localhost, 13001 Documentation

QueensClassSchedule

Server localhost,13001

Author evnul

Created Friday, May 17, 2024 9:06:08 PM

File C:\Users\evnul\OneDrive\Documents\My Database Documentation\localhost,13001_documentation-2024-05-17T21-

Path 06-08.pdf

Table of Contents

able of Contents	2
localhost,13001	5
User databases	7
QueensClassSchedule Database	8
Tables	10
[CourseManagement].[Class]	11
[CourseManagement].[Course]	13
[DbSecurity].[UserAuthorization]	15
[DepartmentManagement].[Department]	17
[InstructionManagement].[ModeOfInstruction]	19
[InstructorManagement].[Instructor]	21
[LocationManagement].[BuildingLocation]	23
[LocationManagement].[RoomLocation]	25
[Process].[WorkflowSteps]	27
[Uploadfile].[CurrentSemesterCourseOfferings]	29
区 Views	31
[Utils].[uvw_FindColumnDefinitionPlusDefaultAndCheckConstraint]	32
Stored Procedures	35
[dbo].[LoadDataFromUpload]	36
[dbo].[sp_AddForeignKey]	39
[dbo].[sp_DropForeignKey]	41
[dbo].[sp_TruncateQueensClassTables]	43
[Process].[usp_ShowWorkflowSteps]	45
[Process].[usp_TrackWorkFlow]	
Table-valued Functions	
[dbo].[DatabaseObjects]	
[dbo].[JSONHierarchy]	
User-Defined Data Types	55
[Udt].[BuildingName]	56
[Udt].[ClassTime]	57
[Udt].[CourseCode]	58
[Udt].[CourseName]	59
[Udt].[DateAdded]	60
[Udt].[DateOfLastUpdate]	61
[Udt].[DepartmentCode]	62
[Udt].[DepartmentName]	
[Udt].[FirstName]	64

101	
[Udt].[FullName]	65
Ü፱ [Udt].[GroupName]	
Ü⊒ [Udt].[IndividualProject]	
ÜŪ [Udt].[LastName]	
Ü⊒ [Udt].[NodeName]	
[Udt].[OldSurrogateKeyInt]	
[Udt].[RoomNumber]	
[Udt].[Schedule]	
[Udt].[SurrogateKeyInt]	
Users	
BiStudent	
▲ dbo	76
EC3\thehitman	77
guest	
student	79
Database Roles	80
db_accessadmin	80
db_backupoperator	80
db_datareader	81
db_datawriter	81
db_ddladmin	82
db_denydatareader	82
db_denydatawriter	82
db_owner	82
db_securityadmin	83
public	83
Schemas	84
⚠ CourseManagement	85
⚠ DbSecurity	86
⚠ DepartmentManagement	87
⚠ GroupNameProject3	88
⚠ InstructionManagement	89
⚠ InstructorManagement	90
	91
↑ Process	
△ Udt	
⚠ Uploadfile	94
⚠ Utils	

Λ	urLastName96
---	--------------

Iocalhost,13001

Databases (1)

• ■ QueensClassSchedule

Server Properties

Property	Value		
Product	Microsoft SQL Server		
Version	15.0.4345.5		
Language	English		
Platform	NT x64		
Edition	Developer Edition (64-bit)		
Engine Edition	3 (Enterprise)		
Processors	12		
OS Version	6.2 (9200)		
Physical Memory	6326		
Is Clustered	False		
Root Directory	C:\		
Collation	SQL_Latin1_General_CP1_CI_AS		

Server Settings

Property	Value		
Default data file path	/var/opt/mssql/data/		
Default backup file path	/var/opt/mssql/backup/		
Default log file path	/var/opt/mssql/data/		
Recovery Interval (minutes)	0		
Default index fill factor	0		
Default backup media retention	0		
Compress Backup	False		

Advanced Server Settings

Property	Value
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	
Scan for startup procs	False	
Transform noise words	False	
CLR enabled	False	
Blocked process threshold	0	
Filestream access level	False	
Optimize for ad hoc workloads	False	
CLR strict security	True	

□ User databases		

Databases (1)

• ■ QueensClassSchedule

☐ QueensClassSchedule Database

Database Properties

Property	Value		
SQL Server Version	SqlServer2019		
Compatibility Level	SQL Server 2016		
Last backup time	05/18/2024		
Last log backup time	05/17/2024		
Creation date	May 15 2024		
Users	7		
Database Encryption Enabled	False		
Database Encryption Algorithm	None		
Database size	208.00 MB		
Unallocated space	65.54 MB		

Database Options

Property	Value
Compatibility Level	130
Database collation	SQL_Latin1_General_CP1_CI_AS
Restrict access	MULTI_USER
Is read-only	False
Auto close	False
Auto shrink	False
Database status	ONLINE
In standby	False
Cleanly shutdown	False
Supplemental logging enabled	False
Snapshot isolation state	OFF
Read committed snapshot on	False
Recovery model	FULL
Page verify option	CHECKSUM
Auto create statistics	True
Auto update statistics	True
Auto update statistics asynchronously	False
ANSI NULL default	False
ANSI NULL enabled	False
ANSI padding enabled	False

ANSI warnings enabled	False
Arithmetic abort enabled	False
Concatenating NULL yields NULL	False
Numeric roundabort enabled	False
Quoted Identifier On	False
Recursive triggers enabled	False
Close cursors on commit	False
Local cursors by default	False
Fulltext enabled	True
Trustworthy	False
Database chaining	False
Forced parameterization	False
Master key encrypted by server	False
Published	False
Subscribed	False
Merge published	False
Is distribution database	False
Sync with backup	False
Service broker GUID	9f5d9c27-bc5c-4d67-9b27-e7ae78bc354f
Service broker enabled	False
Date correlation	False
CDC enabled	False
Encrypted	False
Honor broker priority	False
Default language	English
Default fulltext language LCID	1033
Nested triggers enabled	True
Transform noise words	False
Two-digit year cutoff	2049
Containment	NONE
Target recovery time	60
Database owner	sa

Files

Name	Туре	Size	Maxsize	Autogrowth	File Name
QC2019	Data	72.00 MB	unlimited	64.00 MB	/var/opt/mssql/data/QueensClassScheduleCurrent- Semester.mdf
QC2019_log	Log	136.00 MB	2048.00 GB	64.00 MB	/var/opt/mssql/data/QueensClassScheduleCurrent- Semester_0.ldf

■ Tables

Objects

Name
CourseManagement.Class
CourseManagement.Course
DbSecurity.UserAuthorization
DepartmentManagement.Department
InstructionManagement.ModeOfInstruction
InstructorManagement.Instructor
LocationManagement.BuildingLocation
LocationManagement.RoomLocation
Process.WorkflowSteps
Uploadfile.CurrentSemesterCourseOfferings

[CourseManagement].[Class]

Properties

Property	Value	
Collation	SQL_Latin1_General_CP1_CI_AS	
Row Count (~)	9148	
Created	4:03:49 AM Wednesday, May 15, 2024	
Last Modified	12:07:34 AM Saturday, May 18, 2024	

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity	Default
PK C	ClassId	int	4	NOT NULL	1 - 1	
	Courseld	int	4	NULL allowed		
	Schedule	nvarchar(50)	100	NOT NULL		
	MaxEnrollment	int	4	NULL allowed		
	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL		
	DateAdded	[Udt].[DateAdded]	8	NOT NULL		(sysdatetime())
	DateOfLastUpdate	[Udt].[DateOfLastUpdate]	8	NOT NULL		(sysdatetime())
	Modeld	int	4	NULL allowed		
	Enrollment	int	4	NULL allowed		

Indexes

Key	Name	Key Columns	Unique
PKP C	PKClassCB1927C09FA9BA4F	ClassId	True

Check Constraints

Name	On Column	Constraint
CKClassMaxEnroll09746778	MaxEnrollment	([MaxEnrollment]>=(0))

SQL Script

```
CREATE TABLE [CourseManagement].[Class]
[ClassId] [int] NOT NULL IDENTITY(1, 1),
[CourseId] [int] NULL,
[Schedule] [nvarchar] (50) COLLATE SQL Latin1 General CP1 CI AS NOT NULL,
[MaxEnrollment] [int] NULL,
[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,
[DateAdded] [Udt].[DateAdded] NOT NULL CONSTRAINT [DF Class DateAdded 1C873BEC] DEFAULT
(sysdatetime()),
[DateOfLastUpdate] [Udt].[DateOfLastUpdate] NOT NULL CONSTRAINT [DF__Class__DateOfLas__1D7B6025]
DEFAULT (sysdatetime()),
[ModeId] [int] NULL,
[Enrollment] [int] NULL
) ON [PRIMARY]
ALTER TABLE [CourseManagement].[Class] ADD CONSTRAINT [CK Class MaxEnroll 09746778] CHECK
(([MaxEnrollment]>=(0)))
GO
ALTER TABLE [CourseManagement].[Class] ADD CONSTRAINT [PK Class CB1927C09FA9BA4F] PRIMARY KEY
CLUSTERED ([ClassId]) ON [PRIMARY]
GO
```

Uses

[Udt].[DateAdded] [Udt].[DateOfLastUpdate] [Udt].[SurrogateKeyInt] CourseManagement

Used By

■ [CourseManagement].[Course]

Properties

Property	Value
Collation SQL_Latin1_General_CP1_CI_AS	
Row Count (~)	4424
Created	4:03:49 AM Wednesday, May 15, 2024
Last Modified	11:10:20 PM Friday, May 17, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity	Default
PK C	Courseld	int	4	NOT NULL	1 - 1	
	CourseCode	nvarchar(20)	40	NOT NULL		
	CourseName	nvarchar(100)	200	NOT NULL		
	InstructorId	int	4	NULL allowed		
	Credits	int	4	NULL allowed		
	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL		
	DateAdded	[Udt].[DateAdded]	8	NOT NULL		(sysdatetime())
	DateOfLastUpdate	[Udt].[DateOfLastUpdate]	8	NOT NULL		(sysdatetime())

Indexes

Key	Name	Key Columns	Unique
PK C	PKCourseC92D71A730FE83E2	Courseld	True

Check Constraints

Name	On Column	Constraint
CK_Course_Credits_05A3D694	Credits	([Credits]>=(0))
CK_Course_Credits_66EA454A	Credits	([Credits]>=(0))

```
CREATE TABLE [CourseManagement].[Course]
```

```
[CourseId] [int] NOT NULL IDENTITY(1, 1),
[CourseCode] [nvarchar] (20) COLLATE SQL Latin1 General CP1 CI AS NOT NULL,
[CourseName] [nvarchar] (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
[InstructorId] [int] NULL,
[Credits] [int] NULL,
[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,
[DateAdded] [Udt].[DateAdded] NOT NULL CONSTRAINT [DF Course DateAdde 1A9EF37A] DEFAULT
(sysdatetime()),
[DateOfLastUpdate] [Udt].[DateOfLastUpdate] NOT NULL CONSTRAINT [DF Course DateOfLa 1B9317B3]
DEFAULT (sysdatetime())
) ON [PRIMARY]
ALTER TABLE [CourseManagement].[Course] ADD CONSTRAINT [CK Course Credits 05A3D694] CHECK
(([Credits]>=(0)))
ALTER TABLE [CourseManagement].[Course] ADD CONSTRAINT [CK__Course__Credits__66EA454A] CHECK
(([Credits]>=(0)))
ALTER TABLE [CourseManagement].[Course] ADD CONSTRAINT [PK Course C92D71A730FE83E2] PRIMARY KEY
CLUSTERED ([CourseId]) ON [PRIMARY]
```

Uses

[Udt].[DateAdded] [Udt].[DateOfLastUpdate] [Udt].[SurrogateKeyInt] CourseManagement

Used By

[DbSecurity].[UserAuthorization]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	3
Created	3:20:12 AM Thursday, May 9, 2024
Last Modified	3:20:12 AM Thursday, May 9, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Default
PK C	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL	
	ClassTime	[Udt].[ClassTime]	10	NULL allowed	('09:15')
	IndividualProject	[Udt].[IndividualProject]	120	NULL allowed	('PROJECT 3')
	GroupMemberLastName	[Udt].[LastName]	70	NOT NULL	
	GroupMemberFirstName	[Udt].[FirstName]	40	NOT NULL	
	GroupName	[Udt].[GroupName]	40	NOT NULL	('Group #')
	DateAdded	[Udt].[DateAdded]	8	NULL allowed	(sysdatetime())

Indexes

Key	Name	Key Columns	Unique
PK	PK_UserAuth_0FEC36CB932D9258	UserAuthorizationKey	True

```
CREATE TABLE [DbSecurity].[UserAuthorization]

(
[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,

[ClassTime] [Udt].[ClassTime] NULL CONSTRAINT [DF_UserAutho_Class_756D6ECB] DEFAULT ('09:15'),

[IndividualProject] [Udt].[IndividualProject] NULL CONSTRAINT [DF_UserAutho_Indiv_76619304]

DEFAULT ('PROJECT 3'),

[GroupMemberLastName] [Udt].[LastName] NOT NULL,

[GroupMemberFirstName] [Udt].[FirstName] NOT NULL,

[GroupName] [Udt].[GroupName] NOT NULL CONSTRAINT [DF_UserAutho_Group_7755B73D] DEFAULT

('Group #'),

[DateAdded] [Udt].[DateAdded] NULL CONSTRAINT [DF_UserAutho_DateA_7849DB76] DEFAULT

(sysdatetime())

) ON [PRIMARY]
```

```
GO
ALTER TABLE [DbSecurity].[UserAuthorization] ADD CONSTRAINT [PK_UserAuth_OFEC36CB932D9258]
PRIMARY KEY CLUSTERED ([UserAuthorizationKey]) ON [PRIMARY]
GO
```

Uses

[Udt].[ClassTime] [Udt].[DateAdded] [Udt].[FirstName] [Udt].[GroupName] [Udt].[IndividualProject] [Udt].[LastName] [Udt].[SurrogateKeyInt] DbSecurity

[DepartmentManagement].[Department]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	5
Created	3:59:05 AM Wednesday, May 15, 2024
Last Modified	4:56:24 AM Wednesday, May 15, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity	Default
PK C	DepartmentId	int	4	NOT NULL	1 - 1	
	DepartmentName	nvarchar(100)	200	NOT NULL		
.	DepartmentCode	nvarchar(20)	40	NOT NULL		
	MaxEnrollment	int	4	NULL allowed		
	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL		
	DateAdded	[Udt].[DateAdded]	8	NOT NULL		(sysdatetime())
	DateOfLastUpdate	[Udt].[DateOfLastUpdate]	8	NOT NULL		(sysdatetime())

Indexes

Key	Name	Key Columns	Unique
PK2 C	PKDepartmeB2079BEDF711B050	DepartmentId	True
	UQDepartme6EA8896DD00FC9C5	DepartmentCode	True

Check Constraints

Name	On Column	Constraint
CKDepartmenMaxEn7D0E9093	MaxEnrollment	([MaxEnrollment]>=(0))

```
CREATE TABLE [DepartmentManagement].[Department]
(
[DepartmentId] [int] NOT NULL IDENTITY(1, 1),
```

```
[DepartmentName] [nvarchar] (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
[DepartmentCode] [nvarchar] (20) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
[MaxEnrollment] [int] NULL,
[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,
[DateAdded] [Udt].[DateAdded] NOT NULL CONSTRAINT [DF_Departmen_DateA_16CE6296] DEFAULT
(sysdatetime()),
[DateOfLastUpdate] [Udt].[DateOfLastUpdate] NOT NULL CONSTRAINT [DF_Departmen_DateO_17C286CF]
DEFAULT (sysdatetime())
) ON [PRIMARY]

GO
ALTER TABLE [DepartmentManagement].[Department] ADD CONSTRAINT [CK_Departmen_MaxEn_7D0E9093]
CHECK (([MaxEnrollment]>=(0)))

GO
ALTER TABLE [DepartmentManagement].[Department] ADD CONSTRAINT [PK_Departme_B2079BEDF711B050]
PRIMARY KEY CLUSTERED ([DepartmentId]) ON [PRIMARY]

GO
ALTER TABLE [DepartmentManagement].[Department] ADD CONSTRAINT [UQ_Departme_6EA8896DD00FC9C5]
UNIQUE NONCLUSTERED ([DepartmentCode]) ON [PRIMARY]

GO
```

Uses

[Udt].[DateAdded]
[Udt].[DateOfLastUpdate]
[Udt].[SurrogateKeyInt]
DepartmentManagement

Used By

[InstructionManagement].[ModeOfInstruction]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	4
Created	4:03:49 AM Wednesday, May 15, 2024
Last Modified	4:56:24 AM Wednesday, May 15, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity	Default
PK C	Modeld	int	4	NOT NULL	1 - 1	
4	ModeName	nvarchar(50)	100	NOT NULL		
	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL		
	DateAdded	[Udt].[DateAdded]	8	NOT NULL		(sysdatetime())
	DateOfLastUpdate	[Udt].[DateOfLastUpdate]	8	NOT NULL		(sysdatetime())

Indexes

Key	Name	Key Columns	Unique
PK G	PKModeOfInD83F75C45AEED8A3	Modeld	True
	UQModeOfInE29BFAFDEA2CFEEE	ModeName	True

```
CREATE TABLE [InstructionManagement].[ModeOfInstruction]

(
[ModeId] [int] NOT NULL IDENTITY(1, 1),

[ModeName] [nvarchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,

[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,

[DateAdded] [Udt].[DateAdded] NOT NULL CONSTRAINT [DF_ModeOfIns_DateA_22401542] DEFAULT (sysdatetime()),

[DateOfLastUpdate] [Udt].[DateOfLastUpdate] NOT NULL CONSTRAINT [DF_ModeOfIns_DateO_2334397B] DEFAULT (sysdatetime())

ON [PRIMARY]

GO

ALTER TABLE [InstructionManagement].[ModeOfInstruction] ADD CONSTRAINT [PK_ModeOfIn_-D83F75C45AEED8A3] PRIMARY KEY CLUSTERED ([ModeId]) ON [PRIMARY]

GO
```

ALTER TABLE [InstructionManagement].[ModeOfInstruction] ADD CONSTRAINT [UQ__ModeOfIn__- E29BFAFDEA2CFEEE] UNIQUE NONCLUSTERED ([ModeName]) ON [PRIMARY]

GO

Uses

[Udt].[DateAdded] [Udt].[DateOfLastUpdate] [Udt].[SurrogateKeyInt] InstructionManagement

Used By

[InstructorManagement].[Instructor]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	1597
Created	4:03:49 AM Wednesday, May 15, 2024
Last Modified	11:09:05 PM Friday, May 17, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity	Default
PK C	InstructorId	int	4	NOT NULL	1 - 1	
	MaxCredits	int	4	NULL allowed		
	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL		
	DateAdded	[Udt].[DateAdded]	8	NOT NULL		(sysdatetime())
	DateOfLastUpdate	[Udt].[DateOfLastUpdate]	8	NOT NULL		(sysdatetime())
	FullName	varchar(255)	255	NULL allowed		(NULL)
	FirstName	varchar(255)	255	NULL allowed		(NULL)
	LastName	varchar(255)	255	NULL allowed		(NULL)

Indexes

Key	Name	Key Columns	Unique
PKP C	PKInstruct9D010A9B6452E85A	InstructorId	True

Check Constraints

Name	On Column	Constraint	
CK_Instructo_MaxCr_00DF2177	MaxCredits	([MaxCredits]>=(0))	

SQL Script

CREATE TABLE [InstructorManagement].[Instructor]

```
[InstructorId] [int] NOT NULL IDENTITY(1, 1),
[MaxCredits] [int] NULL,
[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,
[DateAdded] [Udt].[DateAdded] NOT NULL CONSTRAINT [DF__Instructo__DateA__18B6AB08] DEFAULT
(sysdatetime()),
[DateOfLastUpdate] [Udt].[DateOfLastUpdate] NOT NULL CONSTRAINT [DF Instructo DateO 19AACF41]
DEFAULT (sysdatetime()),
[FullName] [varchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL CONSTRAINT [DF__Instructo__-
Fulln 54CB950F] DEFAULT (NULL),
[FirstName] [varchar] (255) COLLATE SQL_Latin1_General_CP1_CI_AS NULL CONSTRAINT [DF__-
Instructo First 55BFB948] DEFAULT (NULL),
[LastName] [varchar] (255) COLLATE SQL Latin1 General CP1 CI AS NULL CONSTRAINT [DF Instructo -
LastN 56B3DD81] DEFAULT (NULL)
) ON [PRIMARY]
ALTER TABLE [InstructorManagement].[Instructor] ADD CONSTRAINT [CK Instructo MaxCr 00DF2177]
CHECK (([MaxCredits]>=(0)))
ALTER TABLE [InstructorManagement].[Instructor] ADD CONSTRAINT [PK__Instruct__9D010A9B6452E85A]
PRIMARY KEY CLUSTERED ([InstructorId]) ON [PRIMARY]
```

Uses

[Udt].[DateAdded] [Udt].[DateOfLastUpdate] [Udt].[SurrogateKeyInt] InstructorManagement

Used By

☐ [LocationManagement].[BuildingLocation]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	24
Created	4:03:49 AM Wednesday, May 15, 2024
Last Modified	11:09:05 PM Friday, May 17, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity	Default
PKP C	BuildingId	int	4	NOT NULL	1 - 1	
	BuildingName	nvarchar(100)	200	NOT NULL		
	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL		
	DateAdded	[Udt].[DateAdded]	8	NOT NULL		(sysdatetime())
	DateOfLastUpdate	[Udt].[DateOfLastUpdate]	8	NOT NULL		(sysdatetime())

Indexes

Key	Name		Unique
PKP C	PK_Building_5463CDC4628F2BCE	BuildingId	True

```
CREATE TABLE [LocationManagement].[BuildingLocation]

(
[BuildingId] [int] NOT NULL IDENTITY(1, 1),

[BuildingName] [nvarchar] (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,

[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,

[DateAdded] [Udt].[DateAdded] NOT NULL CONSTRAINT [DF_BuildingL_DateA_1E6F845E] DEFAULT (sysdatetime()),

[DateOfLastUpdate] [Udt].[DateOfLastUpdate] NOT NULL CONSTRAINT [DF_BuildingL_DateO_1F63A897] DEFAULT (sysdatetime())

) ON [PRIMARY]

GO

ALTER TABLE [LocationManagement].[BuildingLocation] ADD CONSTRAINT [PK_-
Building_5463CDC4628F2BCE] PRIMARY KEY CLUSTERED ([BuildingId]) ON [PRIMARY]

GO
```

Project > localhost,13001 > User databases > QueensClassSchedule > Tables > LocationManagement.BuildingLocation

Uses

[Udt].[DateAdded] [Udt].[DateOfLastUpdate] [Udt].[SurrogateKeyInt] LocationManagement

Used By

☐ [LocationManagement].[RoomLocation]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	319
Created	4:03:49 AM Wednesday, May 15, 2024
Last Modified	11:09:05 PM Friday, May 17, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Identity	Default
PK C	Roomld	int	4	NOT NULL	1 - 1	
	BuildingId	int	4	NULL allowed		
	RoomNumber	nvarchar(20)	40	NOT NULL		
	Capacity	int	4	NULL allowed		
	UserAuthorizationKey	[Udt].[SurrogateKeyInt]	4	NOT NULL		
	DateAdded	[Udt].[DateAdded]	8	NOT NULL		(sysdatetime())
	DateOfLastUpdate	[Udt].[DateOfLastUpdate]	8	NOT NULL		(sysdatetime())

Indexes

Key	Name	Key Columns	Unique
PKP C	PK_RoomLoca_32863939A8DA8CEE	Roomld	True

Check Constraints

Name	On Column	Constraint
CK_RoomLocat_Capac_0F2D40CE	Capacity	([Capacity]>=(0))

```
CREATE TABLE [LocationManagement].[RoomLocation]
[RoomId] [int] NOT NULL IDENTITY(1, 1),
[BuildingId] [int] NULL,
```

```
[RoomNumber] [nvarchar] (20) COLLATE SQL_Latin1_General_CP1_CT_AS NOT NULL,

[Capacity] [int] NULL,

[UserAuthorizationKey] [Udt].[SurrogateKeyInt] NOT NULL,

[DateAdded] [Udt].[DateAdded] NOT NULL CONSTRAINT [DF_RoomLocat_DateA_2057CCD0] DEFAULT (sysdatetime()),

[DateOfLastUpdate] [Udt].[DateOfLastUpdate] NOT NULL CONSTRAINT [DF_RoomLocat_DateO_214BF109] DEFAULT (sysdatetime())

ON [PRIMARY]

GO

ALTER TABLE [LocationManagement].[RoomLocation] ADD CONSTRAINT [CK_RoomLocat_Capac_0F2D40CE] CHECK (([Capacity]>=(0)))

GO

ALTER TABLE [LocationManagement].[RoomLocation] ADD CONSTRAINT [PK_RoomLocat_32863939A8DA8CEE] PRIMARY KEY CLUSTERED ([RoomId]) ON [PRIMARY]

GO
```

Uses

[Udt].[DateAdded] [Udt].[DateOfLastUpdate] [Udt].[SurrogateKeyInt] LocationManagement

[Process].[WorkflowSteps]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Row Count (~)	0
Created	5:08:58 AM Wednesday, May 15, 2024
Last Modified	5:08:58 AM Wednesday, May 15, 2024

Columns

Key	Name	Data Type	Max Length (Bytes)	Nullability	Default
PK C	WorkFlowStepKey	int	4	NOT NULL	
	WorkFlowStepDescription	nvarchar(100)	200	NOT NULL	
	WorkFlowStepTableRowCount	int	4	NULL allowed	((0))
	StartingDateTime	datetime2	8	NULL allowed	(sysdatetime())
	EndingDateTime	datetime2	8	NULL allowed	(sysdatetime())
	ClassTime	char(5)	5	NULL allowed	('09:15')
	UserAuthorizationKey	int	4	NOT NULL	

Indexes

Key	Name	Key Columns	Unique
PK	PKWorkflow27726CBF1B202E99	WorkFlowStepKey	True

```
CREATE TABLE [Process].[WorkflowSteps]

(
[WorkFlowStepKey] [int] NOT NULL,
[WorkFlowStepDescription] [nvarchar] (100) COLLATE SQL_Latin1_General_CP1_CI_AS NOT NULL,
[WorkFlowStepTableRowCount] [int] NULL CONSTRAINT [DF__WorkflowS__WorkF__27F8EE98] DEFAULT ((0)),
[StartingDateTime] [datetime2] NULL CONSTRAINT [DF__WorkflowS__Start__28ED12D1] DEFAULT (sysdatetime()),
[EndingDateTime] [datetime2] NULL CONSTRAINT [DF__WorkflowS__Endin__29E1370A] DEFAULT (sysdatetime()),
[ClassTime] [char] (5) COLLATE SQL_Latin1_General_CP1_CI_AS NULL CONSTRAINT [DF__WorkflowS__-Class__2AD55B43] DEFAULT ('09:15'),
[UserAuthorizationKey] [int] NOT NULL
) ON [PRIMARY]
```

GO
ALTER TABLE [Process].[WorkflowSteps] ADD CONSTRAINT [PK_Workflow_27726CBF1B202E99] PRIMARY KEY
CLUSTERED ([WorkFlowStepKey]) ON [PRIMARY]
GO

Uses

Process

Used By

[Process].[usp_ShowWorkflowSteps] [Process].[usp_TrackWorkFlow]

[Uploadfile].[CurrentSemesterCourseOfferings]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
Неар	True
Row Count (~)	4522
Created	4:31:20 PM Monday, April 12, 2021
Last Modified	4:33:21 PM Monday, April 12, 2021

Columns

Name	Data Type	Max Length (Bytes)	Nullability	Default
Semester	varchar(50)	50	NULL allowed	('Current Semester')
Sec	varchar(50)	50	NULL allowed	
Code	varchar(50)	50	NULL allowed	
Course (hr, crd)	varchar(50)	50	NULL allowed	
Description	varchar(50)	50	NULL allowed	
Day	varchar(50)	50	NULL allowed	
Time	varchar(50)	50	NULL allowed	
Instructor	varchar(50)	50	NULL allowed	
Location	varchar(50)	50	NULL allowed	
Enrolled	varchar(50)	50	NULL allowed	
Limit	varchar(50)	50	NULL allowed	
Mode of Instruction	varchar(50)	50	NULL allowed	

```
CREATE TABLE [Uploadfile].[CurrentSemesterCourseOfferings]

(
[Semester] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL CONSTRAINT [DF_Current-SemesterCourseOfferings_Semester] DEFAULT ('Current Semester'),

[Sec] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Code] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Course (hr, crd)] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Description] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Time] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Instructor] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Location] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Enrolled] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Enrolled] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,
```

Project > localhost,13001 > User databases > QueensClassSchedule > Tables > Uploadfile.CurrentSemesterCourseOfferings

```
[Limit] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL,

[Mode of Instruction] [varchar] (50) COLLATE SQL_Latin1_General_CP1_CI_AS NULL

) ON [PRIMARY]

GO
```

Uses

Uploadfile

Used By



Objects

Name

 $Utils.uvw_FindColumnDefinitionPlusDefaultAndCheckConstraint$

[Utils].[uvw_FindColumnDefinitionPlusDefaultAndCheckConstraint]

Properties

Property	Value
Collation	SQL_Latin1_General_CP1_CI_AS
ANSI Nulls On	True
Quoted Identifier On	True
Created	9:19:15 AM Sunday, October 27, 2019
Last Modified	9:19:15 AM Sunday, October 27, 2019

Columns

Name	Data Type	Max Length (Bytes)
FullyQualifiedTableName	nvarchar(257)	514
SchemaName	[sys].[sysname]	256
TableName	[sys].[sysname]	256
ColumnName	[sys].[sysname]	256
OrdinalPosition	int	4
FullyQualifiedDomainName	nvarchar(257)	514
DomainName	[sys].[sysname]	256
DataType	nvarchar(128)	256
IsNullable	varchar(3)	3
DefaultName	[sys].[sysname]	256
DefaultNameDefinition	nvarchar(4000)	8000
CheckConstraintRuleName	[sys].[sysname]	256
CheckConstraintRuleNameDefinition	nvarchar(4000)	8000

```
CREATE VIEW [Utils].[uvw_FindColumnDefinitionPlusDefaultAndCheckConstraint] AS

SELECT CONCAT(tbl.TABLE_SCHEMA, '.', tbl.TABLE_NAME) AS FullyQualifiedTableName,

tbl.TABLE_SCHEMA AS SchemaName,

tbl.TABLE_NAME AS TableName,

col.COLUMN_NAME AS ColumnName,

col.ORDINAL_POSITION AS OrdinalPosition,

CONCAT(col.DOMAIN_SCHEMA, '.', col.DOMAIN_NAME) AS FullyQualifiedDomainName,

col.DOMAIN_NAME AS DomainName,
```

```
CASE
             WHEN col.DATA TYPE = 'varchar'
             THEN CONCAT('varchar(', CHARACTER MAXIMUM LENGTH, ')')
             WHEN col.DATA TYPE = 'nvarchar'
             THEN CONCAT('nvarchar(', CHARACTER MAXIMUM LENGTH, ')')
             WHEN col.DATA TYPE = 'nchar'
             THEN CONCAT('nchar(', CHARACTER MAXIMUM LENGTH, ')')
             WHEN col.DATA TYPE = 'numeric'
             THEN CONCAT('numeric(', NUMERIC PRECISION RADIX, ', ',
                        NUMERIC SCALE, ')')
             WHEN col.DATA_TYPE = 'decimal'
             THEN CONCAT('decimal(', NUMERIC PRECISION RADIX, ', ',
                       NUMERIC SCALE, ')')
             ELSE col.DATA TYPE
        END AS DataType ,
        col.IS NULLABLE AS IsNullable,
        dcn.DefaultName ,
        col.COLUMN DEFAULT AS DefaultNameDefinition ,
        cc.CONSTRAINT NAME AS CheckConstraintRuleName,
        cc.CHECK CLAUSE AS CheckConstraintRuleNameDefinition
        ( SELECT TABLE CATALOG ,
FROM
                   TABLE SCHEMA ,
                   TABLE NAME ,
                   TABLE TYPE
                   INFORMATION SCHEMA. TABLES
         FROM
                  ( TABLE TYPE = 'BASE TABLE' )
         WHERE
        ) AS tbl
        INNER JOIN ( SELECT TABLE CATALOG ,
                            TABLE SCHEMA ,
                            TABLE NAME ,
                            COLUMN NAME ,
                            ORDINAL POSITION ,
                            COLUMN DEFAULT ,
                            IS NULLABLE ,
                            DATA TYPE ,
                            CHARACTER MAXIMUM LENGTH ,
                            CHARACTER OCTET LENGTH ,
                            NUMERIC PRECISION ,
                            NUMERIC PRECISION RADIX ,
                            NUMERIC SCALE ,
                            DATETIME PRECISION ,
                            CHARACTER SET CATALOG ,
                            CHARACTER SET SCHEMA ,
                            CHARACTER SET NAME ,
                            COLLATION CATALOG ,
                            COLLATION SCHEMA ,
                            COLLATION NAME ,
                            DOMAIN CATALOG ,
                            DOMAIN SCHEMA ,
                            DOMAIN NAME
                     FROM INFORMATION SCHEMA.COLUMNS
                   ) AS col ON col.TABLE CATALOG = tbl.TABLE CATALOG
```

```
AND col.TABLE SCHEMA = tbl.TABLE SCHEMA
                               AND col.TABLE NAME = tbl.TABLE NAME
       LEFT OUTER JOIN ( SELECT t.name AS TableName ,
                                   SCHEMA_NAME(s.schema_id) AS SchemaName ,
                                   ac.name AS ColumnName ,
                                   d.name AS DefaultName
                          FROM
                                  sys.all columns AS ac
                                   INNER JOIN sys.tables AS t ON ac.object id = t.object id
                                   INNER JOIN sys.schemas AS s ON t.schema id = s.schema id
                                    INNER JOIN sys.default_constraints AS d ON
ac.default object id = d.object id
                       ) AS dcn ON dcn.SchemaName = tbl.TABLE SCHEMA
                                   AND dcn.TableName = tbl.TABLE NAME
                                   AND dcn.ColumnName = col.COLUMN NAME
       LEFT OUTER JOIN ( SELECT
                                   cu.TABLE CATALOG ,
                                   cu.TABLE SCHEMA ,
                                    cu.TABLE NAME ,
                                    cu.COLUMN_NAME ,
                                    c.CONSTRAINT CATALOG ,
                                    c.CONSTRAINT SCHEMA ,
                                    c.CONSTRAINT NAME ,
                                   c.CHECK CLAUSE
                          FROM
                                   INFORMATION SCHEMA.CONSTRAINT COLUMN USAGE
                                   AS cu
                                   INNER JOIN INFORMATION SCHEMA.CHECK CONSTRAINTS
                                   AS c ON c.CONSTRAINT NAME = cu.CONSTRAINT NAME
                       ) AS cc ON cc.TABLE_SCHEMA = tbl.TABLE_SCHEMA
                                  AND cc.TABLE_NAME = tbl.TABLE_NAME
                                  AND cc.COLUMN NAME = col.COLUMN NAME
GO
```

Uses

Utils

■ Stored Procedures

Objects

Name
Name
dbo.LoadDataFromUpload
dbo.sp_AddForeignKey
dbo.sp_DropForeignKey
dbo.sp_TruncateQueensClassTables
Process.usp_ShowWorkflowSteps
Process.usp_TrackWorkFlow

[dbo].[LoadDataFromUpload]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

```
-- -----
                <Evnul Hossain>
-- Create date: <05/5/2024>
-- Description: <To load data from the upload file given to us>
CREATE PROCEDURE [dbo].[LoadDataFromUpload]
BEGIN
   SET NOCOUNT ON;
   -- Load data into CourseManagement.Course table, avoiding duplicates
   INSERT INTO CourseManagement.Course (CourseCode, CourseName, Credits, InstructorId, User-
AuthorizationKey, DateAdded, DateOfLastUpdate)
   SELECT DISTINCT
       [Code] AS CourseCode,
       [Description] AS CourseName,
       CASE
           WHEN CHARINDEX('(', [Course (hr, crd)]) > 0 AND CHARINDEX(')', [Course (hr, crd)]) >
0
               THEN TRY CAST(SUBSTRING([Course (hr, crd)], CHARINDEX('(', [Course (hr, crd)]) +
1, CHARINDEX(')', [Course (hr, crd)]) - CHARINDEX('(', [Course (hr, crd)]) - 1) AS DECIMAL(10,
2))
           ELSE 0 -- Set default value for Credits if extraction fails
       END AS Credits,
       HASHBYTES('MD5', [Instructor]) AS InstructorId, -- Assuming numeric InstructorId
       1 AS UserAuthorizationKey,
       GETDATE() AS DateAdded,
       GETDATE() AS DateOfLastUpdate
   FROM [QueensClassSchedule].[Uploadfile].[CurrentSemesterCourseOfferings]
   WHERE [Code] NOT IN (SELECT CourseCode FROM CourseManagement.Course);
   -- Load data into InstructorManagement.Instructor table with valid names
   INSERT INTO InstructorManagement.Instructor (LastName, FirstName, MaxCredits, User-
AuthorizationKey, DateAdded, DateOfLastUpdate)
   SELECT DISTINCT
       PARSENAME ([Instructor], 1) AS LastName,
       PARSENAME([Instructor], 2) AS FirstName,
       15 AS MaxCredits,
```

```
1 AS UserAuthorizationKey,
        GETDATE() AS DateAdded,
        GETDATE() AS DateOfLastUpdate
    FROM [QueensClassSchedule].[Uploadfile].[CurrentSemesterCourseOfferings]
    WHERE [Instructor] IS NOT NULL
   AND [Instructor] != ' '
   AND LEN(RTRIM(LTRIM([Instructor]))) > 0
    AND LEN(RTRIM(LTRIM(PARSENAME([Instructor], 2)))) > 0;
    -- Load data into LocationManagement.BuildingLocation table
    INSERT INTO LocationManagement.BuildingLocation (BuildingName, UserAuthorizationKey, Date-
Added, DateOfLastUpdate)
    SELECT DISTINCT
        [Location] AS BuildingName,
        1 AS UserAuthorizationKey,
        GETDATE() AS DateAdded,
        GETDATE() AS DateOfLastUpdate
   FROM [QueensClassSchedule].[Uploadfile].[CurrentSemesterCourseOfferings]
    WHERE [Location] IS NOT NULL;
    -- Load data into DepartmentManagement.Department table
    INSERT INTO DepartmentManagement.Department (DepartmentName, DepartmentCode, MaxEnrollment,
UserAuthorizationKey, DateAdded, DateOfLastUpdate)
    SELECT DISTINCT
        SUBSTRING([Description], 1, CHARINDEX(' - ', [Description]) - 1) AS DepartmentName,
        SUBSTRING([Description], CHARINDEX(' - ', [Description]) + 3, LEN([Description])) AS
DepartmentCode,
        100 AS MaxEnrollment,
        1 AS UserAuthorizationKey,
        GETDATE() AS DateAdded,
        GETDATE() AS DateOfLastUpdate
    FROM [QueensClassSchedule].[Uploadfile].[CurrentSemesterCourseOfferings]
    WHERE [Description] LIKE '% - %' AND SUBSTRING([Description], 1, CHARINDEX(' - ',
[Description]) - 1) NOT IN (SELECT DepartmentName FROM DepartmentManagement.Department);
    -- Check and insert ModeOfInstruction for 'In-Person', 'Hybrid', and 'Online'
    IF NOT EXISTS (SELECT 1 FROM InstructionManagement.ModeOfInstruction WHERE ModeName = 'In-
Person')
    BEGIN
        INSERT INTO InstructionManagement.ModeOfInstruction (ModeName, UserAuthorizationKey,
DateAdded, DateOfLastUpdate)
        VALUES ('In-Person', 1, GETDATE());
    END
   IF NOT EXISTS (SELECT 1 FROM InstructionManagement.ModeOfInstruction WHERE ModeName =
'Hybrid')
   BEGIN
       INSERT INTO InstructionManagement.ModeOfInstruction (ModeName, UserAuthorizationKey,
DateAdded, DateOfLastUpdate)
       VALUES ('Hybrid', 1, GETDATE(), GETDATE());
    END
     \texttt{IF} \texttt{ NOT} \texttt{ EXISTS} \texttt{ (SELECT 1 FROM InstructionManagement.ModeOfInstruction WHERE ModeName} = \\
```

```
BEGIN
        INSERT INTO InstructionManagement.ModeOfInstruction (ModeName, UserAuthorizationKey,
DateAdded, DateOfLastUpdate)
       VALUES ('Online', 1, GETDATE(), GETDATE());
   END
   IF NOT EXISTS (SELECT 1 FROM InstructionManagement.ModeOfInstruction WHERE ModeName = 'Web-
Enhanced')
   BEGIN
       INSERT INTO InstructionManagement.ModeOfInstruction (ModeName, UserAuthorizationKey,
DateAdded, DateOfLastUpdate)
       VALUES ('Web-Enhanced', 1, GETDATE(), GETDATE());
   END
     -- Load data into CourseManagement.Class table with valid CourseId and ModeId
   INSERT INTO CourseManagement.Class (CourseId, Schedule, MaxEnrollment, ModeId, User-
AuthorizationKey, DateAdded, DateOfLastUpdate)
   SELECT DISTINCT
      cm.CourseId,
       c.[Time] AS Schedule,
       c.[Limit] AS MaxEnrollment,
       mi.ModeId, -- Insert ModeId from InstructionManagement.ModeOfInstruction
       1 AS UserAuthorizationKey,
        GETDATE() AS DateAdded,
       GETDATE() AS DateOfLastUpdate
   FROM [QueensClassSchedule].[Uploadfile].[CurrentSemesterCourseOfferings] c
   JOIN CourseManagement.Course cm ON cm.CourseCode = c.[Code]
   JOIN InstructionManagement.ModeOfInstruction mi ON mi.ModeName = c.[Mode Of Instruction]
   WHERE EXISTS (SELECT 1 FROM CourseManagement.Course WHERE CourseCode = c.[Code])
   AND mi.ModeId IS NOT NULL;
   PRINT 'Data loaded successfully into all tables.';
END
GO
```

Uses

[CourseManagement].[Class]
[CourseManagement].[Course]
[DepartmentManagement].[Department]
[InstructionManagement].[ModeOfInstruction]
[InstructorManagement].[Instructor]
[LocationManagement].[BuildingLocation]
[Uploadfile].[CurrentSemesterCourseOfferings]

[dbo].[sp_AddForeignKey]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

Parameters

Name	Data Type	Max Length (Bytes)
@ForeignKeyName	nvarchar(50)	100
@TableName	nvarchar(50)	100
@ColumnName	nvarchar(50)	100
@ReferencedTableName	nvarchar(50)	100
@ReferencedColumnName	nvarchar(50)	100

```
-- Author: Haiim Lalehzarzadeh
-- Procedure: sp_AddForeignKey
-- Create date: 2024-05-15
-- Description: Adds a foreign key constraint to a table
CREATE PROCEDURE [dbo].[sp_AddForeignKey]
   @ForeignKeyName NVARCHAR(50),
   @TableName NVARCHAR(50),
   @ColumnName NVARCHAR(50),
   @ReferencedTableName NVARCHAR (50),
   @ReferencedColumnName NVARCHAR(50)
AS
BEGIN
   SET NOCOUNT ON;
   DECLARE @SQL NVARCHAR (MAX);
   SET @SQL = 'ALTER TABLE ' + QUOTENAME(@TableName) +
               ' ADD CONSTRAINT ' + QUOTENAME(@ForeignKeyName) +
               ' FOREIGN KEY (' + QUOTENAME(@ColumnName) + ')' +
               ' REFERENCES ' + QUOTENAME (@ReferencedTableName) +
               ' (' + QUOTENAME(@ReferencedColumnName) + ');';
    EXEC sp_executesql @SQL;
```

Project > localhost,13001 > User databases > QueensClass-Schedule > Programmability > Stored Procedures > dbo.sp_Add-ForeignKey

```
PRINT 'Foreign key ' + @ForeignKeyName + ' added successfully.';
END;
GO
```

[dbo].[sp_DropForeignKey]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

Parameters

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

```
-- Author:
           <Haiim Lalehzarzadeh>
-- Create date: <05/5/2024>
-- Description: <To drop foreign keys from tables>
-- -----
   CREATE PROCEDURE [dbo].[sp_DropForeignKey]
      @ForeignKeyName NVARCHAR(50),
       @TableName NVARCHAR(50)
   AS
   BEGIN
      SET NOCOUNT ON;
       IF EXISTS (
          SELECT 1
          FROM sys.foreign_keys
          WHERE name = @ForeignKeyName
          AND parent object id = OBJECT ID(@TableName)
       BEGIN
           DECLARE @SQL NVARCHAR (MAX);
           SET @SQL = 'ALTER TABLE ' + QUOTENAME(@TableName) +
                     ' DROP CONSTRAINT ' + QUOTENAME(@ForeignKeyName);
           EXEC sp executesql @SQL;
           PRINT 'Foreign key ' + @ForeignKeyName + ' dropped successfully.';
       END
       ELSE
```

Project > localhost,13001 > User databases > QueensClass-Schedule > Programmability > Stored Procedures > dbo.sp_Drop-ForeignKey

```
BEGIN

PRINT 'Foreign key ' + @ForeignKeyName + ' does not exist on ' + @TableName + '.';

END

END;

GO
```

[dbo].[sp_TruncateQueensClassTables]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

```
-- -----
-- Author: Hanlin Wang
-- Procedure: sp_TruncateQueensClassTables
-- Create date: 2024-05-10
-- Description: Truncates all tables related to QueensClassSchedule database
CREATE PROCEDURE [dbo].[sp TruncateQueensClassTables]
AS
BEGIN
   SET NOCOUNT ON;
   DECLARE @TableName NVARCHAR(255);
   DECLARE @SchemaName NVARCHAR(255);
   DECLARE @ConstraintName NVARCHAR(255);
   DECLARE @SQL NVARCHAR (MAX);
   -- Drop foreign key constraints
   DECLARE constraint cursor CURSOR FOR
   SELECT s.name AS SchemaName, t.name AS TableName, fk.name AS ConstraintName
   FROM sys.foreign keys fk
   INNER JOIN sys.tables t ON fk.parent object id = t.object id
   INNER JOIN sys.schemas s ON t.schema id = s.schema id
   WHERE s.name IN ('Udt', 'DepartmentManagement', 'InstructorManagement', 'LocationManagement',
'InstructionManagement', 'CourseManagement');
   OPEN constraint cursor;
   FETCH NEXT FROM constraint_cursor INTO @SchemaName, @TableName, @ConstraintName;
   WHILE @@FETCH STATUS = 0
      SET @SQL = 'ALTER TABLE ' + QUOTENAME (@SchemaName) + '.' + QUOTENAME (@TableName) + ' DROP
CONSTRAINT ' + QUOTENAME(@ConstraintName) + ';';
       EXEC sp executesql @SQL;
       FETCH NEXT FROM constraint cursor INTO @SchemaName, @TableName, @ConstraintName;
   END;
```

```
CLOSE constraint cursor;
   DEALLOCATE constraint cursor;
   -- Truncate tables
   DECLARE table cursor CURSOR FOR
   SELECT s.name AS SchemaName, t.name AS TableName
   FROM sys.tables t
   INNER JOIN sys.schemas s ON t.schema id = s.schema id
   WHERE s.name IN ('Udt', 'DepartmentManagement', 'InstructorManagement', 'LocationManagement',
'InstructionManagement', 'CourseManagement');
   OPEN table cursor;
   FETCH NEXT FROM table_cursor INTO @SchemaName, @TableName;
   WHILE @@FETCH STATUS = 0
   BEGIN
       SET @SQL = 'TRUNCATE TABLE ' + QUOTENAME (@SchemaName) + '.' + QUOTENAME (@TableName) +
1,10
       EXEC sp executesql @SQL;
       FETCH NEXT FROM table cursor INTO @SchemaName, @TableName;
   END:
   CLOSE table cursor;
   DEALLOCATE table cursor;
   -- Recreate foreign key constraints
   OPEN constraint cursor;
   FETCH NEXT FROM constraint cursor INTO @SchemaName, @TableName, @ConstraintName;
   WHILE @@FETCH STATUS = 0
   BEGIN
        -- The script to create the foreign key constraint needs to be stored or retrieved
        -- Assuming they are stored in another table or known, you need to recreate them here
       -- Example:
       -- SET @SQL = 'ALTER TABLE ' + QUOTENAME(@SchemaName) + '.' + QUOTENAME(@TableName) +
        -- ' ADD CONSTRAINT ' + QUOTENAME(@ConstraintName) + ' FOREIGN KEY (Column) REFERENCES
OtherTable(Column);';
       -- EXEC sp executesql @SQL;
       FETCH NEXT FROM constraint cursor INTO @SchemaName, @TableName, @ConstraintName;
   END;
   CLOSE constraint cursor;
   DEALLOCATE constraint_cursor;
   PRINT 'All QueensClassSchedule related tables truncated successfully.';
END:
GO
```

[Process].[usp_ShowWorkflowSteps]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

SQL Script

Uses

[Process].[WorkflowSteps]

Process

[Process].[usp_TrackWorkFlow]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

Parameters

Name	Data Type	Max Length (Bytes)
@StartTime	datetime2	8
@WorkFlowDescription	nvarchar(100)	200
@WorkFlowStepTableRowCount	int	4
@UserAuthorizationKey	int	4

```
-- Author: <Evnul Hossain>
-- Create date: <05/5/2024>
-- Description: <To show track each WorkFlowStep within the database when processing>
-- Create the usp_TrackWorkFlow stored procedure
CREATE PROCEDURE [Process].[usp_TrackWorkFlow]
            @StartTime DATETIME2,
              @WorkFlowDescription NVARCHAR (100),
             @WorkFlowStepTableRowCount INT,
            @UserAuthorizationKey INT
AS
BEGIN
           SET NOCOUNT ON;
              -- Insert tracking data into the WorkflowSteps table
            INSERT INTO Process.WorkflowSteps (WorkFlowStepKey, WorkFlowStepDescription, WorkFlowStepDescrip
TableRowCount, StartingDateTime, UserAuthorizationKey)
            VALUES (
                            (SELECT ISNULL(MAX(WorkFlowStepKey), 0) + 1 FROM Process.WorkflowSteps),
                             @WorkFlowDescription,
                            @WorkFlowStepTableRowCount,
                            @StartTime,
                             @UserAuthorizationKey
              );
```

Project > localhost,13001 > User databases > QueensClass-Schedule > Programmability > Stored Procedures > Process.usp_-TrackWorkFlow

```
PRINT 'Workflow step tracked successfully.';
END;
GO
```

Uses

[Process].[WorkflowSteps] Process

■ Table-valued Functions

Objects

Name

dbo.DatabaseObjects

dbo.JSONHierarchy

🖪 [dbo].[DatabaseObjects]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

Parameters

Name	Data Type	Max Length (Bytes)
@ListOfObjectIDs	varchar(max)	max

```
create function [dbo].[DatabaseObjects]
   Summary: >
    lists out the full names, schemas and (where appropriate)
     the owner of the object.
   Author: PhilFactor
   Date: 10/9/2017
   Examples:
      - Select * from dbo.DatabaseObjects('2123154609,960722475,1024722703')
    A table with the id, name of object and so on.
   @ListOfObjectIDs varchar(max)
returns table
--WITH ENCRYPTION|SCHEMABINDING, ..
as
return
   select object id
        , schema name(schema id) + '.' + coalesce(object name(parent object id) + '.', '') +
name as name
   from sys.objects as ob
       inner join openjson(N'[' + @ListOfObjectIDs + N']')
          on convert(int, Value) = ob.object id
);
```

[dbo].[JSONHierarchy]

Properties

Property	Value
ANSI Nulls On	True
Quoted Identifier On	True

Parameters

Name	Data Type	Max Length (Bytes)
@JSONData	varchar(max)	max
@Parent_object_ID	int	4
@MaxObject_id	int	4
@type	int	4

```
create function [dbo].[JSONHierarchy]
   @JSONData varchar(max)
  , @Parent object ID int = null
 , @MaxObject id int = 0
  , @type int = null
returns @ReturnTable table
   -- https://www.red-gate.com/simple-talk/blogs/consuming-hierarchical-json-documents-sql-
server-using-openjson/?utm source=simpletalk&utm medium=pubemail&utm content=20171010-
slota1&utm term=simpletalkmain
   Element_ID int identity(1, 1) primary key /* internal surrogate primary key gives the order
of parsing and the list order */
 , SequenceNo int null
                                             /* the sequence number in a list */
                                             /* if the element has a parent then it is in this
  , Parent ID int
column. The document is the ultimate parent, so you can get the structure from recursing from the
document */
  , Object ID int
                                             /* each list or object has an object id. This ties
all elements to a parent. Lists are treated as objects here */
 , Name nvarchar(2000)
                                            /* the name of the object */
  , StringValue nvarchar(max) not null
                                            /*the string representation of the value of the
element. */
  , ValueType varchar(10) not null
                                            /* the declared type of the value represented as a
string in StringValue*/
```

```
begin
   --the types of JSON
   declare @null int = 0
         , @string int = 1
         , @int int = 2
         , @boolean int = 3
         , @object int = 5;
   declare @OpenJSONData table
      sequence int identity(1, 1)
     , [key] varchar(200)
     , Value varchar(max)
     , type int
   );
   declare @key
                   varchar(200)
       , @Value varchar(max)
         , @Thetype int
         , @ii
                    int
                  int
         , @iiMax
         , @NewObject int
         , @firstchar char(1);
   insert into @OpenJSONData
       [key]
     , Value
     , type
   select [Key]
       , Value
        , Type
   from openjson(@JSONData);
   select @ii = 1
        , @iiMax = scope identity();
   select @firstchar
       = --the first character to see if it is an object or an array
       substring(
                  @JSONData
                  , patindex(
                              '%[^' + char(0) + '- ' + char(160) + ']%'
                             , ' ' + @JSONData + '!!' collate SQL_Latin1_General_CP850_BIN
                           ) - 1
                 , 1
               );
   if @type is null
      and @firstchar in ( '[', '{')
       insert into @ReturnTable
```

```
SequenceNo
      , Parent ID
      , Object ID
      , Name
      , StringValue
      , ValueType
    select 1
         , null
         , 1
         , '-1
         , 11
         , case @firstchar
               when '[' then
                  'array'
               else
                  'object'
           end:
    select @type
                             = case @firstchar
                                   when '[' then
                                       @array
                                   else
                                       @object
                               end
         , @Parent_object_ID = 1
         , @MaxObject_id = coalesce(@MaxObject_id, 1) + 1;
end;
while (@ii <= @iiMax)</pre>
begin
    --OpenJSON renames list items with 0-nn which confuses the consumers of the table
   select @key = case
                          when [key] like '[0-9]%' then
                          else
                             [key]
                      end
         , @Value = Value
         , @Thetype = type
    from @OpenJSONData
    where sequence = @ii;
   if @Thetype in ( @array, @object ) --if we have been returned an array or object
   begin
        select @MaxObject id = coalesce(@MaxObject id, 1) + 1;
        --just in case we have an object or array returned
        insert into @ReturnTable --record the object itself
           SequenceNo
          , Parent ID
          , Object ID
          , Name
          , StringValue
```

```
, ValueType
   select @ii
        , @Parent_object_ID
        , @MaxObject_id
        , @key
        , 11
         , case @Thetype
              when @array then
                 'array'
              else
                  'object'
           end;
   insert into @ReturnTable --and return all its children
       SequenceNo
     , Parent_ID
     , Object_ID
      , [Name]
     , StringValue
     , ValueType
   select SequenceNo
        , Parent_ID
        , Object_ID
        , [Name]
        , StringValue
        , ValueType
   from dbo.JSONHierarchy(@Value, @MaxObject_id, @MaxObject_id, @type);
   select @MaxObject id = max(Object ID) + 1
   from @ReturnTable;
end:
else
   insert into @ReturnTable
       SequenceNo
     , Parent ID
     , Object_ID
     , Name
     , StringValue
     , ValueType
   select @ii
        , @Parent_object_ID
        , null
        , @key
        , @Value
         , case @Thetype
              when @string then
                  'string'
              when @null then
```

User-Defined Data Types

Objects

Name
Udt.BuildingName
Udt.ClassTime
Udt.CourseCode
Udt.CourseName
Udt.DateAdded
Udt.DateOfLastUpdate
Udt.DepartmentCode
Udt.DepartmentName
Udt.FirstName
Udt.FullName
Udt.GroupName
Udt.IndividualProject
Udt.LastName
Udt.NodeName
Udt.OldSurrogateKeyInt
Udt.RoomNumber
Udt.Schedule
Udt.SurrogateKeyInt

[Udt].[BuildingName]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	100

SQL Script

CREATE TYPE [Udt].[BuildingName] FROM nvarchar (100) NOT NULL
GO

Uses

[Udt].[ClassTime]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nchar
Length	5

SQL Script

CREATE TYPE [Udt].[ClassTime] FROM nchar (5) NOT NULL
GO

Uses

Udt

Used By

[Udt].[CourseCode]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	20

SQL Script

CREATE TYPE [Udt].[CourseCode] FROM nvarchar (20) NOT NULL
GO

Uses

[Udt].[CourseName]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	100

SQL Script

CREATE TYPE [Udt].[CourseName] FROM nvarchar (100) NOT NULL
GO

Uses

[Udt].[DateAdded]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	datetime
Length	8

SQL Script

CREATE TYPE [Udt].[DateAdded] FROM datetime NOT NULL
GO

Uses

Udt

Used By

[CourseManagement].[Class]
[CourseManagement].[Course]
[DbSecurity].[UserAuthorization]
[DepartmentManagement].[Department]
[InstructionManagement].[ModeOfInstruction]
[InstructorManagement].[Instructor]
[LocationManagement].[BuildingLocation]
[LocationManagement].[RoomLocation]

[Udt].[DateOfLastUpdate]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	datetime
Length	8

SQL Script

CREATE TYPE [Udt].[DateOfLastUpdate] FROM datetime NOT NULL
GO

Uses

Udt

Used By

[CourseManagement].[Class]
[CourseManagement].[Course]
[DepartmentManagement].[Department]
[InstructionManagement].[ModeOfInstruction]
[InstructorManagement].[Instructor]
[LocationManagement].[BuildingLocation]
[LocationManagement].[RoomLocation]

[Udt].[DepartmentCode]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	20

SQL Script

CREATE TYPE [Udt].[DepartmentCode] FROM nvarchar (20) NOT NULL
GO

Uses

[Udt].[DepartmentName]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	100

SQL Script

CREATE TYPE [Udt].[DepartmentName] FROM nvarchar (100) NOT NULL
GO

Uses

[Udt].[FirstName]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	20

SQL Script

CREATE TYPE [Udt].[FirstName] FROM nvarchar (20) NOT NULL
GO

Uses

Udt

Used By

[Udt].[FullName]

Properties

Property	Value
Nullability	NULL allowed
Base Type Name	nvarchar
Length	100

SQL Script

CREATE TYPE [Udt].[FullName] FROM nvarchar (100) NULL
GO

Uses

[Udt].[GroupName]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	20

SQL Script

CREATE TYPE [Udt].[GroupName] FROM nvarchar (20) NOT NULL
GO

Uses

Udt

Used By

[Udt].[IndividualProject]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	60

SQL Script

CREATE TYPE [Udt].[IndividualProject] FROM nvarchar (60) NOT NULL
GO

Uses

Udt

Used By

[Udt].[LastName]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	35

SQL Script

CREATE TYPE [Udt].[LastName] FROM nvarchar (35) NOT NULL
GO

Uses

Udt

Used By

[Udt].[NodeName]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	50

SQL Script

CREATE TYPE [Udt].[NodeName] FROM nvarchar (50) NOT NULL
GO

Uses

[Udt].[OldSurrogateKeyInt]

Properties

Property	Value
Nullability	NULL allowed
Base Type Name	int
Length	4

SQL Script

CREATE TYPE [Udt].[OldSurrogateKeyInt] FROM int NULL
GO

Uses

[Udt].[RoomNumber]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	20

SQL Script

CREATE TYPE [Udt].[RoomNumber] FROM nvarchar (20) NOT NULL
GO

Uses

[Udt].[Schedule]

Properties

Property	Value
Nullability	NOT NULL
Base Type Name	nvarchar
Length	50

SQL Script

CREATE TYPE [Udt].[Schedule] FROM nvarchar (50) NOT NULL
GO

Uses

[Udt].[SurrogateKeyInt]

Properties

Property	Value
Nullability	NULL allowed
Base Type Name	int
Length	4

SQL Script

CREATE TYPE [Udt].[SurrogateKeyInt] FROM int NULL
GO

Uses

Used By

Udt

-

[CourseManagement].[Class]

[Course Management]. [Course]

[DbSecurity].[UserAuthorization]

[Department Management]. [Department]

[Instruction Management]. [Mode Of Instruction]

[InstructorManagement].[Instructor]

[LocationManagement].[BuildingLocation]

[LocationManagement].[RoomLocation]

1 Users

Objects

Name
BiStudent
dbo
EC3\thehitman
guest
student



Property	Value
Туре	SqlUser
Default Schema	dbo

Database Level Permissions

Туре	Action
CONNECT	Grant

SQL Script

CREATE USER [BiStudent] WITHOUT LOGIN GO



Property	Value
Туре	SqlUser
Login Name	sa
Default Schema	dbo

Database Level Permissions

Туре	Action
CONNECT	Grant

SQL Script

GO



Property	Value
Туре	WindowsUser
Login Name	EC3\thehitman
Default Schema	dbo

Database Level Permissions

Туре	Action
CONNECT	Grant

SQL Script

```
IF NOT EXISTS (SELECT * FROM master.dbo.syslogins WHERE loginname = N'EC3\thehitman')
CREATE LOGIN [EC3\thehitman] FROM WINDOWS
GO
CREATE USER [EC3\thehitman] FOR LOGIN [EC3\thehitman]
GO
```



Property	Value
Туре	SqlUser
Default Schema	guest

SQL Script

GO



Property	Value
Туре	SqlUser
Default Schema	dbo

Database Level Permissions

Туре	Action
CONNECT	Grant

SQL Script

CREATE USER [student] WITHOUT LOGIN GO

La Database Roles

Objects

Name
db_accessadmin
db_backupoperator
db_datareader
db_datawriter
db_ddladmin
db_denydatareader
db_denydatawriter
db_owner
db_securityadmin
public

♣ db_accessadmin

Properties

Property	Value
Owner	dbo

db_backupoperator

Properties

Property	Value
Owner	dbo

db_datareader

Properties

Property	Value
Owner	dbo

Members

- BiStudent
- student

SQL Script

```
ALTER ROLE [db_datareader] ADD MEMBER [BiStudent]

GO

ALTER ROLE [db_datareader] ADD MEMBER [student]

GO
```

Uses

BiStudent student

db_datawriter

Properties

Property	Value
Owner	dbo

db_ddladmin

Properties

Property	Value
Owner	dbo

db_denydatareader

Properties

Property	Value
Owner	dbo

db_denydatawriter

Properties

Property	Value
Owner	dbo

♣ db_owner

Property	Value
Owner	dbo

Members

• EC3\thehitman

SQL Script

ALTER ROLE [db_owner] ADD MEMBER [EC3\thehitman]
GO

Uses

EC3\thehitman

♣ db_securityadmin

Properties

Property	Value
Owner	dbo

🚣 public

Properties

Property	Value
Owner	dbo

△ Schemas

Objects

Name
CourseManagement
DbSecurity
DepartmentManagement
GroupNameProject3
InstructionManagement
InstructorManagement
LocationManagement
Process
Udt
Uploadfile
Utils
YourLastName

△ CourseManagement

Properties

Property	Value
Owner	dbo

SQL Script

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

Used By

[CourseManagement].[Class] [CourseManagement].[Course]

♣ DbSecurity

Properties

Property	Value
Owner	dbo

SQL Script

CREATE SCHEMA [DbSecurity]
AUTHORIZATION [dbo]
GO

Used By

[DbSecurity].[UserAuthorization]

♣ DepartmentManagement

Properties

Property	Value
Owner	dbo

SQL Script

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

Used By

[DepartmentManagement].[Department]

Properties

Property	Value
Owner	dbo

SQL Script

CREATE SCHEMA [GroupNameProject3]
AUTHORIZATION [dbo]
GO

♣ InstructionManagement

Properties

Property	Value
Owner	dbo

SQL Script

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

Used By

[InstructionManagement].[ModeOfInstruction]

△ InstructorManagement

Properties

Property	Value
Owner	dbo

SQL Script

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

Used By

[InstructorManagement].[Instructor]

△ LocationManagement

Properties

Property	Value
Owner	dbo

SQL Script

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

Used By

[LocationManagement].[BuildingLocation] [LocationManagement].[RoomLocation]

△ Process

Properties

Property	Value
Owner	dbo

SQL Script

CREATE SCHEMA [Process]
AUTHORIZATION [dbo]
GO

Used By

[Process].[WorkflowSteps]
[Process].[usp_ShowWorkflowSteps]
[Process].[usp_TrackWorkFlow]

. Udt

Properties

Property	Value
Owner	dbo

SQL Script

CREATE SCHEMA [Udt]
AUTHORIZATION [dbo]
GO

Used By

[Udt].[BuildingName]

[Udt].[ClassTime]

[Udt].[CourseCode]

[Udt].[CourseName]

[Udt].[DateAdded]

[Udt].[DateOfLastUpdate]

[Udt].[DepartmentCode]

[Udt].[DepartmentName]

[Udt].[FirstName]

[Udt].[FullName]

[Udt].[GroupName]

[Udt].[IndividualProject]

[Udt].[LastName]

[Udt].[NodeName]

[Udt].[OldSurrogateKeyInt]

[Udt].[RoomNumber]

[Udt].[Schedule]

[Udt].[SurrogateKeyInt]

♣ Uploadfile

Properties

Property	Value
Owner	dbo

SQL Script

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

Used By

[Upload file]. [Current Semester Course Offerings]



Property	Value
Owner	dbo

SQL Script

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

Used By

 $[Utils]. [uvw_FindColumnDefinitionPlusDefaultAndCheckConstraint] \\$

Properties

Property	Value
Owner	dbo

SQL Script

CREATE SCHEMA [YourLastName]
AUTHORIZATION [dbo]
GO