CTI EDUCATION GROUP

Suzi Yoga Studio

Project 3

Tejas Dwarkaram

2012

User Documentation

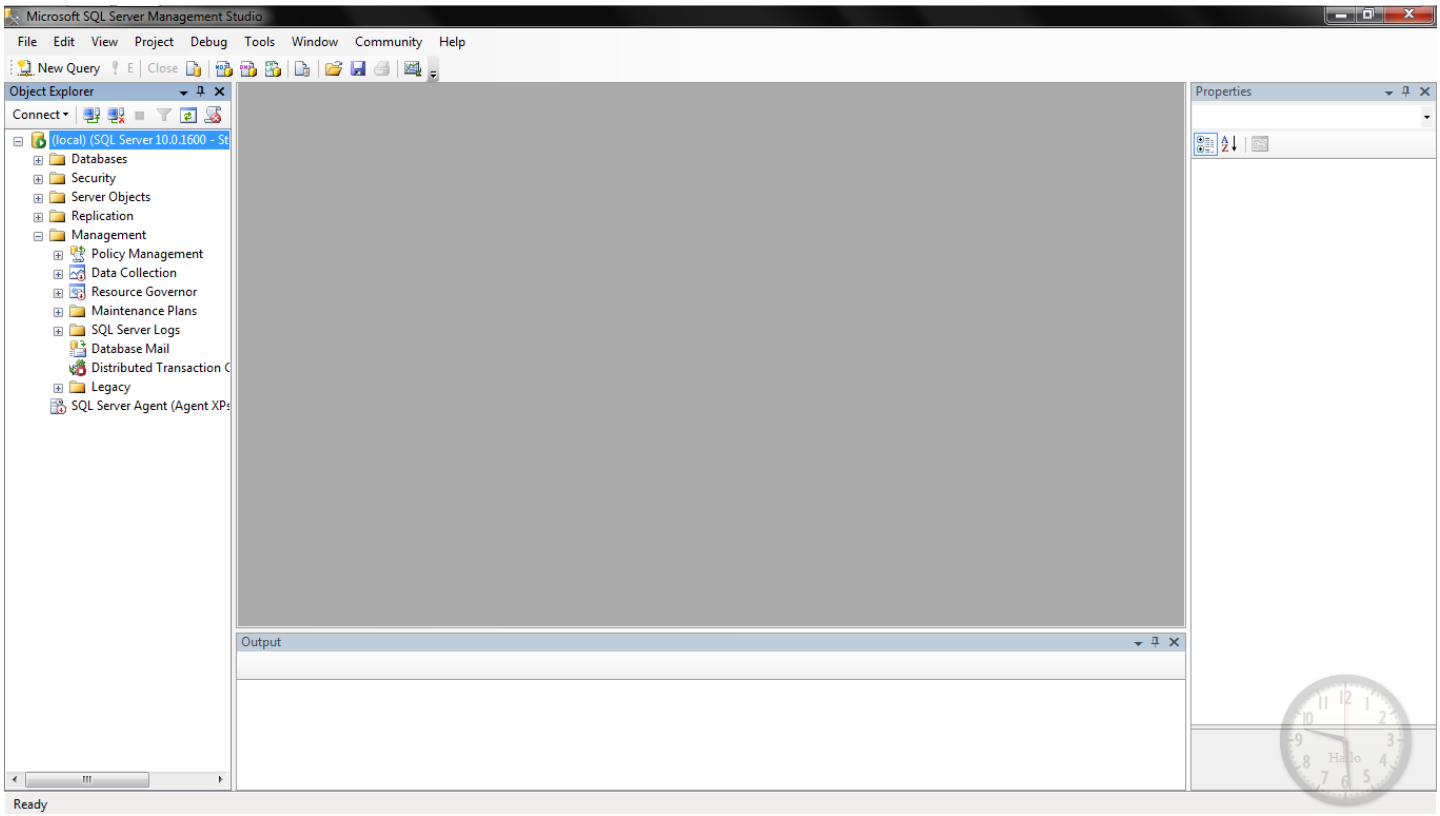
TEJAS DWARKARAM NS2012-0340

Table of Contents

Using SQL Management Studio

To begin SQL Management Studio, double click the desktop shortcut
(or if not found on desktop, locate the 'Microsoft SQL Server 2008' folder in the start menu)

Once opened connect to the '(local)' server, and then you should be presented with the following layout



To use any of the codes that are going to follow, click on "New Query"(usually found in the top left corner).
Please Note.
To execute a query press "F5"
And to end a query press "Alt" + "Break/Pause" together.

Using the created views

A view is used to view certain values of any table. What is represented is a kind of filter that contains data that is specified to be shown.

To use a view

A SELECT statement, is used on which we 'SELECT' everything from the 'view container'

The following needs to be typed in a 'New Query Editor Window'.

To view vw_Exercises

```
USE Suzi_Yoga_Studio
GO

SELECT *
FROM vw_Exercises
```

To view vw_ExerciseUsed

```
USE Suzi_Yoga_Studio
GO

SELECT *
FROM vw_ExerciseUsed
```

To view vw_ClassAttendance

```
USE Suzi_Yoga_Studio
GO

SELECT *
FROM vw_ClasAttendance
```

To view vw_TotalClassesPerMember

```
USE Suzi_Yoga_Studio
GO

SELECT *
FROM vw_TotalClassesPerMember
```

Using the created Stored Procedures

Stored procedures are also referred to as precompiled statements. This means that after one compilation, it is ready to be reused over and over again.

To use a stored procedure

We use the 'EXEC' or 'EXECUTE' statement to run a stored procedure. Most stored procedures have certain variables that must be provided for the procedure to function.

Eg.

A stored procedure for adding 2 numbers, will require 2 numbers. Variables are shown with "@" in the front of the variable name.

The format of a stored procedure is usually as follows;

<..procedure_name..>(@variable1, @variable2)

For example.

addNumbers(@num1, @num2)

So if we were to want to add 5 and 3, the code will look like this

```
USE Suzi_Yoga_Studio
GO

EXECUTE addNumbers @num1 = 3, @num2 = 5
GO
```

And upon execution(pressF5)

The answer '8' will be printed.

The following needs to be typed in a 'New Query Editor Window'.

To use sp_AddNewExercise

Variables required

@bookID which is the ID of the book
@name which is the name of the book
@description which is the description of the book
@length which is how long the exercise must be done for
@timesdone which is how many times the exercise must be done

What to type in order to execute the procedure

```
USE Suzi_Yoga_Studio
GO

EXEC sp_AddNewExercise @bookID=<..'bookID'..>, @name=<..'name'..>,
                       @description=<..'description'..>, @length=<..'length'..>,
                       @timesdone=<..'timestobedone'..>
GO
```

After execution a new record will be added to the Exercises table.

The following needs to be typed in a 'New Query Editor Window'.

To use sp_UpdateExerciseTimesUsed

Variables required

@bookID which is the ID of the book

@times which is how many times the exercise must be done

What to type in order to execute the procedure

```
USE Suzi_Yoga_Studio
GO

EXEC sp_UpdateExerciseTimeUsed  @bookID=<..'bookID'..>,
@timesdone=<..'timestobedone'..>
GO
```

After execution the number of times that the exercise will need to be done will be updated where the bookID is the value that you input.

The following needs to be typed in a 'New Query Editor Window'.

To use sp_DeleteBook

Variables required

@bookID which is the ID of the book

What to type in order to execute the procedure

```
USE Suzi_Yoga_Studio
GO

EXEC sp_DeleteBook @bookID=<..'bookID'..>
GO
```

After execution the procedure will first check whether or not the record exists as part of the Exercises view, if it does not, the record where the bookID is of the value provided, else the record will not be deleted.

The following needs to be typed in a 'New Query Editor Window'.

To use sp_Report

Variables required

@class_code which is the code of the class you want check

What to type in order to execute the procedure

```
USE Suzi_Yoga_Studio
GO

EXEC sp_Report @class_code = <..'code_of_the_class'..>
GO
```

After execution the procedure will print a report containing details about the class, including members and their cell numbers that are part of that class.

Author Notes

Name : Tejas

Surname : Dwarkaram

Date : 2012/05/10

Project Notes

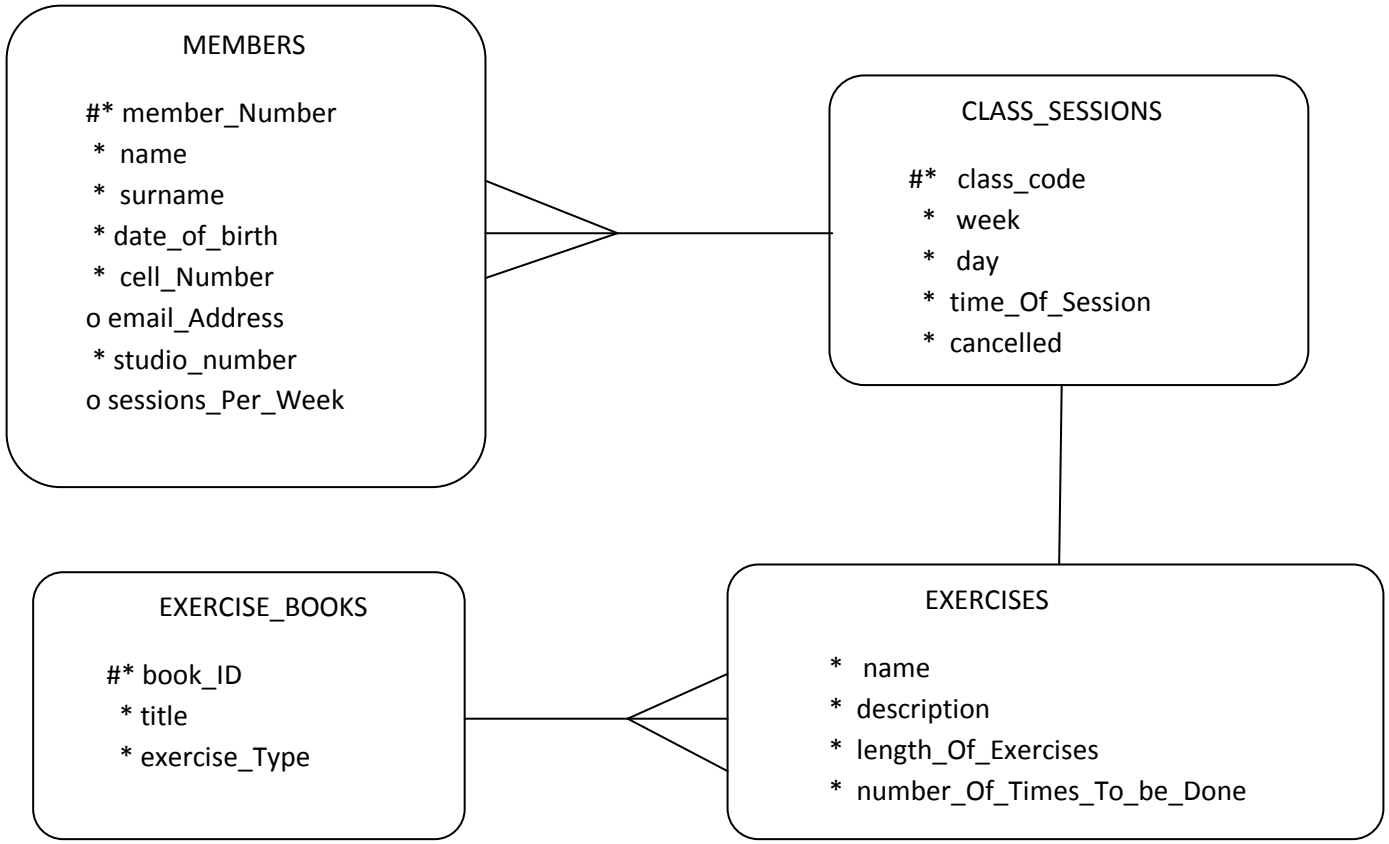
Purpose

The purpose of this project is to create a database as a student project. The database is created under scenario of a Yoga Studio.

Description

The database is for a Yoga Studio, named Suzi's Yoga Studio. This database will house information about the members, classes and exercises of Suzi's Yoga Studio. There are 4 tables that are utilized in this database. By making use of triggers, certain aspects of the tables are secured. Views have been created to cater for different criteria's. Stored procedures have been designed to carry out certain tasks easier, like adding records to the "Exercises" table.

ER Diagram


MEMBERS
#* member_Number
* name
* surname
* date_of_birth
* cell_Number
o email_Address
* studio_number
o sessions_Per_Week
CLASS_SESSIONS
#* class_code
* week
* day
* time_Of_Session
* cancelled
EXERCISE_BOOKS
#* book_ID
* title
* exercise_Type
EXERCISES
* name
* description
* length_Of_Exercises
* number_Of_Times_To_be_Done