**DBAssist Documentation**

Crated by Adi Ronen for GGFBM, Ben-Gurion University of the Negev, 2018.

# The purpose of the program

The program allows students with a click of a button to open a database on a Microsoft SQL server 2016. The student should choose a course and type of database (personal or group) and DBAssist creates the database and gives privileges to the lecturers and practitioners of that course based on the DBAssistSettings.xml settings page that is located on the server.

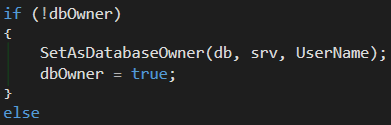
# DBAssist User

For the program to be able to open databases and to give permissions, it needs a user on the server which has the minimum permissions to do the actions the system needs. Open a new user named DBAssist and give it the following privileges:

**Securityadmin**- manage logins and their properties. Can GRANT, DENY, and REVOKE server-level permissions. Can also GRANT, DENY, and REVOKE database-level permissions if have access to a database. Additionally, can reset passwords for SQL Server logins. **Dbcreator**- can create, alter, drop, and restore any database. **Public**- Every SQL Server login belongs to the public server role. When a server principal has not been granted or denied specific permissions on a securable object, the user inherits the permissions granted to public on that object. Only assign public permissions on any object when you want the object to be available to all users. You cannot change membership in public.  
*Note: public is implemented differently than other roles, and permissions can be granted, denied, or revoked from the public fixed server roles.*

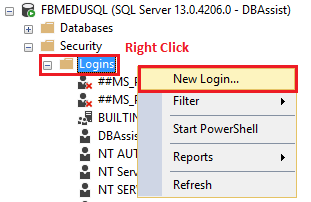
## Note! If you wish to give the user privileges to preform **backups** and to **restore** databases from backups, DBAssist user should have “**sysadmin**” server roles. Also changes in the code should be performed as followed in **Model**.**cs** file:

1) Under “*SetPersonalPermissions*” function, comment out *“UserPermissions****”*** and uncomment “*SetAsDatabaseOwner*” function.

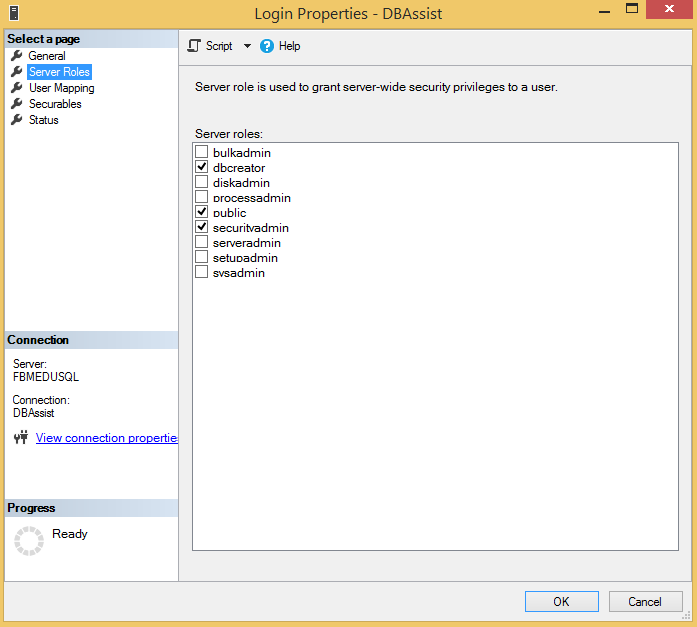
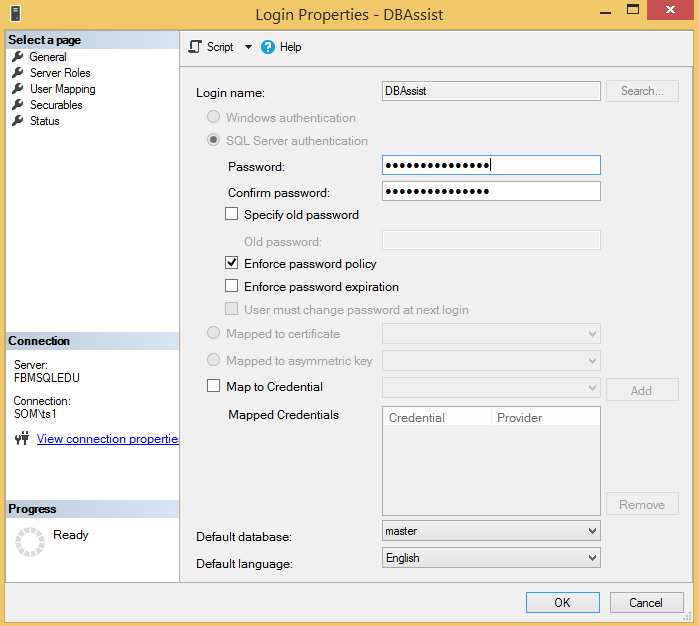
2) Under “SetGroupPermissions” function, uncomment the inner if else statement:   
 

### How to create DBAssist user

Open SQL Server Management Studio with an administrator account such as SA 🡪under **Security Right Click** on **Logins 🡪 New Login...**



Fill up Login – New as follows:



# Files needed

## Settings.settings

In the DBAssist.sln under Properties, there is Settings.settings File with the following settings:

SettingLocation (string) – in this folder “DBAssistSettings.xml” is placed.

SQLServerName (string) – The default SQL server name.

ConnectorName (string) – the user name of the user that have the needed permissions. In this case is DBAssist user from above.

ConnectorPassword (string) – The user’s password.

Note, if you wish to change one of the above you will have to compile DBAssist project with the new Settings.

## DBAssistSettings.xml

To allow centralized data management, this file should be placed on the server in a shared folder.  
 For example -   
Path: D:\SQL DBAssist Settings\DBAssistSettings.xml  
Share: \\XXXXXX\SQL DBAssist Setting$  
Comments: Provide Read access permission for the DBAssist application to the configuration XML-file.

This xml file contains all the information the program need about the course that need SQL databases and their students and faculty staff.

The xml tree is built as follows:

<**DBAssist**> head of the tree

<**Year** *Value*=”20XX”> the year of the courses

<**Semester** *Value*=”X”> the semester of the courses (A/B/C)

<**Course** *Number*="XXX.X.XXX" *Name*="XXXXXXX" *UseCourseNumberInDBName*="True/False">

(\**Use Course Number in DB Name*- if to include the course number in the database name only True or False value. This value should be true if there is more than one course in the same year and semester with overlapping students)

<**Faculty**> the faculty members of this course (lecturers and staff) each name in separated line.

XXXX

XXXX

XXXX

</**Faculty**>

<**Students**> the students of this course that can open a personal database each name in separated line. (If there are no personal dbs in this course, leave this tag empty).

XXXX

XXXX

XXXX

</**Students**>

< **Groups**> under groups are all the groups in this course that opens a database together. (If there are no group dbs in this course, leave this tag empty).

<**Group** *Name*="XXXX"> team name in the tag and each name of participant in separated line.

XXXX

XXXX

</**Group**>

<**Group** *Name*="XXXX">

XXXX

XXXX

</**Group**>

…

…

</ **Groups**>

</**Course**>

<**Course** *Number*="XXX.X.XXX" *Name*="XXXXXXX" *UseCourseNumberInDBName*="True/False">

…

…

</**Course**>

</**Semester**>

<**Semester** *Value*=””> the semester of the courses (A/B/C)

…

…

</**Semester**>

…

…

</**Year**>

<**Year** *Value*=””>

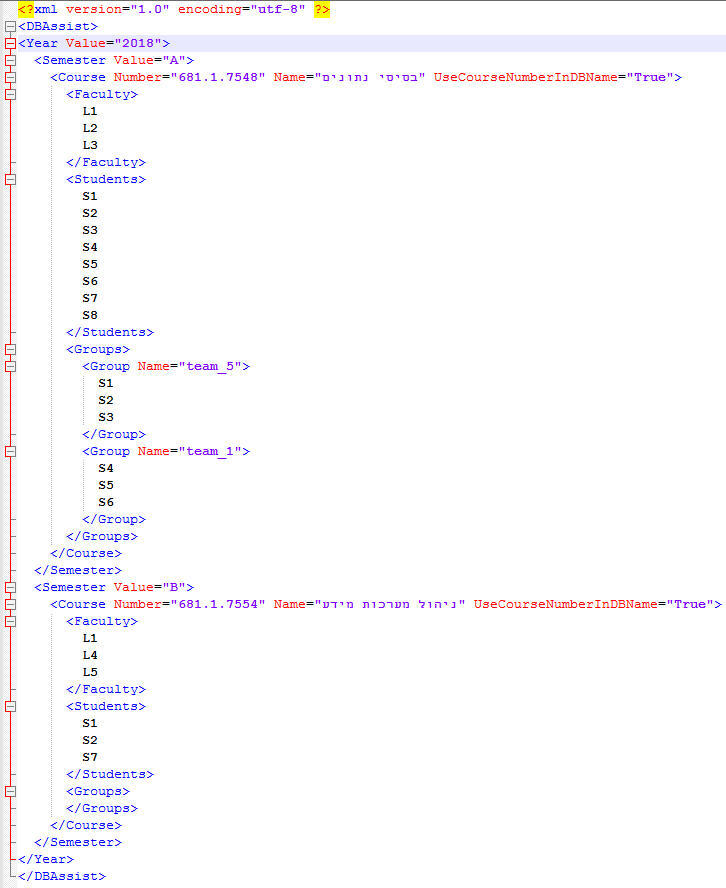
…

…

</**Year**>

</**DBAssist**>

EXAMPLE:



# Possible errors and possible solutions

* The server may be disconnected
* There is a possibility that the firewall is working and this does not allow communication. Close the firewall or open the relevant ports.
* The username and password may not be correct in the settings. Check whether the system is properly configured and compiled.
* It is possible that the configuration file is not found, check that it exists and if it exists, check its name as in the above settings and it is in the same folder of the run file or the path it is in is specified in system settings and compiled as above. It may also be that the data in the file is not properly organized, check with the template above.