SQL Server Unit Testing

Using

Visual Studio 2017

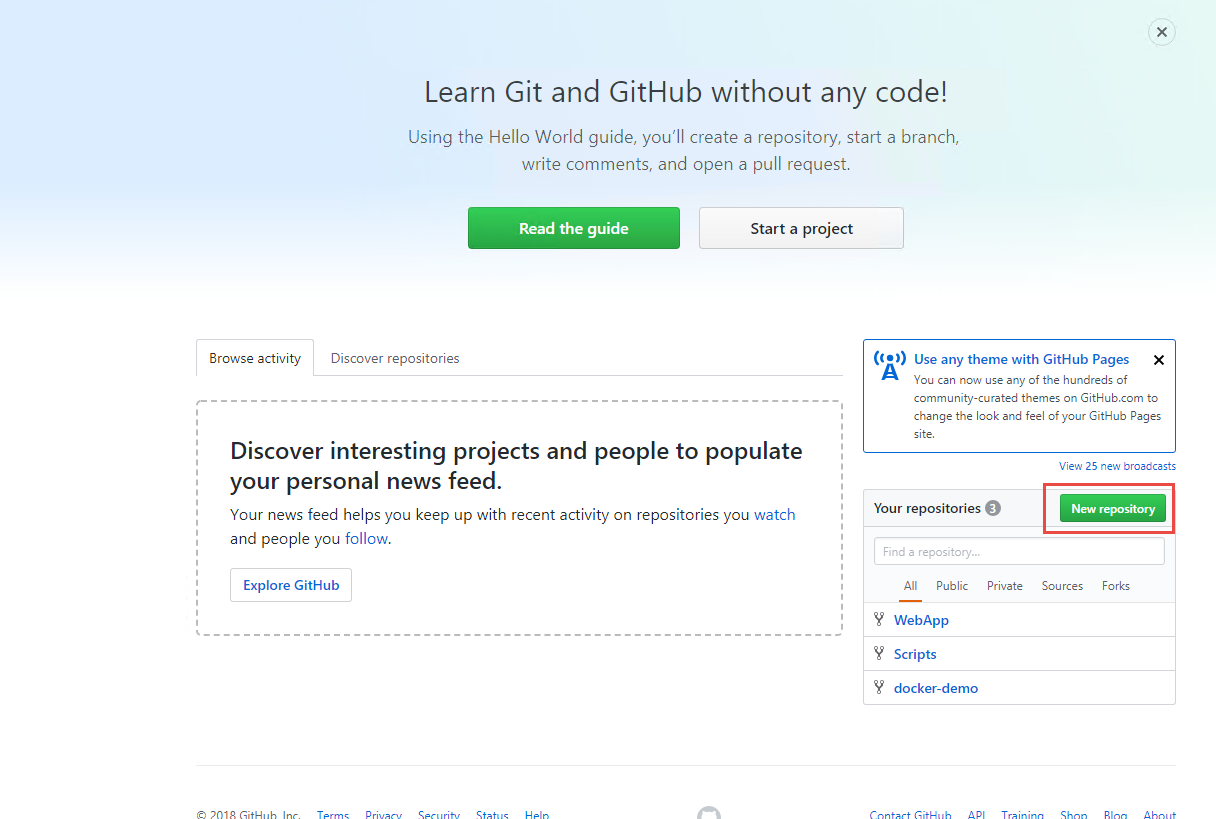
Index

# Content Page No

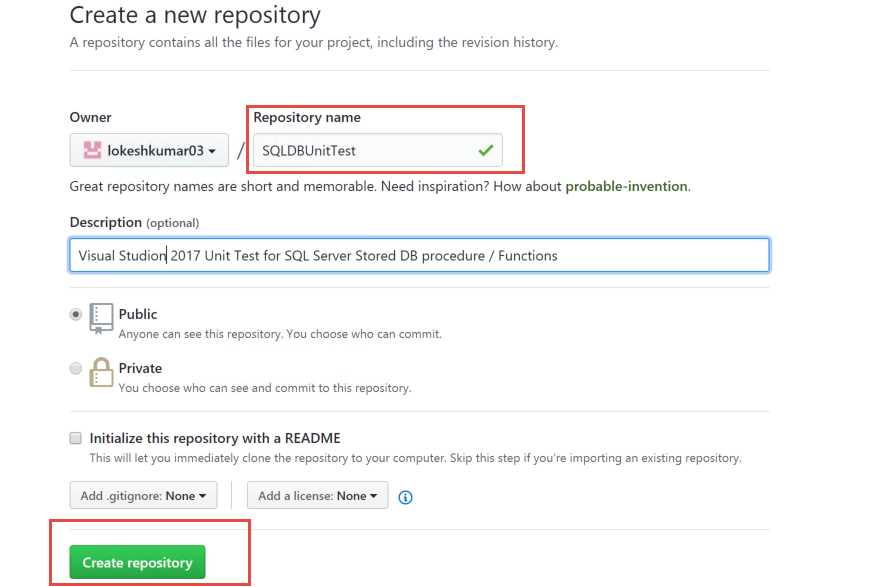
1. Create GitHub Repository [3](#_Create_GitHub_Repository)
2. Database Scripts [5](#_Database_Script)
3. Create SQL Server Database Project [6](#_Create_SQL_Server)
4. Writing Unit Test [8](#_Writing_Unit_Test)
5. Execution of Unit Tests [12](#_Execution_of_Unit)
6. Commit Changes to GitHub [13](#_Commit_Changes_to)

# Create GitHub Repository

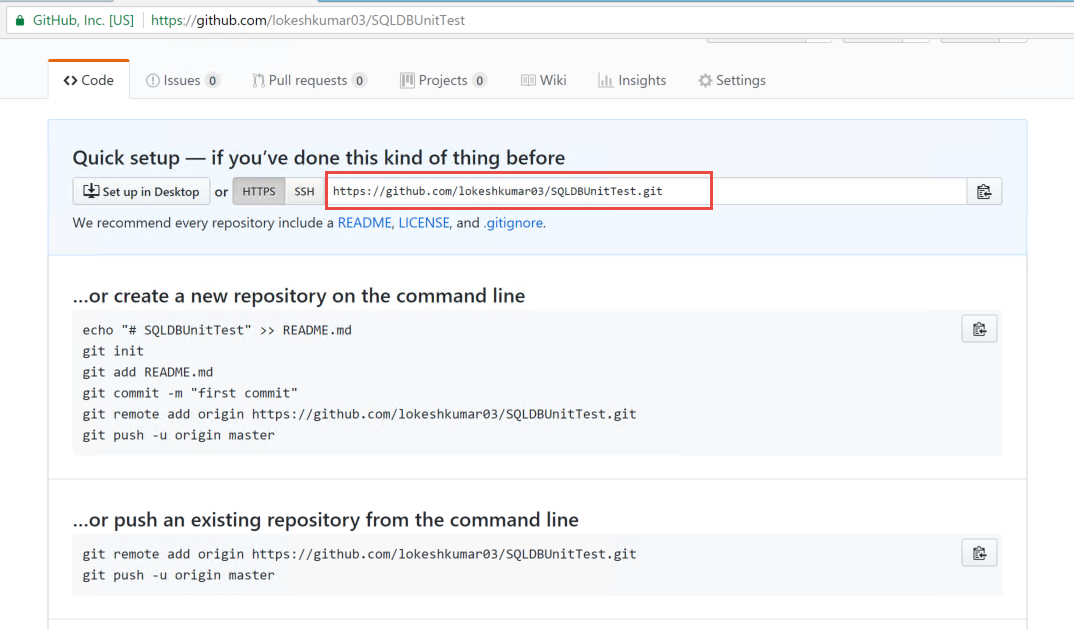
1. Login to Github
2. Click “New repository” button which will redirect to new page.



1. Enter Repository name and click Create repository button.



It will redirect to new page with SSH URL of the new repository created.



# Database Script

**Scripts to create tables and stored procedure:**

USE [TestSQLDB]

GO

/\*\*\*\*\*\* Object: Table [dbo].[User] Script Date: 2/23/2018 8:16:26 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[User](

[Id] [uniqueidentifier] NOT NULL,

[Name] [nvarchar](50) NULL,

[Email] [nvarchar](50) NULL

) ON [PRIMARY]

GO

CREATE PROCEDURE [dbo].[InsertUser]

(

@id uniqueidentifier,

@name VARCHAR(30),

@email VARCHAR(30)

)

AS

BEGIN

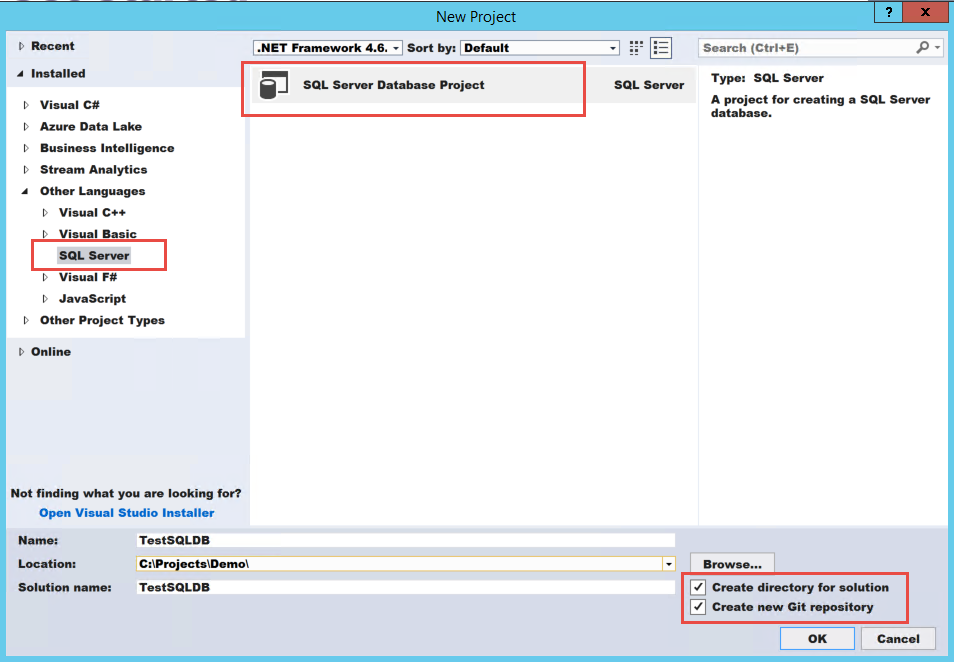
INSERT INTO dbo.[User] (Id,Name,Email) VALUES( @id, @name, @email)

END

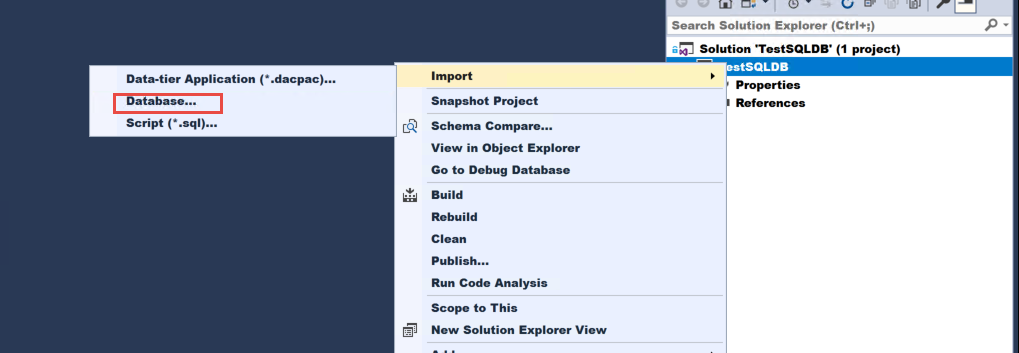
# Create SQL Server Database Project

We will create SQL Server Database project and import the schema from existing database.

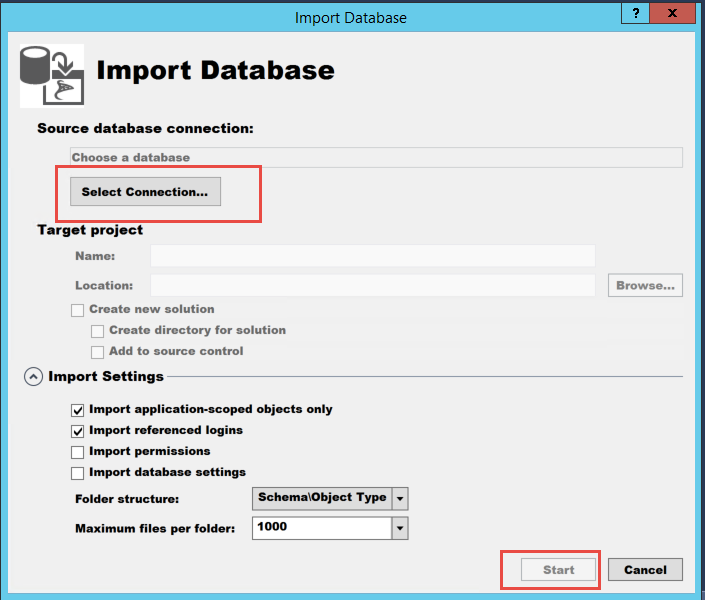
1. Create SQL server project. Provide name ,Location and Solution name
2. Select check box for create directory for solution
3. Create new Git repository (if you require to create which is optional)



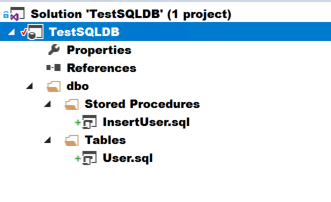
1. Right click project and click on Import and select “Database...”



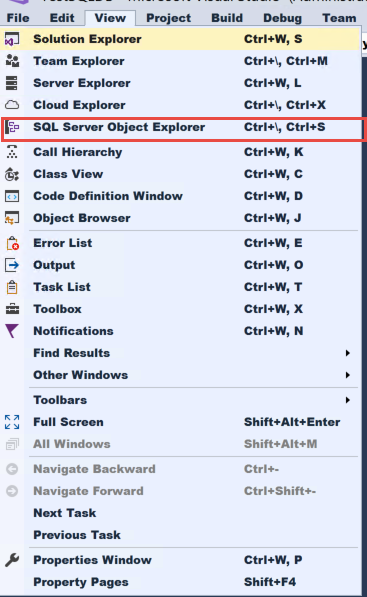
1. Click on “Select Connection” and provide the connection of SQL server database and click on start. Once the import is done click finish.



**Project structure of SQL Database:**



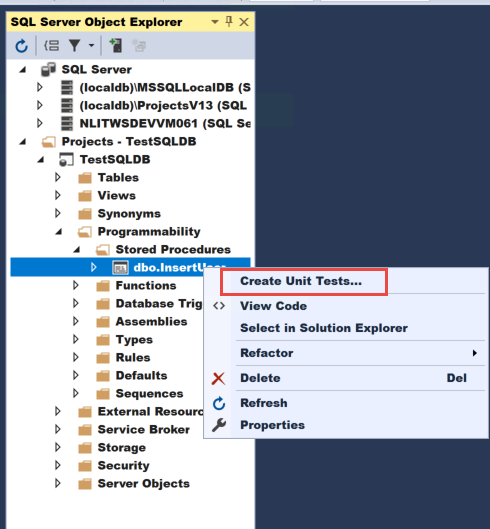
1. Open SQL Server Object Explorer



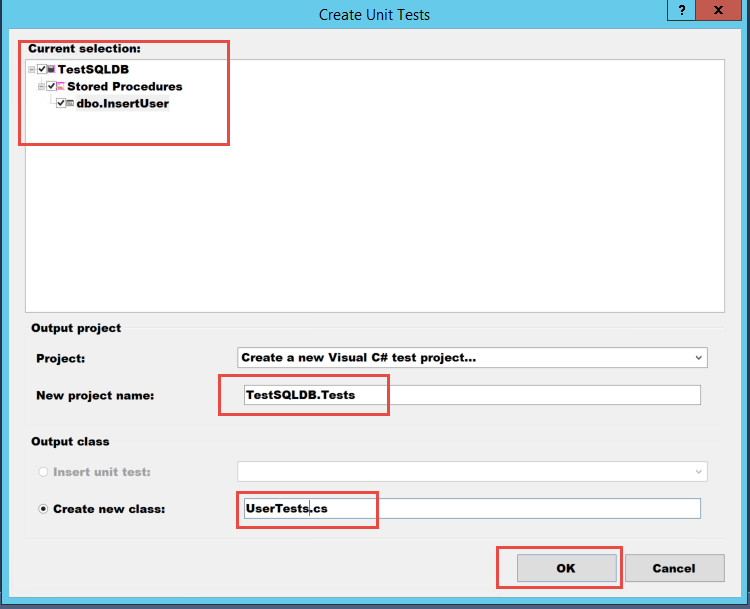
# Writing Unit Test

We can create unit test for each stored procedures and functions.

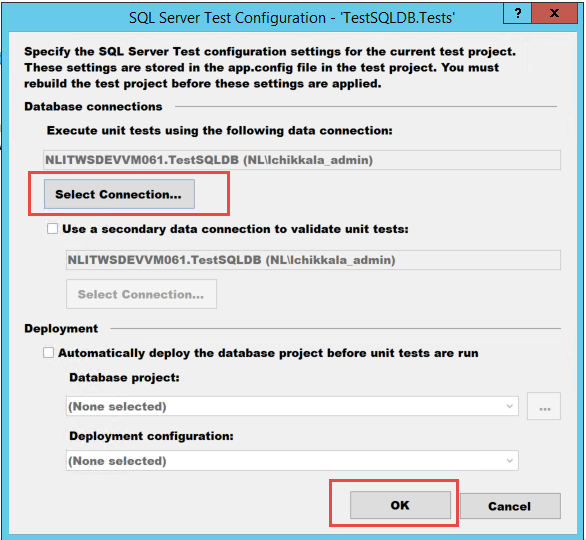
1. Right click on stored procedure and click on “Create Unit tests”



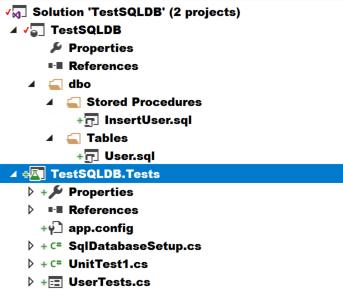
1. Select stored procedure for which a unit test is expected to be create.
2. Provide “New project name:” and click “OK” to proceed.



1. It will prompt you for the configuration. Click “Select Connection” and provide the connection and then click ok. It will create a unit test project.



**Project structure of Unit test:**

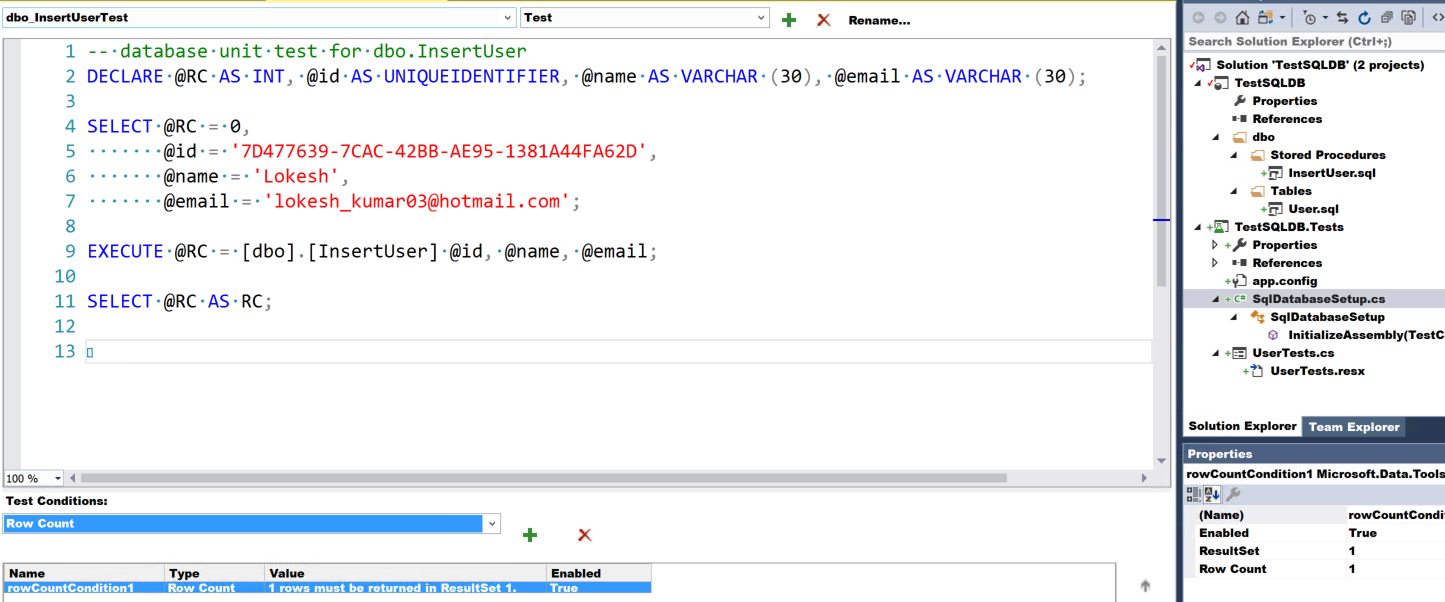


**Design mode of unit test:**

We have three type of design mode

1. Pre-test
2. Test
3. Post-test

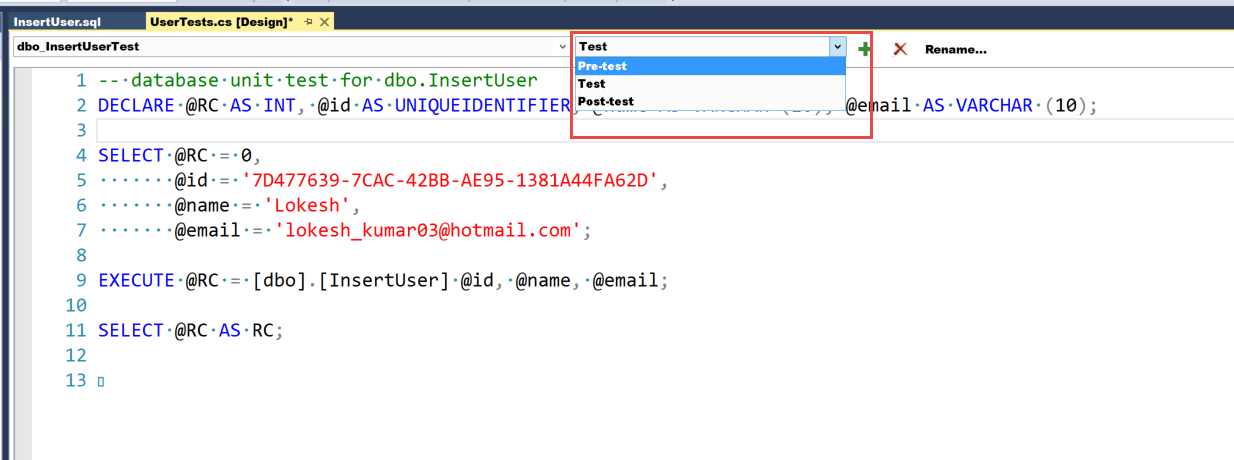
**Test:** To execute the stored procedure.



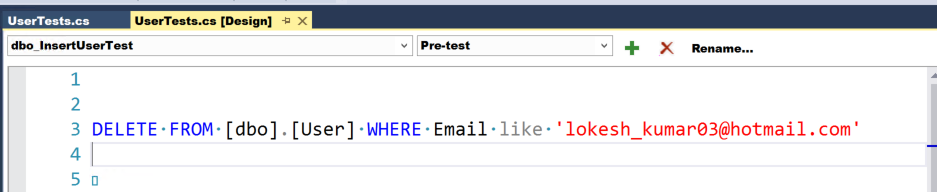
**Pre-test:** To execute script before “Test”.

Make sure record doesn’t exist before we execute “Test”. Here we create Pre – test to delete the record and then we execute test.

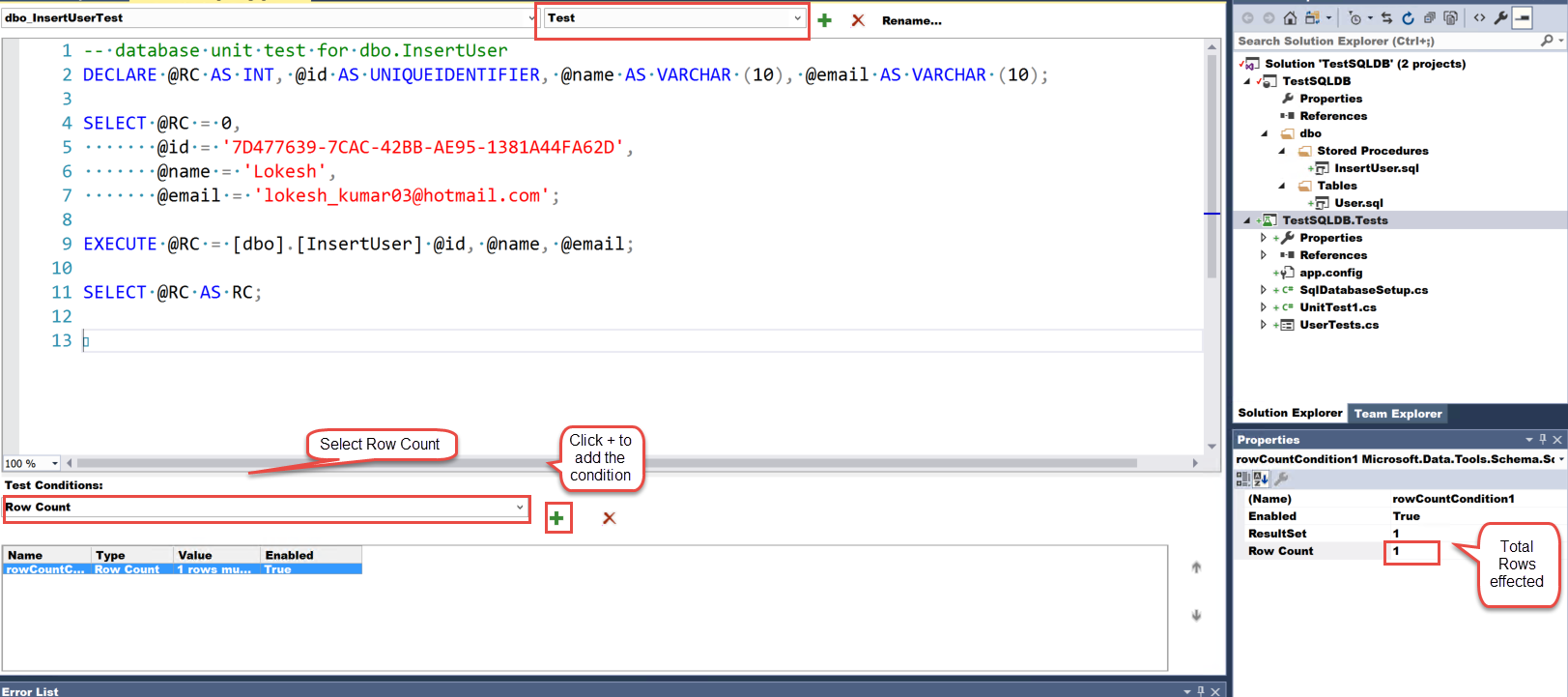
1. Select Pre-test from the drop down.



1. Add delete script to delete the record .This will execute before the test script is executed.

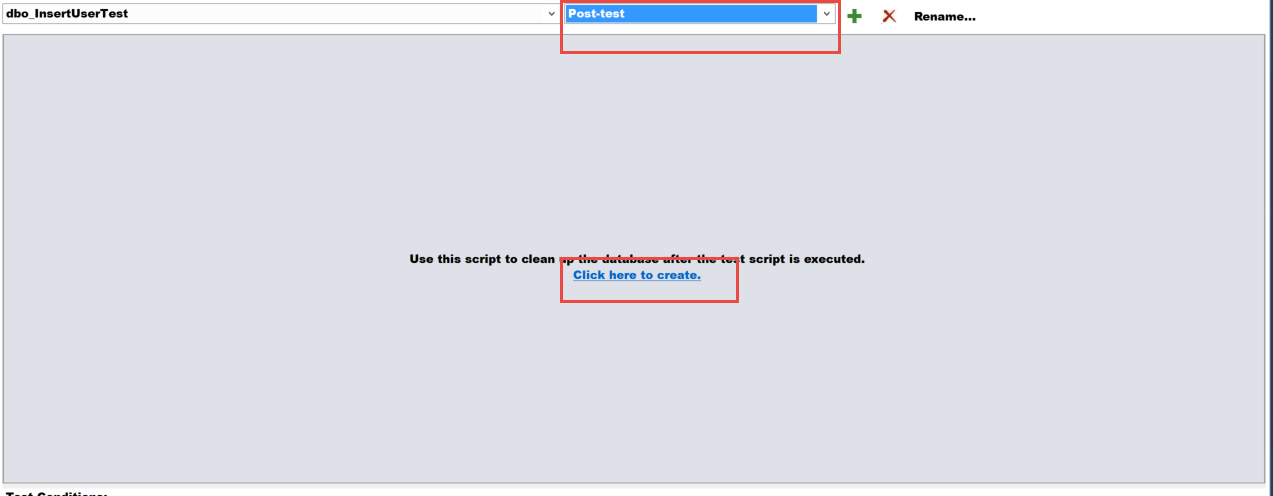


1. Go to Test mode. Provide the values for the script.
2. Delete the default Test condition (select the default row and click X )
3. Select Row Count from drop down of “Test Conditions:” and then click “+”.
4. Modify the Row count from the properties. Which is the expected value after deletion. Here stored procedure returns 1.

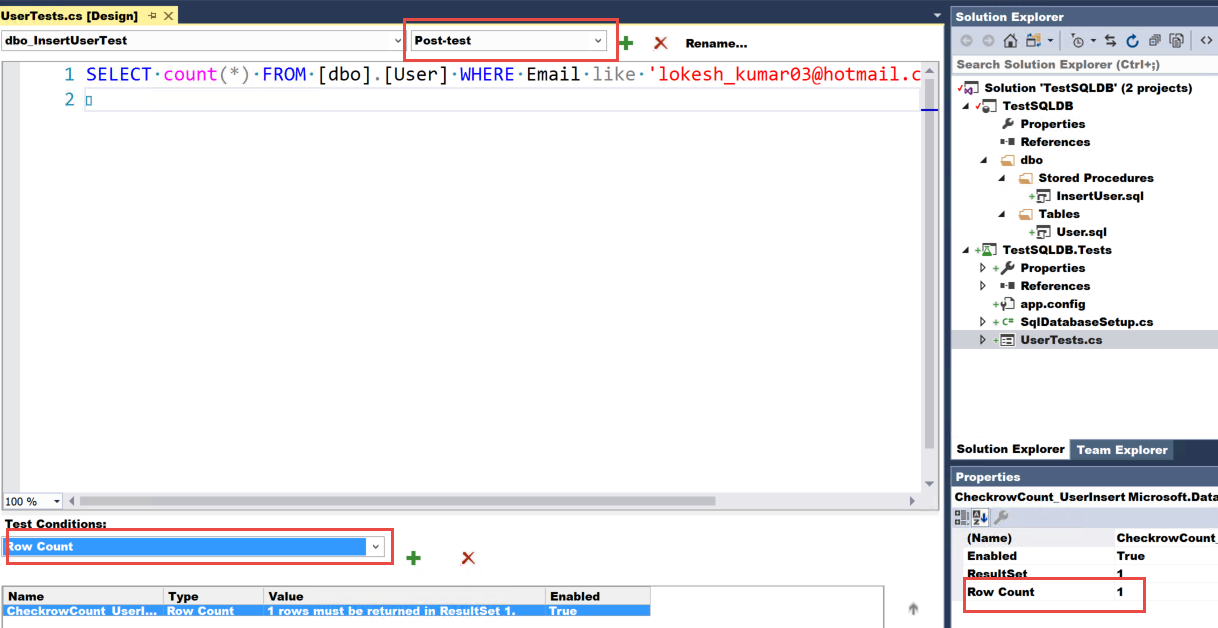


**Post-test:** To execute script after “Test”.

1. Select Post-test from the drop down. To create a script click on “Click here to create”



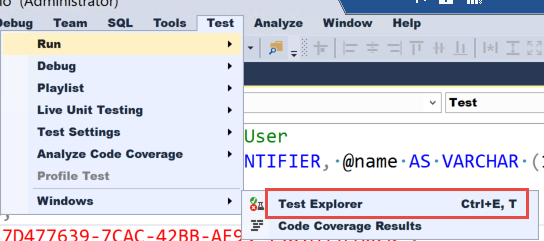
1. Add the select script.
2. Select Row Count from drop down of “Test Conditions:” and then click “+”.
3. Modify the Row count from the properties. Which is the expected value after deletion. Here stored procedure returns 1.

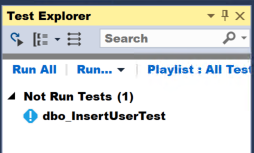


# Execution of Unit Tests

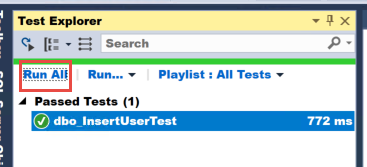
**Test Explorer:**

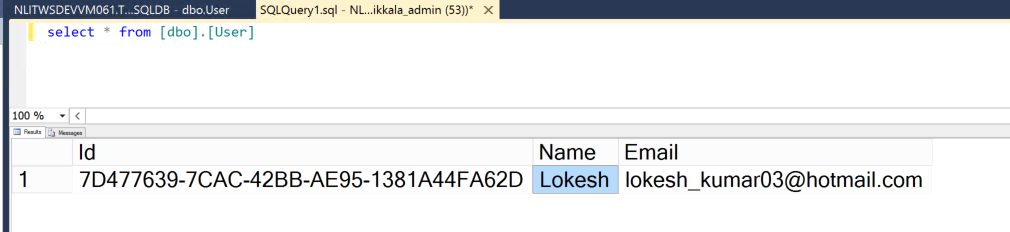
Go To Menu. Select windows under Test menu and then select Test Explorer.





Click Run All

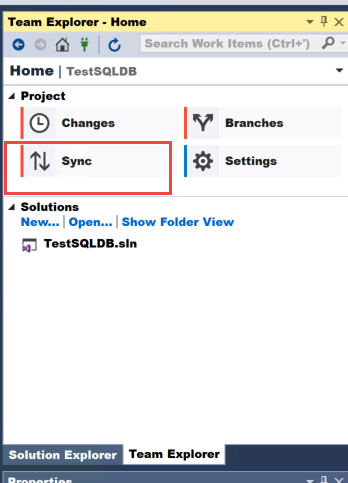




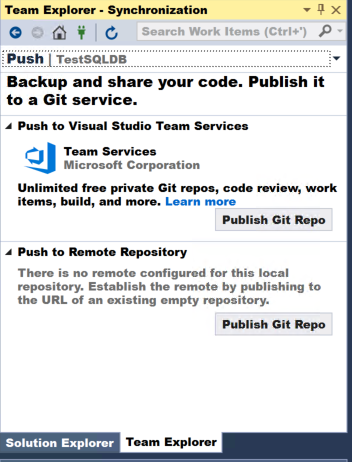
# Commit Changes to GitHub

Push your changes to Git Repo using Visual studio.

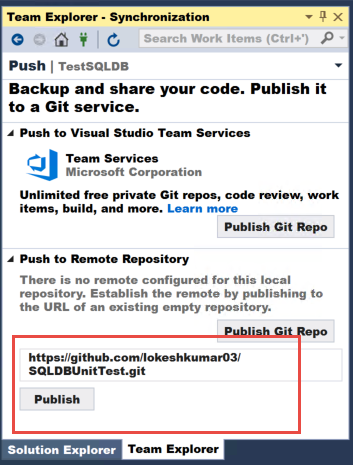
1. Click on Home on the “Team Explorer”
2. Click on “Sync”



1. As you have already created repository. Copy the URL from SSH of GitHub and add it to the “Publish Git Repo”



1. Click Publish.



Delete a GitHub Repository:

1. Go to Settings of an existing Repository.
2. Go to end of the page for Danger Zone.
3. Click Delete this repository.
4. You will get a prompt to type the repository name. Enter the repository name to confirm.
5. Click the button “I understand the consequences, delete this repository”

