MS-SQL Integration

# In Unreal Engine 4

User Manual

MS-SQL Integration is a plugin that lets you connect to your SQL Server database, and store and retrieve data from the server via SQL queries, directly from the Blueprint. This plugin brings the power of C#.NET in Unreal Engine 4.

If you have worked with MS SQL Server, you must be very familiar with writing Select, Update, Insert and Delete SQL queries, using joins and sub-queries. If you know C#.Net, and worked with ADO.Net, you must be knowing how to set Connection properties for your SQL Server, and how to write and execute your SQL Queries within your .NET application.

But now you can use Blueprints in Unreal Engine 4 to do your job. You are still writing the same queries that you are familiar with, but here you are writing it within Blueprints, which gives you flexibility to store and retrieve data as well as Images from your SQL Server to your UE4 game.

If you are not familiar with SQL Server at all, you will get plenty of online resources to get you started, and you can definitely test it out by downloading SQL Server Express.

SQL Server Express is an entry level database server having a limitation of 10 GB size, which is perfect from small scale applications.

Here I will be explaining you in details how to set up your database connection and how to write and execute SQL queries from UE4.

*Working with SQL Server Express*

**If you already have your database and tables ready to be integrated with your UE4 project, you may skip this section*.***

Assuming that you have your database server up and running, either locally or remotely, to get started working with this plugin, you first need to create a database and a table in your server, where you would like to store all your in-game data.

If you are unsure regarding how to work with SQL Express, let me quickly walk you through the basic steps.

1>  Download and install SQL Server Express edition from the below link:

<https://go.microsoft.com/fwlink/?LinkID=799012>

2>  Open your SQL Server client and connect to your database server by using Windows Authentication or SQL Server authentication.

 By default, you will get a local DB server called **locahost**, which will be saved in your local machine. To create a dedicated server, please go through the steps given in the below link:

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-get-started>

To understand how to create a login, please visit the below link:

<https://msdn.microsoft.com/en-in/library/aa337562.aspx>

3>Once you are connected to the server, you need to create a new database. Visit the below link to help you out with the database creation:

<https://msdn.microsoft.com/en-us/library/ms186312.aspx>

4>Once your database is created, it should be visible in object explorer, within **Databases.**

Right click on your database name, and open New Query Window. Then create one or more tables, which you want to use for storing and retrieving your game data. Below is a simple query that lets you create a table called **Products**. (Source - <https://msdn.microsoft.com/en-us/library/ms365315.aspx>)

CREATE TABLE dbo.Products

   (

    ProductID int PRIMARY KEY NOT NULL,

    ProductName varchar(25) NOT NULL,

    Price money NULL,

    ProductDescription text NULL

   )

GO

*Setting Up Database Connection in UE4*

Before we begin writing our SQL queries and executing them in our database, we first need to set up a connection within UE4. This is a onetime setup, and is similar to how we manually connect to our DB Server using Windows or SQL Server Authentication, as shown in the above steps.