

COMP2001 : Information Management & Retrieval

Database Principles: Workbook 1

CONTENTS

| | |
|--|---|
| CONNECTION TO REMOTE DATABASE..... | 1 |
| CREATE TABLE SQL STATEMENTS..... | 2 |
| APPLY CREATE TABLE STATEMENTS TO RDBMS SYSTEMS | 4 |
| ALTERING AND MANIPULATING YOUR DATABASE..... | 4 |

INTRODUCTION

The student tasks for this unit are to:

- Connect to your database on a remote server environment
- Practice create table statements for a number of different scenarios
- Apply the create table statements written above to different RDBMS systems and analyse the differences
- Practice being able to alter and manipulate your database.

For the purposes of these exercises you are to use Azure Data Studio as the tool for connecting and manipulating your Microsoft SQL Server remote database instance. This is just one tool of a number that can be used.

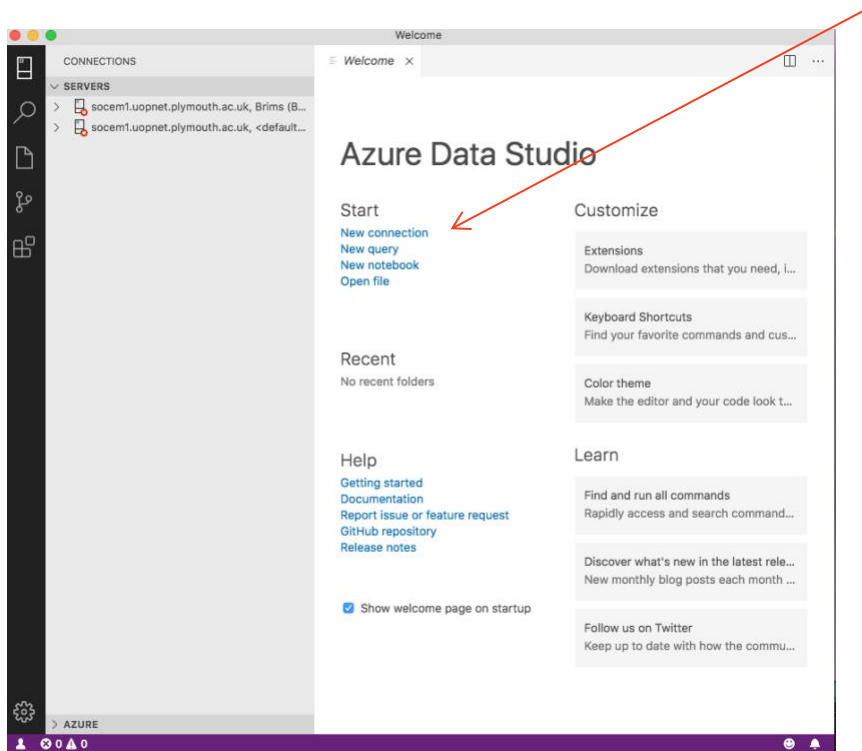
Connection to remote database

For the purposes of this module only you have been allocated a database on the University instance of Microsoft SQL Server. This database is held on the server socem1.uopnet.plymouth.ac.uk. Your database log in details have been emailed to you separately:

NOTE: The database should only be used for COMP2001 learning purposes. This database is not provided to you for the rest of your studies and will only be active until the module has been completed.

Activity 1

Set up your connection in Azure Data Studio by adding a New Connection.



Complete the boxes in the following screen as right:

Server = **socem1.uopnet.plymouth.ac.uk**

Username = Your username – see email

Password = Your password – see email

Database = Enter your database name as email – *do not use the drop down box as it will only show “master” and “tempdb” you do NOT want either of those.*

The screenshot shows the 'Connection Details' dialog box. It has two tabs: 'Recent Connections' and 'Saved Connections'. Under 'Recent Connections', it says 'No recent connection'. Below this is a form with the following fields: 'Connection type' (Microsoft SQL Server), 'Server' (socem1.uopnet.plymouth.ac.uk), 'Authentication type' (SQL Login), 'User name' (SATkinson), 'Password' (masked with asterisks), 'Database' (ISAD251_SATkinson), 'Server group' (<Do not save>), and 'Name (optional)'. There are 'Advanced...', 'Connect', and 'Cancel' buttons at the bottom.

On successful completion of this exercise you should now be looking at a Database Dashboard. There is nothing in the database yet to look at and so the next exercises should take you through addressing this.

CREATE TABLE SQL STATEMENTS

The lecture session and materials have taken you through how to identify entities and relationships. From there you need to learn how to create tables using SQL statements rather than an interface. These exercises are designed to help you develop this skill. The recommended reading is Chapter 8 in the Database Systems book.

You can find information about how to use Azure Data Studio here

<https://www.mssqltips.com/sqlservertip/6029/azure-data-studio-step-by-step-tutorial/>

Activity 2

Download the sql file accompanying this workbook (*2.3 DB Workbook 1.sql*). Using Azure Data Studio, and the connection to your Microsoft SQL database given above, run the given SQL script. Using the side bar in Azure Studio view the tables, the columns and the constraints. Are they what you would expect?

Annotate the sql file so that you label important concepts that are demonstrated there. Use the checklist at the side to help you consider what is important to note.

- ✓ What does “Check” do?
- ✓ How would you use it?

Modify the .sql file so that you now have a foreign key constraint added to the database. Run this SQL and see the results in the database.

Activity 3

For each of the scenarios below identify the **entities** and the **attributes** of those entities. You are to write a collection of create table statements that will form the basis of three separate database applications. Consider here how you would enforce integrity.

NOTE: You only have one allocated database, therefore whilst you are contemplating three separate database applications, you will need to have all your tables in the same database. I recommend you consider how to differentiate the table names.

Scenario 1 is where a medical trial wants to carry out an analysis on the number of times in a day that a Multiple Sclerosis patient falls over during a given period of time and the injuries that result from the falls. The trial will last for approximately 12 weeks and a carefully selected number of patients will be issued with a mobile app that allows them to enter for each day the following information:

The number of times they have fallen

The type of injury (if any) the patient received that day.

The injury types are constrained by a set list – the app will not allow free-text entry of the type of injury received.

Scenario 2 is where the placements information is stored for all students going on placement. Information about the student is stored in terms of their student ID, Programme ID, first and last names as well as their stage of study (eg 2nd year). Information about the placement provider is stored such as the company name, address, postcode, contact name at the company and position of the contact (eg: Jo Bloggs, Lead Developer). Additionally, the start and end date for the student placements are stored along with the name of their allocated placement tutor.

Scenario 3 is for a web-based application for booking into a volunteer hostel. The hostel has a number of rooms which have a number of beds. The application needs to hold data about the dates that volunteers have booked for, who the volunteers are, when the volunteers will leave and their gender. Volunteers are allocated to rooms according to gender.

Apply create table statements to RDBMS systems

The purpose of this activity is for you to explore and experience the differences between the major RDBMS systems that you are likely to encounter. Microsoft SQL Server is a substantial enterprise grade database common in industry. MySQL belongs to Oracle and is a good open source alternative. However, they have their differences and quirks. This exercise is for you to experience and debug for yourself the differences.

Activity 4

Exploring Cloud: Using your Azure account explore how you might deploy into an Azure database. What are the steps you need to take?
How easy or hard is it?

Exploring Open Source: Carry out some research on how you might translate your current SQL statements created for Microsoft SQL Server into another RDBMS system such as MySQL. You can use the PHPMyAdmin interface on the Proj-MySQL server,

`proj-mysql.uopnet.plymouth.ac.uk/index.php`

What do you need to change?
Why?

ALTERING AND MANIPULATING YOUR DATABASE

Having created your database and carefully crafted the tables, you might review it and realise that you have missed something. Rather than delete the whole lot and start again you need to be able to alter and adapt. The following activities should be carried out using SQL statements. Refer to Chapter 8 in the recommended book (Database Systems) for information on how to do these.

Activity 5

1. Return to the original Orders and OrderDetails tables from Activity 2. In the Orders table, amend the column Customer so that it is CustomerID and is an integer.
2. Add a table called Customers that has a field called CustomerID that is the primary key.
3. Add a foreign key constraint so that Orders CustomerID is linked to the new Customers CustomerID
Primary key
4. Add a new column to the OrderDetails table that is the UnitPrice. This should be a suitable data type for a monetary value.

Useful Links

Conventions used in SQL <https://docs.microsoft.com/en-us/sql/t-sql/language-elements/transact-sql-syntax-conventions-transact-sql?view=sql-server-2017>