

How to create ASP.net identity tables in an already created database using code first?

Asked 3 years, 1 month ago Active 1 year ago Viewed 24k times

16
My application has been in development for about a month. I now decided to use ASP.NET Identity. I already have the view models for identity but need to create the tables. I was thinking and I am not exactly sure why I do not have the tables already if I have the view models?? I have drop create on in the initializer with my own custom context, I just need to know how to get EF to build the included Identity tables for users and roles? I looked around and none of the posted answers seem to be what I need?

c# asp.net entity-framework asp.net-identity

edited Apr 17 '17 at 20:59

asked Apr 17 '17 at 20:55



LeRainman

223 ● 1 ● 2 ● 8

1 If you want to just run SQL scripts instead of using EF Migrations you could check [my public gist for ASP.NET Core Identity + IdentityServer4 SQL scripts](#). – Jeroen Jan 18 '18 at 7:54

4 Answers

Active Oldest Votes

Consider Migrations

12
If applicable, you need to consider building a [migration](#), which will allow you to generate (and potentially execute) the necessary scripts to create the appropriate tables or changes within your database.

By default, you should have some type of `ApplicationDbContext` class that looks like the following which will be used to define your "security-related" database:



```
public class ApplicationDbContext : IdentityDbContext<ApplicationUser>
{
    public ApplicationDbContext()
        : base("DefaultConnection", false)
    {
    }

    // Other code omitted for brevity
}
```

You'll then just need to run the **Enable-Migrations** command in Package Manager Console:

```
Enable-Migrations
```

This should generate a Migrations folder within your application that contains various configuration files that control how migrations are preformed as well as an `InitialCreate` migration. This may only be present if you previously had some Code-First related code within your application, if not, don't worry about it. You can then try running the **Update-Database** command, which should execute any migrations (including an initial one) against your database:

```
Update-Database
```

Once your database has been updated, you can continue to make changes to your model and simply create and execute new migrations through the **Add-Migration** command and the previous Update-Database command:

```
Add-Migration "AddedAnotherPropertyToFoo"
Update-Database
```

edited Apr 17 '17 at 21:13

answered Apr 17 '17 at 21:05



Rion Williams

66.3k ● 34 ● 171 ● 292

1 I do know about migrations, I was just trying to find a way to get the tables created without the migrations. Is this not possible if the database is already created? – [LeRainman](#) Apr 17 '17 at 21:07

You can find scripts in [this blog post](#) that you could use to generate the necessary identity-related tables without using migrations I suppose. The post essentially just generates the tables automatically and then the author scripts out each one (e.g. users, roles, etc.) that you could execute on your own. – [Rion Williams](#) Apr 17 '17 at 21:12 ✎

Run this SQL Script on your database and get it done.

6

/****** Object: Table [dbo].[AspNetRoles] Script Date: 15-Mar-17 10:27:06 PM *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

```
CREATE TABLE [dbo].[AspNetRoles](
    [Id] [nvarchar](128) NOT NULL,
    [Name] [nvarchar](256) NOT NULL,
    CONSTRAINT [PK_dbo.AspNetRoles] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
    ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
```

GO

/****** Object: Table [dbo].[AspNetUserClaims] Script Date: 15-Mar-17 10:27:06 PM
*****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

```
CREATE TABLE [dbo].[AspNetUserClaims](
    [Id] [int] IDENTITY(1,1) NOT NULL,
```

```
[UserId] [nvarchar](128) NOT NULL,  
[ClaimType] [nvarchar](max) NULL,  
[ClaimValue] [nvarchar](max) NULL,  
CONSTRAINT [PK_dbo.AspNetUserClaims] PRIMARY KEY CLUSTERED  
(  
    [Id] ASC  
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,  
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]  
  
GO  
  
/***** Object: Table [dbo].[AspNetUserLogins]    Script Date: 15-Mar-17 10:27:06 PM  
*****/  
  
SET ANSI_NULLS ON  
  
GO  
  
SET QUOTED_IDENTIFIER ON  
  
GO  
  
CREATE TABLE [dbo].[AspNetUserLogins](  
    [LoginProvider] [nvarchar](128) NOT NULL,  
    [ProviderKey] [nvarchar](128) NOT NULL,  
    [UserId] [nvarchar](128) NOT NULL,  
    CONSTRAINT [PK_dbo.AspNetUserLogins] PRIMARY KEY CLUSTERED  
(  
        [LoginProvider] ASC,  
        [ProviderKey] ASC,
```

```
[UserId] ASC

)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/***** Object: Table [dbo].[AspNetUserRoles]    Script Date: 15-Mar-17 10:27:06 PM
*****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

CREATE TABLE [dbo].[AspNetUserRoles](
    [UserId] [nvarchar](128) NOT NULL,
    [RoleId] [nvarchar](128) NOT NULL,
CONSTRAINT [PK_dbo.AspNetUserRoles] PRIMARY KEY CLUSTERED
(
    [UserId] ASC,
    [RoleId] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/***** Object: Table [dbo].[AspNetUsers]    Script Date: 15-Mar-17 10:27:06 PM *****/

SET ANSI_NULLS ON
```

```
GO

SET QUOTED_IDENTIFIER ON

GO

CREATE TABLE [dbo].[AspNetUsers](

    [Id] [nvarchar](128) NOT NULL,

    [Email] [nvarchar](256) NULL,

    [EmailConfirmed] [bit] NOT NULL,

    [PasswordHash] [nvarchar](max) NULL,

    [SecurityStamp] [nvarchar](max) NULL,

    [PhoneNumber] [nvarchar](max) NULL,

    [PhoneNumberConfirmed] [bit] NOT NULL,

    [TwoFactorEnabled] [bit] NOT NULL,

    [LockoutEndDateUtc] [datetime] NULL,

    [LockoutEnabled] [bit] NOT NULL,

    [AccessFailedCount] [int] NOT NULL,

    [UserName] [nvarchar](256) NOT NULL,

CONSTRAINT [PK_dbo.AspNetUsers] PRIMARY KEY CLUSTERED

(

    [Id] ASC

)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,

ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO

ALTER TABLE [dbo].[AspNetUserClaims] WITH CHECK ADD CONSTRAINT

[FK_dbo.AspNetUserClaims_dbo.AspNetUsers_UserId] FOREIGN KEY([UserId])
```

```
REFERENCES [dbo].[AspNetUsers] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[AspNetUserClaims] CHECK CONSTRAINT
[FK_dbo.AspNetUserClaims_dbo.AspNetUsers_UserId]

GO

ALTER TABLE [dbo].[AspNetUserLogins] WITH CHECK ADD CONSTRAINT
[FK_dbo.AspNetUserLogins_dbo.AspNetUsers_UserId] FOREIGN KEY([UserId])

REFERENCES [dbo].[AspNetUsers] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[AspNetUserLogins] CHECK CONSTRAINT
[FK_dbo.AspNetUserLogins_dbo.AspNetUsers_UserId]

GO

ALTER TABLE [dbo].[AspNetUserRoles] WITH CHECK ADD CONSTRAINT
[FK_dbo.AspNetUserRoles_dbo.AspNetRoles_RoleId] FOREIGN KEY([RoleId])

REFERENCES [dbo].[AspNetRoles] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[AspNetUserRoles] CHECK CONSTRAINT
[FK_dbo.AspNetUserRoles_dbo.AspNetRoles_RoleId]

GO

ALTER TABLE [dbo].[AspNetUserRoles] WITH CHECK ADD CONSTRAINT
[FK_dbo.AspNetUserRoles_dbo.AspNetUsers_UserId] FOREIGN KEY([UserId])

REFERENCES [dbo].[AspNetUsers] ([Id])

ON DELETE CASCADE

GO
```

```
ALTER TABLE [dbo].[AspNetUserRoles] CHECK CONSTRAINT  
[FK_dbo.AspNetUserRoles_dbo.AspNetUsers_UserId]  
  
GO
```

answered Jul 11 '17 at 9:00

**Mohammed Osman**

2,018 ● 16 ● 13

1 What version does this support? Where did you get this from? – [Barry MSIH](#) Jul 25 '18 at 17:49

This script works with old MVC. I got it from a database created by new web application with authentication. – [Mohammed Osman](#) Jul 26 '18 at 11:33



5



So after a bit of reading a fiddling i got the answer. All I had to do was finally run the register method from ASP.NET Identity and all the tables were created.

answered Apr 18 '17 at 15:33

**LeRainman**

223 ● 1 ● 2 ● 8

2 What is this Register method? Is it on the Account Controller? – [Vasu Inukollu](#) Sep 24 '19 at 15:57

Note: depending on your setup the tables may create in a local database file. – [John M](#) Nov 11 '19 at 17:17



Perhaps just run the ASP.NET Identity Sql Scripts against the database if you do not want to Enable Migrations:

<https://www.codeproject.com/Tips/677279/SQL-script-for-creating-an-ASP-NET-Identity-Databa>

1

▼

🕒

answered Apr 17 '17 at 21:15

 **Brian Ogden**
15k ● 7 ● 69 ● 138