

C# ТІЛІНДЕ ДЕРЕКТЕР БАЗАСЫМЕН ЖҰМЫС ІСТЕУ ТЕХНОЛОГИЯЛАРЫ

ADO.NET-ке кіріспе

Бүгінгі таңда деректермен жұмыс істеудің маңызы зор. Деректерді сақтау үшін әртүрлі дерекқорды басқару жүйелері қолданылады: MS SQL Server, Oracle, MySQL және т.б. Көптеген ірі қосымшалар деректерді сақтау үшін осы дерекқорды басқару жүйелерін пайдаланады. Алайда, мәліметтер базасы мен C# қосымшасы арасында байланыс орнату үшін делдал қажет. ADO.NET технологиясы дәл осындай делдал болып табылады.

ADO.NET - бұл NET.Framework платформасына негізделген деректермен жұмыс істеу технологиясы. Бұл технология бізге мәліметтер базасына сұраныстар жіберуге, қосылымдар орнатуға, мәліметтер базасынан жауап алуға және басқа да бірқатар операцияларды жасауға болатын класстар жиынтығын ұсынады.

Сонымен қатар, мәліметтер базасын басқару жүйелері көп болуы мүмкін екенін атап өткен жөн. Олардың мәні бойынша олар әртүрлі болуы мүмкін. MS SQL Server, мысалы, сұраныстарды құру үшін T-SQL тілін қолданады, ал MySQL және Oracle PL-SQL тілін қолданады. Әр түрлі дерекқор жүйелерінде әр түрлі мәліметтер болуы мүмкін. Алайда функционалдылық ADO.NET әзірлеушілерге әр түрлі ДҚБЖ-мен жұмыс істеу үшін бірыңғай интерфейсті ұсыну үшін салынған.

ADO.NET-те мәліметтер базасымен өзара әрекеттесу интерфейсінің негізі нысандардың шектеулі шеңберін ұсынады: *Connection*, *Command*, *DataReader*, *DataSet* және *DataAdapter*. **Connection** нысанын қолдана отырып, деректер көзіне қосылу орнатылады. **Command** нысаны дерекқордағы деректермен операцияларды орындауға мүмкіндік береді. **DataReader** нысаны сұрау нәтижесінде алынған деректерді оқиды. **DataSet** нысаны деректер базасынан деректерді сақтауға арналған және мәліметтер базасынан тәуелсіз жұмыс істеуге мүмкіндік береді. Ал **DataAdapter** нысаны DataSet пен деректер көзі арасындағы делдал болып табылады. Негізінен, деректер базасымен жұмыс осы нысандар арқылы жүреді.

Дегенмен, әртүрлі деректер көздері үшін бірдей нысандар жиынтығын пайдалану үшін тиісті деректер провайдері қажет. Іс жүзінде деректер провайдері арқылы ADO.NET-те деректер базасымен өзара әрекеттесу жүзеге асырылады. Және әрбір деректер көзі үшін ADO.NET-те жоғарыда аталған класстардың нақты іске асырылуын анықтайтын провайдер болуы мүмкін.

Әдепкі бойынша ADO.NET-те келесі ендірілген провайдерлер бар:

- MS SQL Server үшін провайдер.
- OLE DB үшін провайдері (MS SQL Server-дің кейбір ескі нұсқаларына, сондай-ақ Access МҚ, DB2, MySQL және Oracle дерекқорына қол жеткізуге мүмкіндік береді).
- ODBC үшін провайдер (провайдерлері жоқ деректер көздері үшін провайдер).
- Oracle үшін провайдер.
- EntityClient провайдері. ORM Entity Framework технологиясына арналған деректер провайдер.
- SQL Server Compact 4.0 серверіне арналған провайдер

Ендірілген осы провайдерлерден басқа, MySQL сияқты әртүрлі дерекқорларға арналған көптеген басқалары да бар.

ADO.NET-те пайдаланылатын негізгі аттар кеңістігі:

System.Data: ADO.NET архитектурасын жүзеге асыратын класстарды, интерфейстерді, делегаттарды анықтайды.

System.Data.Common: ADO.NET-тің барлық провайдерлеріне ортақ классқа ие.

System.Data.Design: өз деректер жиынтығын құру үшін қолданылатын класстарды анықтайды.

System.Data.Odbc: ODBC үшін деректер провайдерінің функционалдығын анықтайды.

System.Data.OleDb: OLE DB үшін деректер провайдерінің функционалдығын анықтайды.

System.Data.SqlClient: SQL Server функциясын қолдайтын класстарды сақтайды.

System.Data.OracleClient: Oracle дерекқоры үшін провайдердің функционалдығын анықтайды

System.Data.SqlClient: MS SQL Server дерекқоры үшін провайдердің функционалдығын анықтайды

System.Data.SqlServerCe: SQL Server Compact 4.0 үшін провайдердің функционалдығын анықтайды

System.Data.SqlTypes: MS SQL Server деректер түрлеріне арналған класстарды қамтиды

Microsoft.SqlServer.Server: SQL Server және CLR ортасының өзара әрекеттесуі үшін компоненттерді сақтайды

Схемалық түрде ADO.NET-ті келесі түрде ұсынуға болады:



ADO.NET функционалды класстарын екі деңгейге бөлуге болады: қосылған және ажыратылған. Әрбір .NET деректер провайдері қосылған деңгейді құрайтын Connection, Command, DataReader, DataAdapter және басқа да нысандардың өз нұсқаларын жүзеге асырады. Яғни, олардың көмегімен мәліметтер базасына қосылу орнатылады және онымен өзара әрекеттесу жүзеге асырылады. Әдетте, осы объектілерді әр нақты провайдер үшін жүзеге асыратын олардың атауында провайдерді көрсететін префикс болады:

Object	SQL Server	OLE DB	ODBC
Connection	SqlConnection	OleDbConnection	OdbcConnection
Command	SqlCommand	OleDbCommand	OdbcCommand
Data reader	SqlDataReader	OleDbDataReader	OdbcDataReader
Data adapter	SqlDataAdapter	OleDbDataAdapter	OdbcDataAdapter

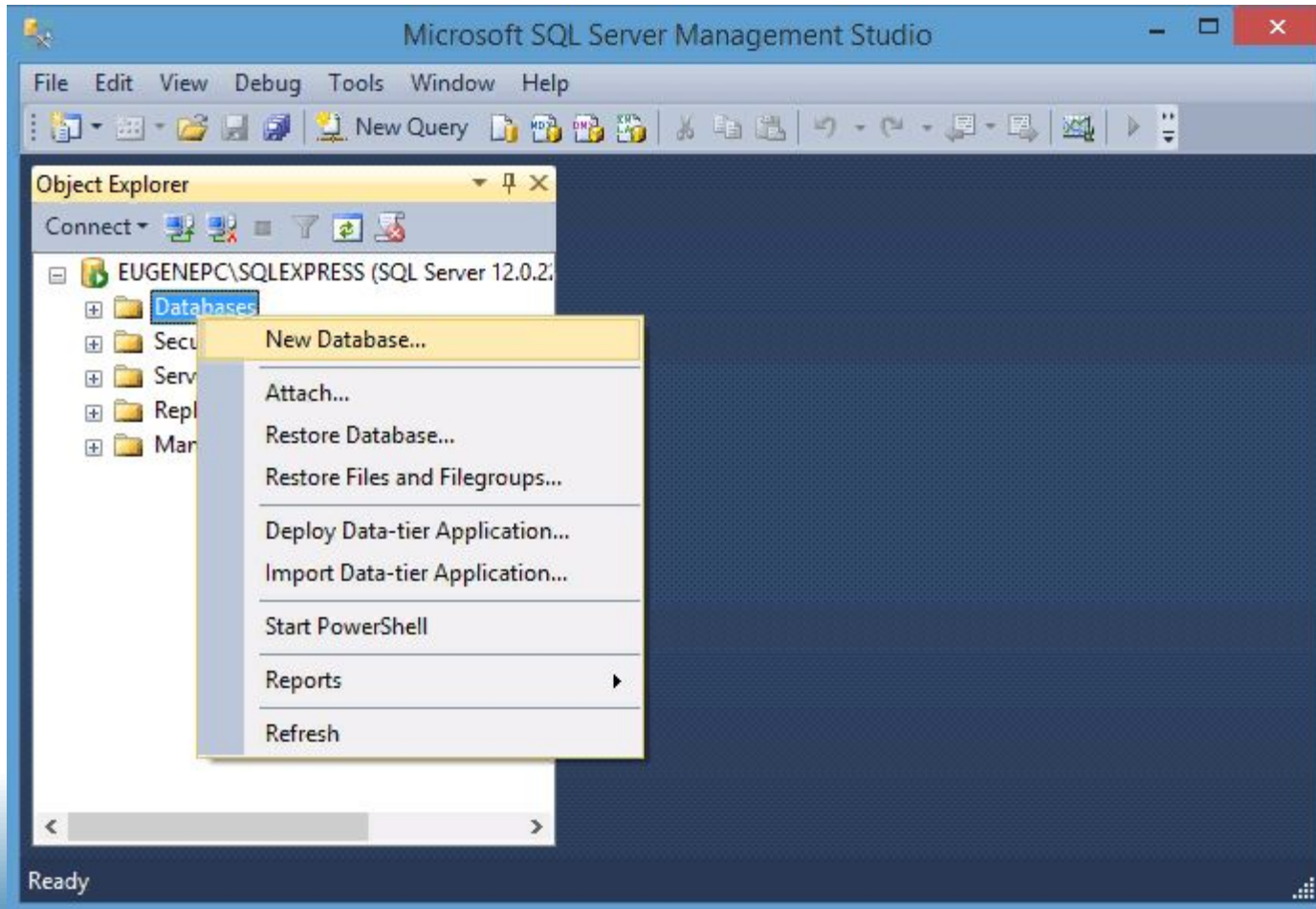
DataSet, DataTable, DataRow, DataColumn және басқалары сияқты басқа класстар өшірілген деңгейді құрайды, өйткені DataSet-те деректерді шығарғаннан кейін біз байланыс орнатылғанына немесе орнатылмағанына қарамастан, осы деректермен жұмыс істей аламыз. Яғни, деректер базасынан деректерді алғаннан кейін қосымшаны деректер көзінен ажыратуға болады.

C# және MS SQL Server

Деректер базасын құру

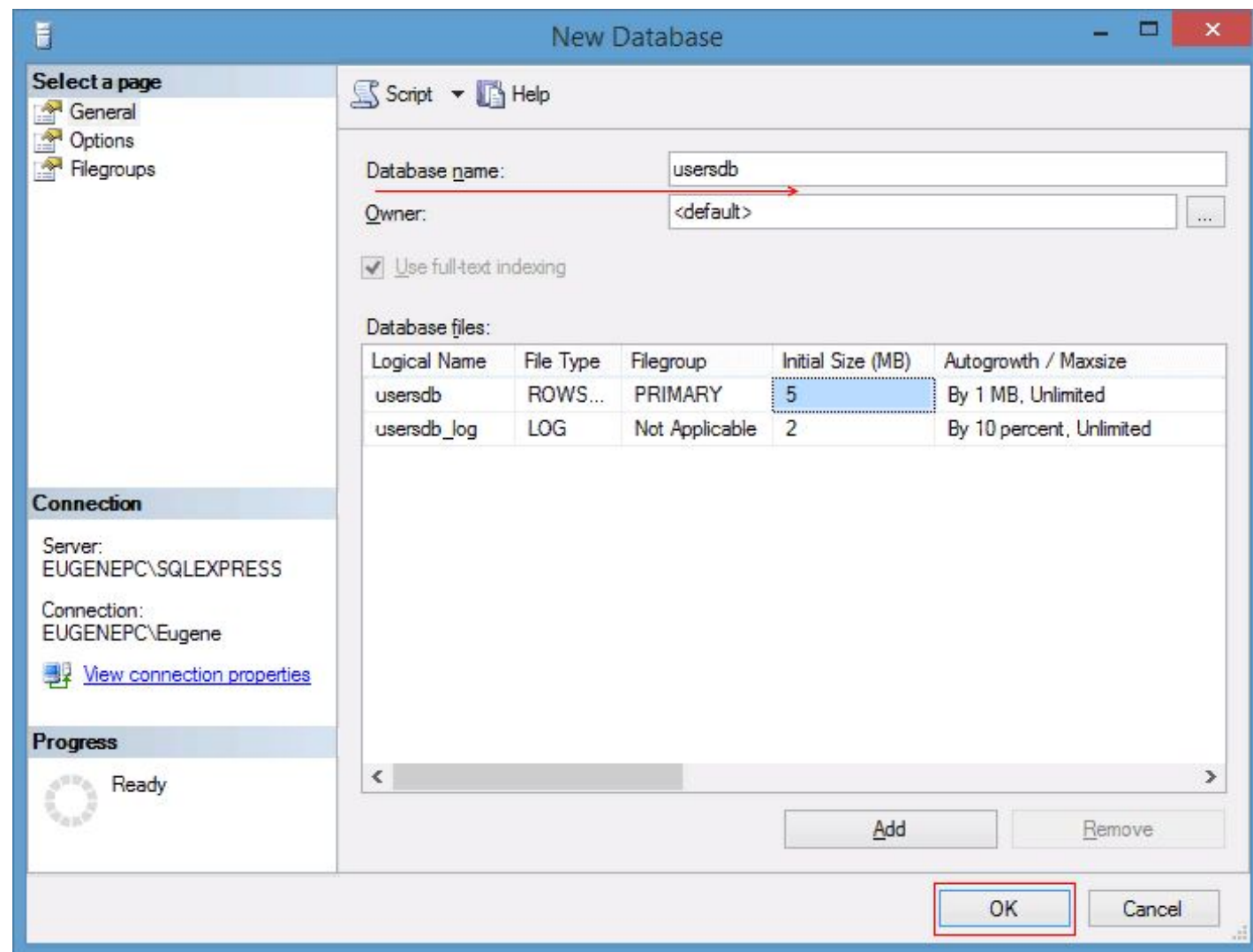
Деректер базасымен жұмыс істеу үшін, ең алдымен, қандай да бір мәліметтер базасы болуы керек. Бұл жағдайда біз ADO.NET негізгі ұғымдарын қарастырамыз, MS SQL Server-і мысалы ретінде. Сондықтан алдымен **SQL Server Express** орнату керек. Орнату үшін барлық қажетті материалдарды мына жерден табуға болады **<https://www.microsoft.com/en-US/download/details.aspx?id=42299>**. Сервермен бірге **SQL Server Management Studio** арнайы бағдарламасы орнатылған, ол сервердегі мәліметтер базасын басқару үшін қолданылады.

Бастау үшін MS SQL Server-де қарапайым мәліметтер базасын жасаңыз. Ол үшін SQL Server Management Studio бағдарламасын ашып, тінтуірдің оң жақ батырмасымен деректер түйінін басыңыз. Осыдан кейін ашылған контекстік мәзірде жаңа деректер базасын таңдаңыз:

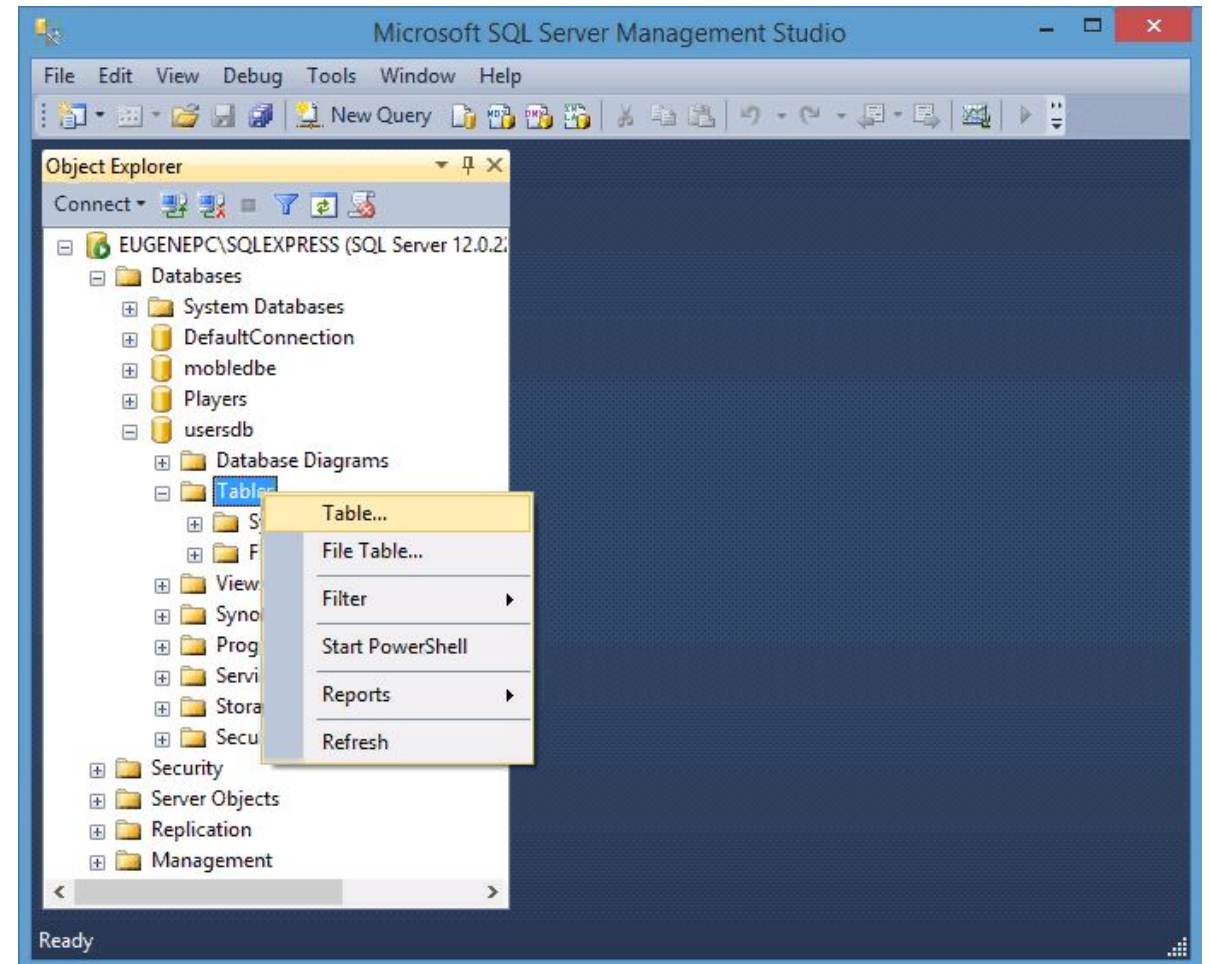


Осыдан кейін бізге мәліметтер базасын құру терезесі ашылады:

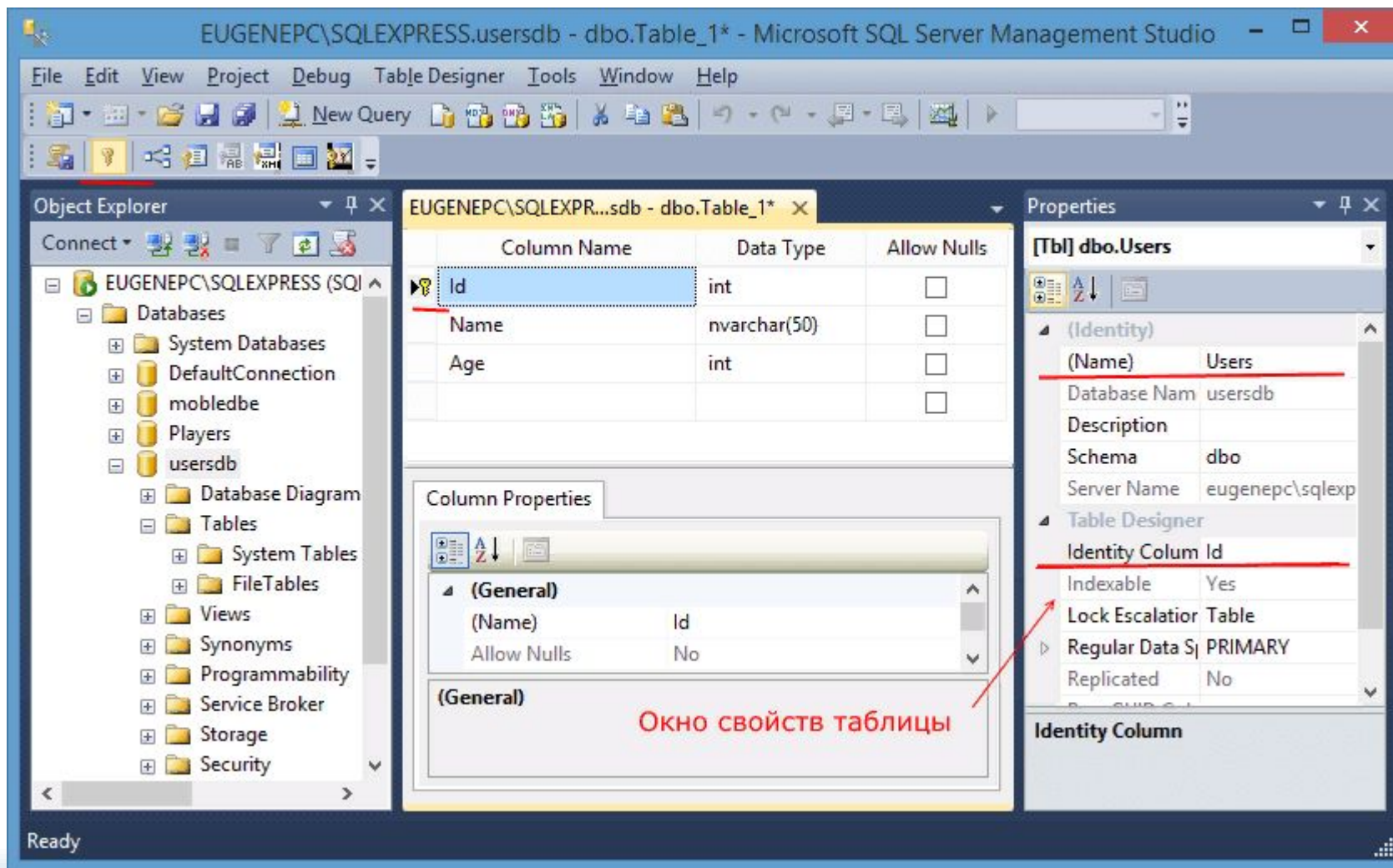
Онда Database Name өрісіне дерекқордың атын енгізу керек. **Usersdb** деп атап, ОК түймесін басыңыз.



Осыдан кейін, жаңа құрылған usersdb дерекқорын ұсынатын **Databases** түйінінде жаңа элемент пайда болады. Біз оны ашып, **Tables** түйініне тінтуірдің оң жақ батырмасымен басамыз:



Содан кейін бізге кесте дизайнері ашылады:



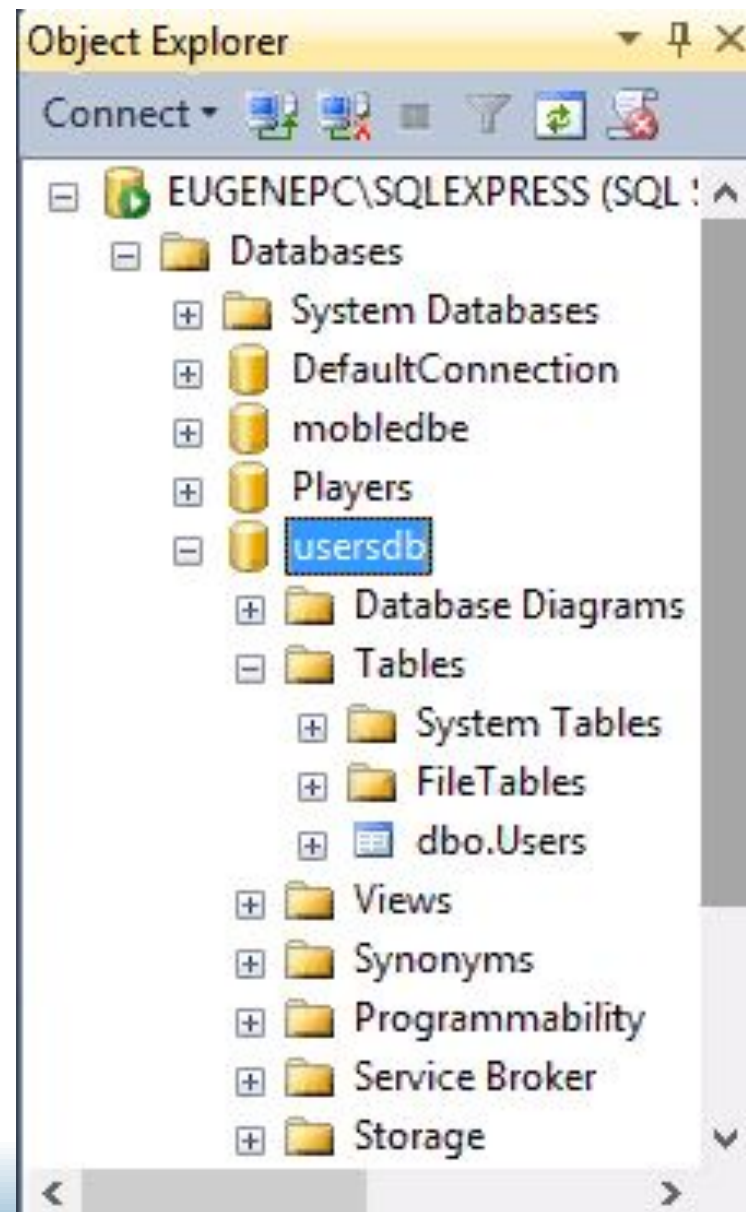
Онда сіз үш бағанды көрсетуіңіз керек: Id, Name және Age, олар сәйкесінше пайдаланушының бірегей идентификаторын, оның атын және жасын білдіреді. Бірінші және үшінші бағандарда int түрін (яғни бүтін сан), ал name бағанында nvarchar түрін (жол) көрсету керек.

Сонымен қатар, name өрісіндегі кесте қасиеттері терезесінде кестенің атын – Users деп енгізу керек, ал Identity өрісіне Id енгізу керек, яғни id бағанының идентификатор болатындығын көрсетеді.

Соңында біз курсорды Id бағанына орнатып, бағдарламаның құралдар тақтасында алтын кілтті басуымыз керек. Осыдан кейін Id өрісіне қарама-қарсы алтын кілт пайда болуы керек. Бұл кілт Id бағанының бастапқы кілт рөлін атқаратындығын көрсетеді.

Осыдан кейін сақтау түймесін, содан кейін F5 пернесін (жаңарту) басыңыз және біздің дерекқорымыздың түйінінде **dbo.Users** деп аталатын жаңа кесте пайда болады:

Сонымен, біз деректерді сақтау функцияларын орындайтын мәліметтер базасы мен кесте құрдық. Енді біз оларға қосылу үшін ADO.NET инфрақұрылымын пайдалана аламыз.



НАЗАРЛАРЫҢЫЗҒА РАХМЕТ!!!