as userLastName,

mainUser.email as userEmail,

student.id as student_id,

student.phone_number as student_phone,

(select name as departmentName from department where id = student.dept_id) as

department

FROM mainUser

INNER JOIN student

ON

mainUser.id = student.user_id

END
```

dbo.insertAnswer

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	MODE	<u>DESCRIPTION</u>
@answer_one	varchar	IN	
@answer_two	varchar	IN	
@answer_three	varchar	IN	
@answer_four	varchar	IN	
@question_id	int	IN	

Related objects

dbo.answers(TABLE)
dbo.questions(TABLE)

☑ dbo.insertCourse

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	DESCRIPTION
@crs_name	varchar	IN	
@crs_total_hours	int	IN	
@crs_topic_id	int	IN	

Related objects

dbo.course(TABLE)

☑ dbo.insertCourseDept

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	MODE	<u>DESCRIPTION</u>
@crs_id	int	IN	
@dept_id	int	IN	

Related objects

dbo.course_dept(TABLE)

dbo.insertExam

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	<u>DESCRIPTION</u>
@exam_name	varchar	IN	
@duration	int	IN	
@total_marks	int	IN	
@instructor_id	varchar	IN	

Related objects

dbo.exam(TABLE)
dbo.instructor(TABLE)

dbo.insertInstructorUser

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	<u>DESCRIPTION</u>
@first_name	varchar	IN	
@last_name	varchar	IN	
@email	varchar	IN	
@password	varchar	IN	
@phone_number	varchar	IN	
@salary	int	IN	
@crs_id	int	IN	
@dept_id	int	IN	

```
CREATE Proc insertInstructorUser
                    @first name as varchar(max),
                    @last_name as varchar(max),
                     @email as varchar(max),
                    @password as varchar(max),
                    @phone_number as varchar(max),
                    @salary as int,
                    @crs id as int,
                    @dept id as int
AS
BEGIN
BEGIN TRANSACTION fill_user
BEGIN TRY
-- USER CODE --
INSERT INTO mainUser
VALUES
                            NEWID(),
                             @first_name,
@last_name,
                             @email,
                             @password
-- END USER CODE --
-- INSTRUCTOR CODE --
declare @recentUserInserted uniqueidentifier
SELECT TOP 1 @recentUserInserted = id FROM mainUser where email = @email
INSERT INTO instructor
VALUES
                             NEWID(),
                             @phone_number,
                             @salary,
                             @crs_id,
                             @dept_id,
@recentUserInserted
                         );
--END INSTRUCTOR CODE--
COMMIT TRANSACTION fill user
END TRY
BEGIN CATCH
ROLLBACK TRANSACTION fill user
END CATCH
END
```

Related objects

dbo.instructor(TABLE)

dbo.mainUser(TABLE)

dbo.insertQuestion

Created: 18 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	MODE	<u>DESCRIPTION</u>
@name	varchar	IN	
@question_type_id	varchar	IN	
@correct_answer	varchar	IN	

Related objects

dbo.questions(TABLE)

dbo.insertStudentExamQuestion

Created: 19 Feb 2022 Last updated: 19 Feb 2022

a Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	<u>DESCRIPTION</u>
@exam_id	int	IN	
@student_id	varchar	IN	
@question_id	int	IN	
@answer	varchar	IN	

Related objects

dbo.student_exam_question(TABLE)

dbo.questions(TABLE)

dbo.exam(TABLE)

dbo.student(TABLE)



dbo.insertStudentUser

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	MODE	<u>DESCRIPTION</u>
@first_name	varchar	IN	
@last_name	varchar	IN	
@email	varchar	IN	
@password	varchar	IN	
@phone_number	varchar	IN	
@ins_id	varchar	IN	

```
CREATE Proc insertStudentUser
                    @first name as varchar(max),
                    @last_name as varchar(max),
@email as varchar(max),
                    @password as varchar(max),
                    @phone number as varchar(max),
                    @ins id as varchar(max)
AS
BEGIN
BEGIN TRANSACTION user filling
BEGIN TRY
-- USER CODE --
INSERT INTO mainUser
VALUES
                             NEWID(),
                             @first_name,
                             @last name,
                             0email,
                             @password
                         );
-- END USER CODE --
-- INSTRUCTOR CODE -
declare @recentUserInserted uniqueidentifier;
SELECT TOP 1 @recentUserInserted = id FROM mainUser where email = @email;
DECLARE @instructorConvertedUUID uniqueidentifier;
SET @instructorConvertedUUID = CONVERT(uniqueidentifier, @ins id);
declare @department_id as int
SELECT TOP 1 @department id = dept id FROM instructor where id = @instructorConvertedUUID;
INSERT INTO student
VALUES
                             NEWID(),
                             @phone_number,
                             @recentUserInserted,
                             @department id
--filling the instructor table
declare @recentStudentInserted uniqueidentifier;
SELECT TOP 1 @recentStudentInserted = id FROM student where phone_number = @phone_number;
INSERT INTO instructor student
VALUES
                             @recentStudentInserted,
                             @instructorConvertedUUID
--END INSTRUCTOR CODE--
```

COMMIT TRANSACTION user_filling
END TRY
BEGIN CATCH
ROLLBACK TRANSACTION user_filling
END CATCH
END

☑ dbo.insertTopic

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	MODE	<u>DESCRIPTION</u>
@topic_name	varchar	IN	
CREATE PROC insertTopic			
(
@to	opic_name AS varchar(max)	
) AS			
BEGIN			
insert into topic values (@topic	name);		

Related objects

dbo.topic(TABLE)

END;

dbo.selectAllAnswers

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllAnswers
AS
BEGIN
SELECT answers.* , name FROM answers, questions
where questions.id = answers.question_id
END;
```

Related objects

dbo.questions(TABLE)
dbo.answers(TABLE)

☑ dbo.selectAllCourses

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllCourses
AS
BEGIN
SELECT course.name as courseName, course.total_hours as totalHours, course.id as courseId,
topic.name as topicName FROM course, topic where course.topic_id = topic.id
END;
```

Related objects

dbo.topic(TABLE)
dbo.course(TABLE)

☑ dbo.selectAllCoursesDept

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllCoursesDept
AS
BEGIN
SELECT course_dept.id ,course.name as courseName, department.name as departmentName FROM course, department,course_dept
where course_id = course_dept.crs_id and course_dept.dept_id =department.id
END;
```

Related objects

dbo.course(TABLE) dbo.department(TABLE) dbo.course_dept(TABLE)

dbo.selectAllDepartments

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllDepartments
AS
BEGIN
SELECT * FROM department
END;
```

☑ dbo.selectAllExams

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllExams
AS
BEGIN
SELECT * FROM exam
END;
```

Related objects

dbo.exam(TABLE)

☑ dbo.selectAllQuestions

Created: 18 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllQuestions
AS
BEGIN
SELECT * FROM questions
END;
```

Related objects

dbo.questions(TABLE)

dbo.selectAllStudentExamQuestions

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllStudentExamQuestions
AS
BEGIN
SELECT * FROM student_exam_question
END;
```

Related objects

dbo.student_exam_question(TABLE)

☑ dbo.selectAllTopics

Created: 19 Feb 2022 Last updated: 19 Feb 2022

```
CREATE PROC selectAllTopics
AS
BEGIN
SELECT * FROM topic
END;
```

Related objects

dbo.topic(TABLE)

☑ dbo.updateAnswerByld

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

<u>DATA TYPE</u>	<u>MODE</u>	<u>DESCRIPTION</u>
int	IN	
varchar	IN	
int	IN	
	int varchar varchar varchar varchar	int IN varchar IN varchar IN varchar IN varchar IN

Related objects

dbo.answers(TABLE)
dbo.questions(TABLE)

esc.com to

register)

☑ dbo.updateExamByld

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	<u>DESCRIPTION</u>
@id	int	IN	
@exam_name	varchar	IN	
@duration	int	IN	
@total_marks	int	IN	
@instructor_id	varchar	IN	

dbdesc.com to register)

☑ dbo.updateQuestionByld

Created: 18 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	DESCRIPTION
@id	int	IN	
@name	varchar	IN	
@question_type_id	varchar	IN	
@correct_answer	varchar	IN	

Related objects

dbo.questions(TABLE)

dbdesc.com to register)

dbo.updateStudentExamQuestionByld

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	<u>DESCRIPTION</u>
@id	int	IN	
@exam_id	int	IN	
@student_id	varchar	IN	
@question_id	int	IN	
@answer	varchar	IN	

```
CREATE PROC updateStudentExamQuestionById
                                  @id AS int,
                                  @exam_id AS int,
                                  @student_id AS varchar(max),
                                  @question id AS int,
                                  @answer AS varchar(max)
AS
BEGIN
update student_exam_question
set exam_id = @exam_id, student_id = @student_id,
                                 question id = @question id, answer = @answer
where id = @id And Exists(select id from exam
where id = @exam_id)
And Exists(select id from student
where id = @student_id)
And Exists(select id from questions
where id = @question_id)
END;
```

dbdesc.com to register)

☑ dbo.updateTopicByID

Created: 19 Feb 2022 Last updated: 19 Feb 2022

Parameters

PARAMETER NAME	DATA TYPE	<u>MODE</u>	<u>DESCRIPTION</u>
@topic_id	int	IN	
@topic_name	varchar	IN	

Related objects

dbo.topic(TABLE)





sys.sys
Created: 13 Apr 2009 Last updated: 22 Aug 2017



COMPONENT NAME	TYPE	DEBIVATION	NAMESDACE
COMPONENT NAME anySimpleType	ANY_SIMPLE_TYPE	DERIVATION RESTRICTION	NAMESPACE http://www.w3.org/2001/XMLSchema
anyType	ANY TYPE	NONE	http://www.w3.org/2001/XMLSchema
base	ATTRIBUTE	NONE	http://www.w3.org/XML/1998/namespace
base64Encoded	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
clrTypeName	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
lang	ATTRIBUTE	NONE	http://www.w3.org/XML/1998/namespace
localeId	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
maxLength	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
precision	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
scale	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
space	ATTRIBUTE	NONE	http://www.w3.org/XML/1998/namespace
sqlCollationVersion	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
sqlCompareOptions	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
sqlDbType	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
sqlSortId	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
sqlTypeAlias	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
supplementaryCharacters	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
xmlSchemaCollection	ATTRIBUTE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
xml	COMPLEX_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
ENTITIES	LIST_TYPE	RESTRICTION	http://www.w3.org/20 <mark>01</mark> /XMLSchema
IDREFS	LIST_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
NMTOKENS	LIST_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
sqlCompareOptionsList	LIST_TYPE	NONE	http://schemas.microsoft.com/sqlserver/2004/sqltypes
NOTATION	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
QName	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
anyURI	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
base64Binary	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
boolean	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
date	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
dateTime	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
decimal	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
double	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
duration	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
float	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
gDay	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
gMonth	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
gMonthDay	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
gYear	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
gYearMonth	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
hexBinary	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
string	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
time	PRIMITIVE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
ENTITY	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema

ExamDB database documentation - XML Schemas

ID	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
IDREF	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
NCName	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
NMTOKEN	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
Name	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
bigint	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
binary	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
bit	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
byte	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
char	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
date	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
datetime	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
datetime2	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
datetimeoffset	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
dbobject	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
decimal	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
float	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
hierarchyid	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
image	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
int	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
int	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
integer	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
language	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
long	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
money	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
nchar	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
negativeInteger	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
nonNegativeInteger	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
nonPositiveInteger	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
normalizedString	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
ntext	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
numeric	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
nvarchar	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
positiveInteger	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
real	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
short	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
smalldatetime	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
smallint	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
smallmoney	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
sqlCompareOptionsEnum	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
sqlDbTypeEnum	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
text	SIMPLE TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
time	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
timestamp	SIMPLE TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
timestampNumeric	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
tinyint	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
token	SIMPLE TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
uniqueidentifier	SIMPLE_TYPE	RESTRICTION	http://schemas.microsoft.com/sqlserver/2004/sqltypes
unsignedByte	SIMPLE TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
unsignedInt	SIMPLE TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
unsignedLong	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
unsignedShort	SIMPLE_TYPE	RESTRICTION	http://www.w3.org/2001/XMLSchema
anagnoron	5 EE_111 E	RECTRICTION	-

ExamDB database documentation - XML Schemas

 varbinary
 SIMPLE_TYPE
 RESTRICTION
 http://schemas.microsoft.com/sqlserver/2004/sqltypes

 varchar
 SIMPLE_TYPE
 RESTRICTION
 http://schemas.microsoft.com/sqlserver/2004/sqltypes

 xmlSpaceEnum
 SIMPLE_TYPE
 RESTRICTION
 http://www.w3.org/XML/1998/namespace

Microsoft.SqlServer.Types

Created: 22 Aug 2017 Last updated: 22 Aug 2017

CLR NAME: microsoft.sqlserver.types, version=14.0.0.0, culture=neutral, publickeytoken=89845dcd8080cc91,

processorarchitecture=msil

PERMISSION: UNSAFE_ACCESS

DESCRIPTION:

User name	Login name	Created	Roles	User description
dbo	root	4/8/2003	db_owner	

Role name	App role	Users with this role	Role description
db_accessadmin			
db_backupoperator			
db_datareader			
db_datawriter			
db_ddladmin			
db_denydatareader			
db_denydatawriter			
db_owner		dbo	
db_securityadmin			
public			