

My First Web Service with API Rest in Azure

STEP 1 – WE NEW CONNECT WITH OUR DATA

- Now, we need add our data azure server

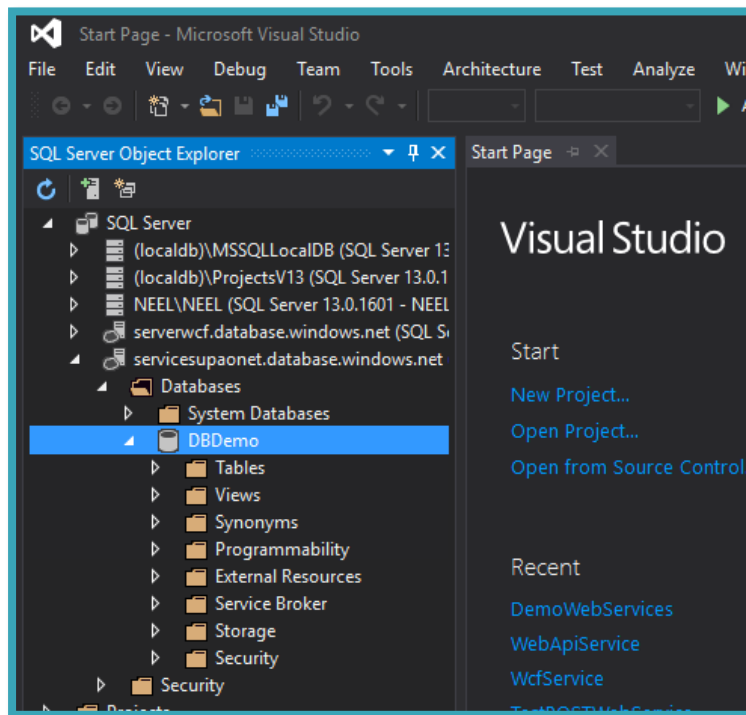
The screenshot shows the 'Connect' dialog box with the following fields and values:

- Server Name:** servicesupaonet.database.windows.net
- Authentication:** Sql Server Authentication
- User Name:** neel
- Password:** (masked with dots)
- Remember Password:** ☐
- Database Name:** DBDemo

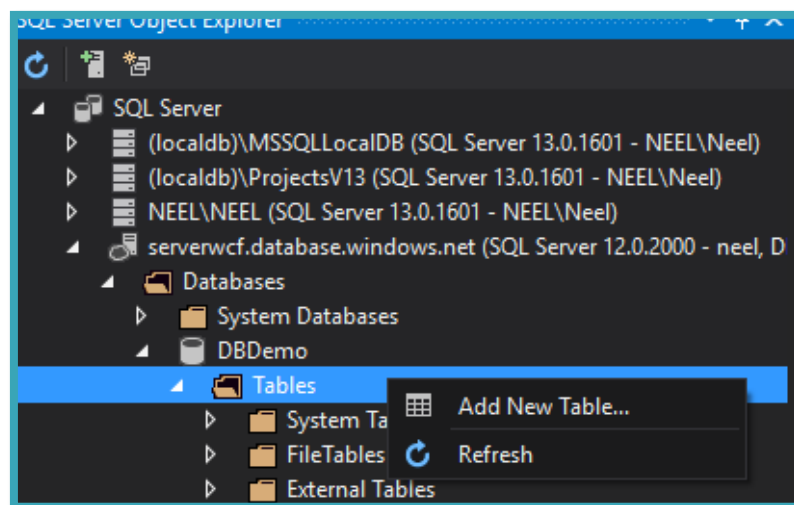
Buttons at the bottom: Connect, Cancel. A 'Details...' link is also present.

STEP 2 – WE CHECK OUR CONNECTION TO OUR DATABASE IN AZURE FROM VISUAL STUDIO

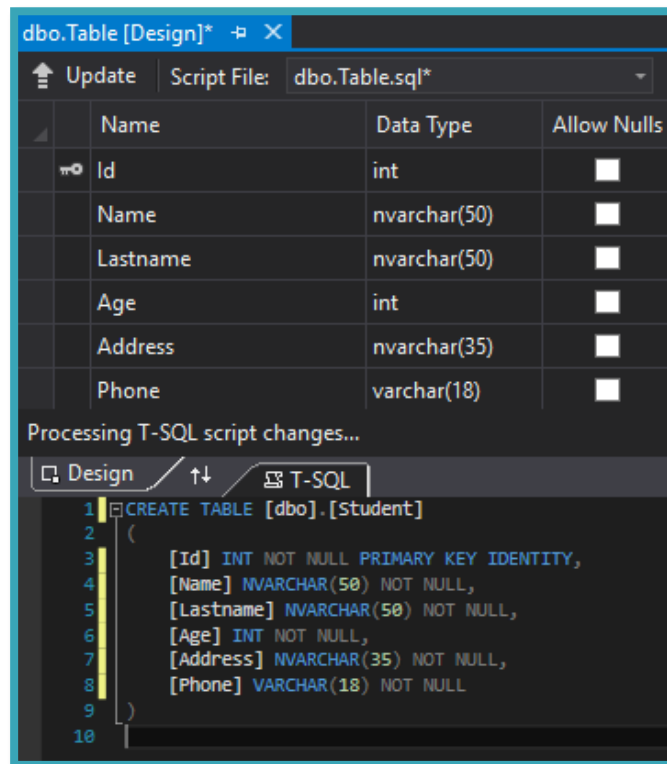
- that's it, now we can use our azure database from visual studio
- I use an azure database



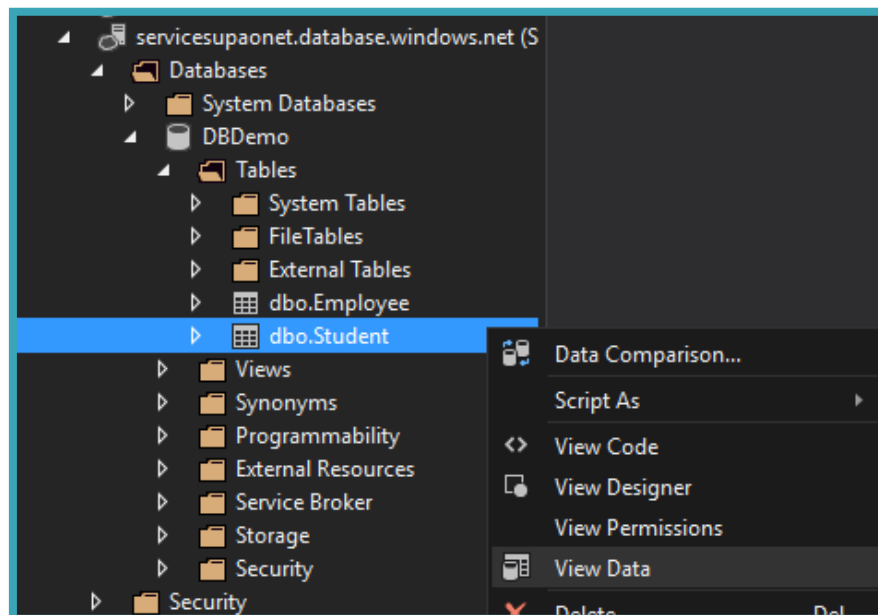
STEP 3 – WE NEED CREATE A TABLE WITHIN OF OUR DATABASE



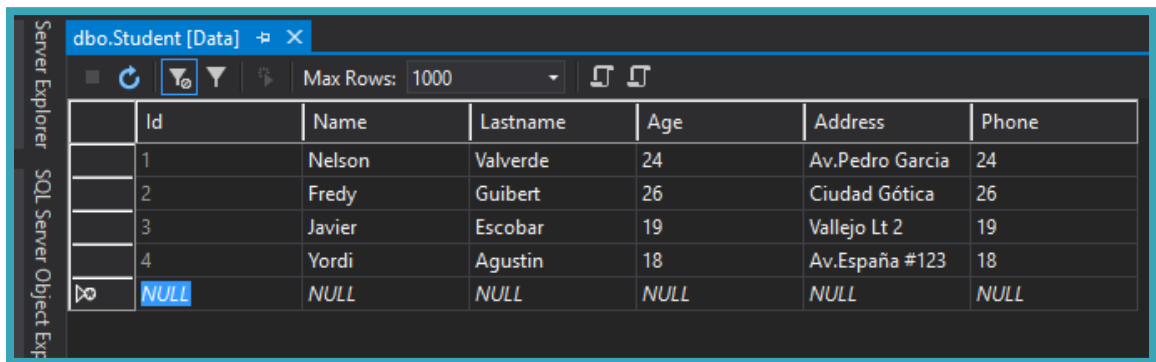
- We need create a table **Student** and update database



- Then add data in our table



- This is my table **Student** with my data

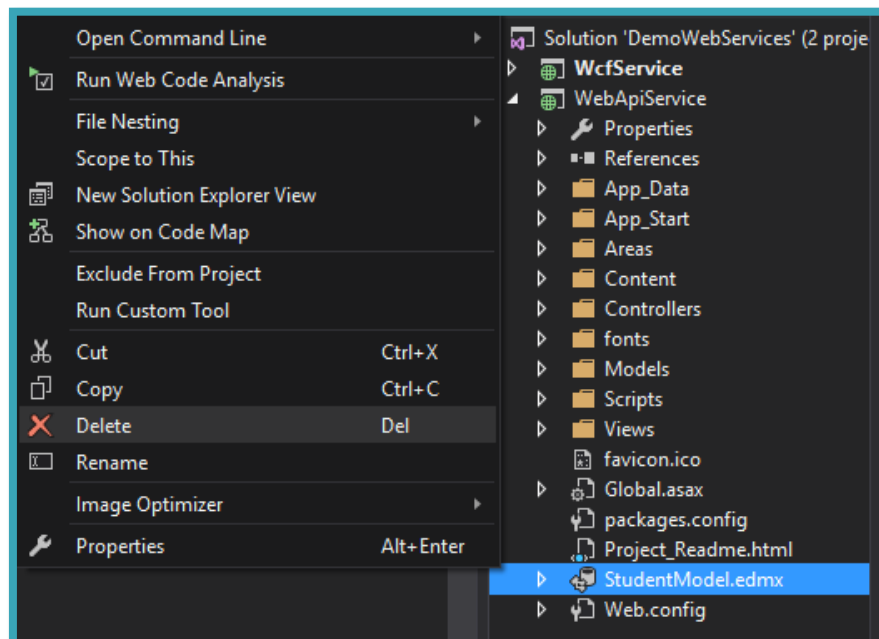


	Id	Name	Lastname	Age	Address	Phone
	1	Nelson	Valverde	24	Av.Pedro Garcia	24
	2	Fredy	Guibert	26	Ciudad Gótica	26
	3	Javier	Escobar	19	Vallejo Lt 2	19
	4	Yordi	Agustin	18	Av.España #123	18
	NULL	NULL	NULL	NULL	NULL	NULL

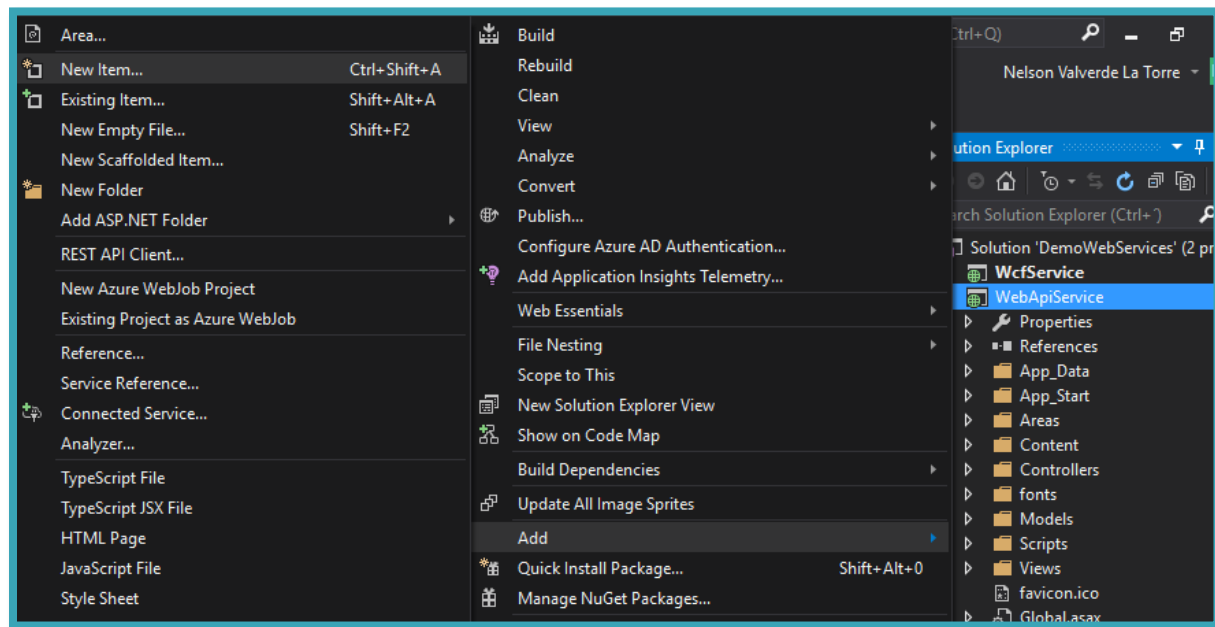
STEP 4 - WE PREPARED OUR PROJECT

- Delete our StudentModel.edmx

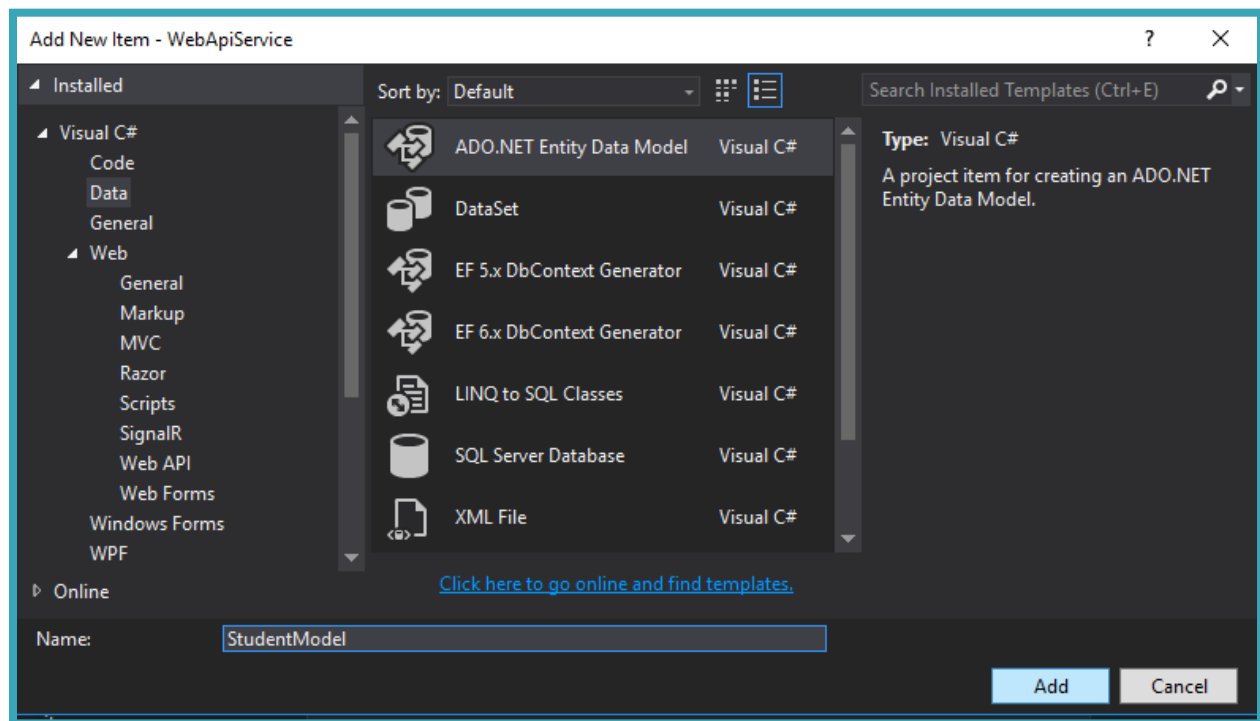
This is our Local Web service of our previous project



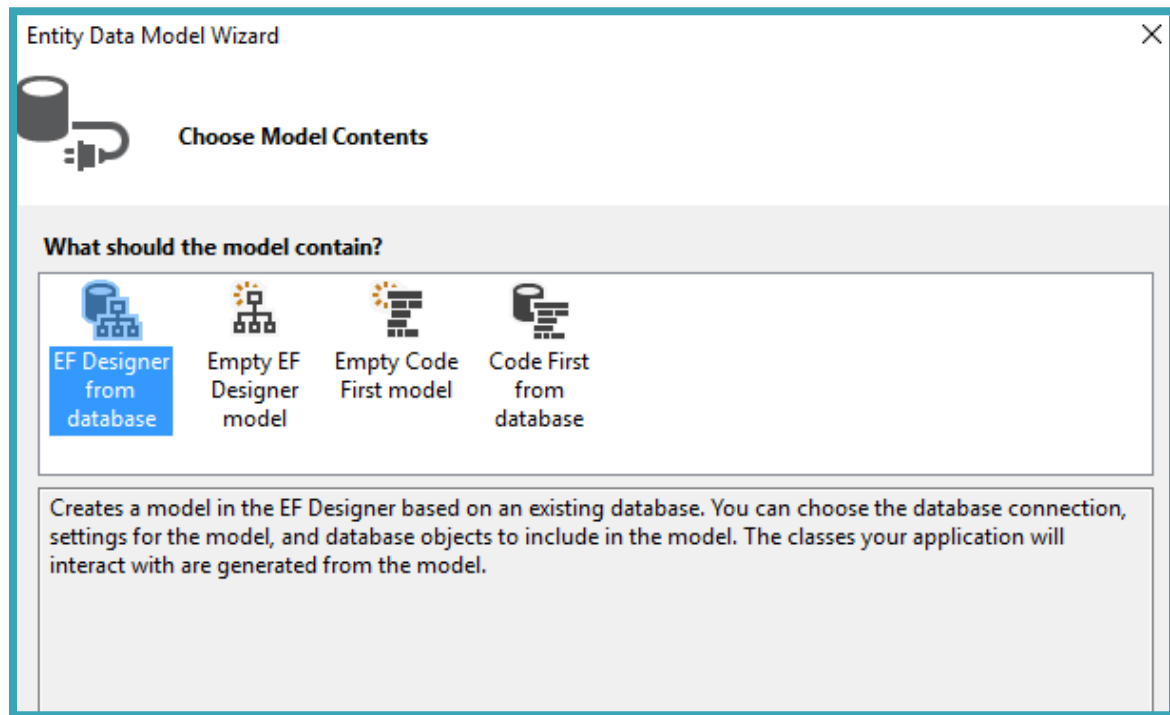
We Add a new item



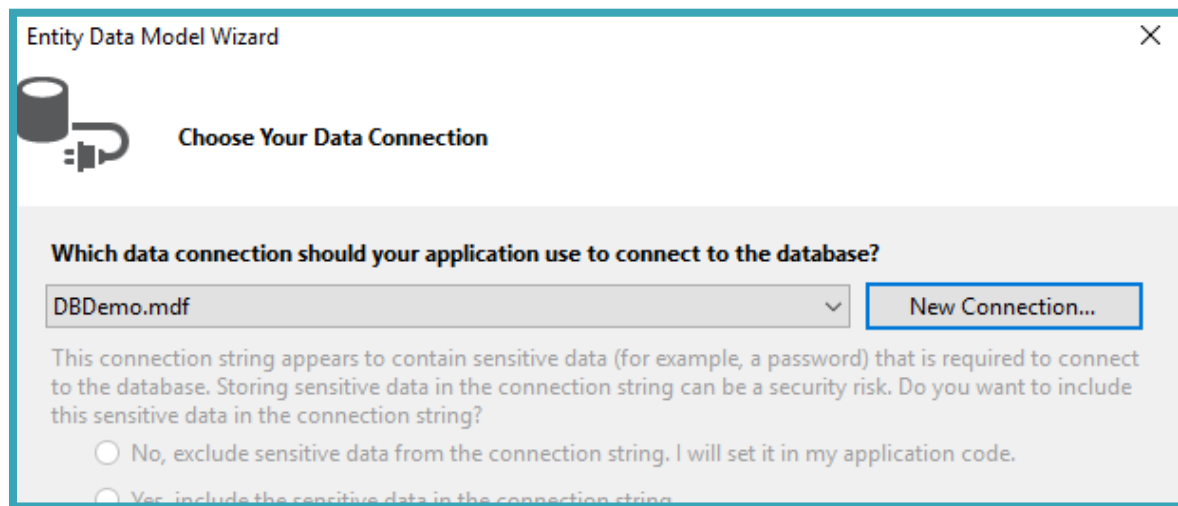
Now we create our new Item → StudentModel



We select EF Designer from database and Next



We select New connection



Now we add our information of azure database

Connection Properties

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:
Microsoft SQL Server (SqlClient) Change...

Server name:
servicesupaonet.database.windows.net Refresh

Log on to the server

Authentication: SQL Server Authentication

User name: neel

Password: ••••••••

☐ Save my password

Connect to a database

☒ Select or enter a database name:
DBDemo

☐ Attach a database file:
Browse...

Logical name:

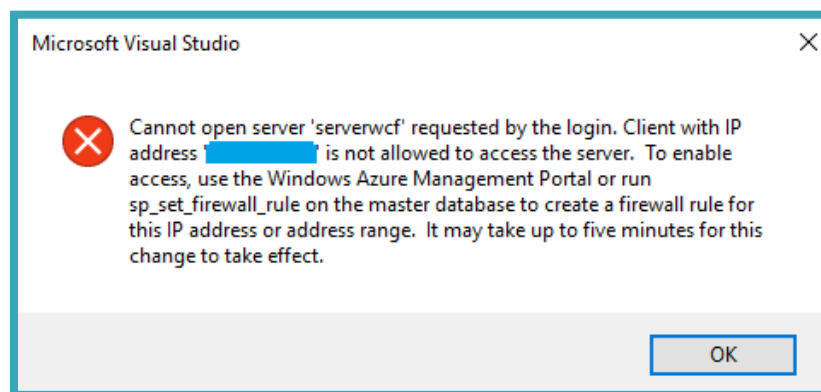
Advanced...

Test Connection OK Cancel

IMPORTANT

Solution N° 1:

If you have this error, you need add our IP in the section of firewall of our azure database



You have to do this:

Add your ip public and save or add a custom ip and Save, then Follow these steps.

The screenshot displays the Azure portal interface for configuring firewall settings for a resource named 'serverwcf'.

- Left Pane (Resources):** A list of resources is shown. The 'serverwcf' resource is selected and highlighted with a blue box, labeled with a blue '1'.
- Center Pane (Settings):** The 'Firewall' setting is selected and highlighted with a blue box, labeled with a blue '2'.
- Right Pane (Firewall Settings):**
 - The 'Add client IP' button is highlighted with a red box.
 - The 'Save' button is highlighted with a blue box, labeled with a blue '4'.
 - A red text annotation 'This is your ip public' points to the 'Add client IP' button.
 - The 'Client IP address' field displays '190.42.87.93'.
 - Below, a table lists firewall rules. The first rule, 'ClientIPAddress_2016-7-6_1...', has empty 'START IP' and 'END IP' fields, which are highlighted with a blue box, labeled with a blue '3'.

Solution N° 2:

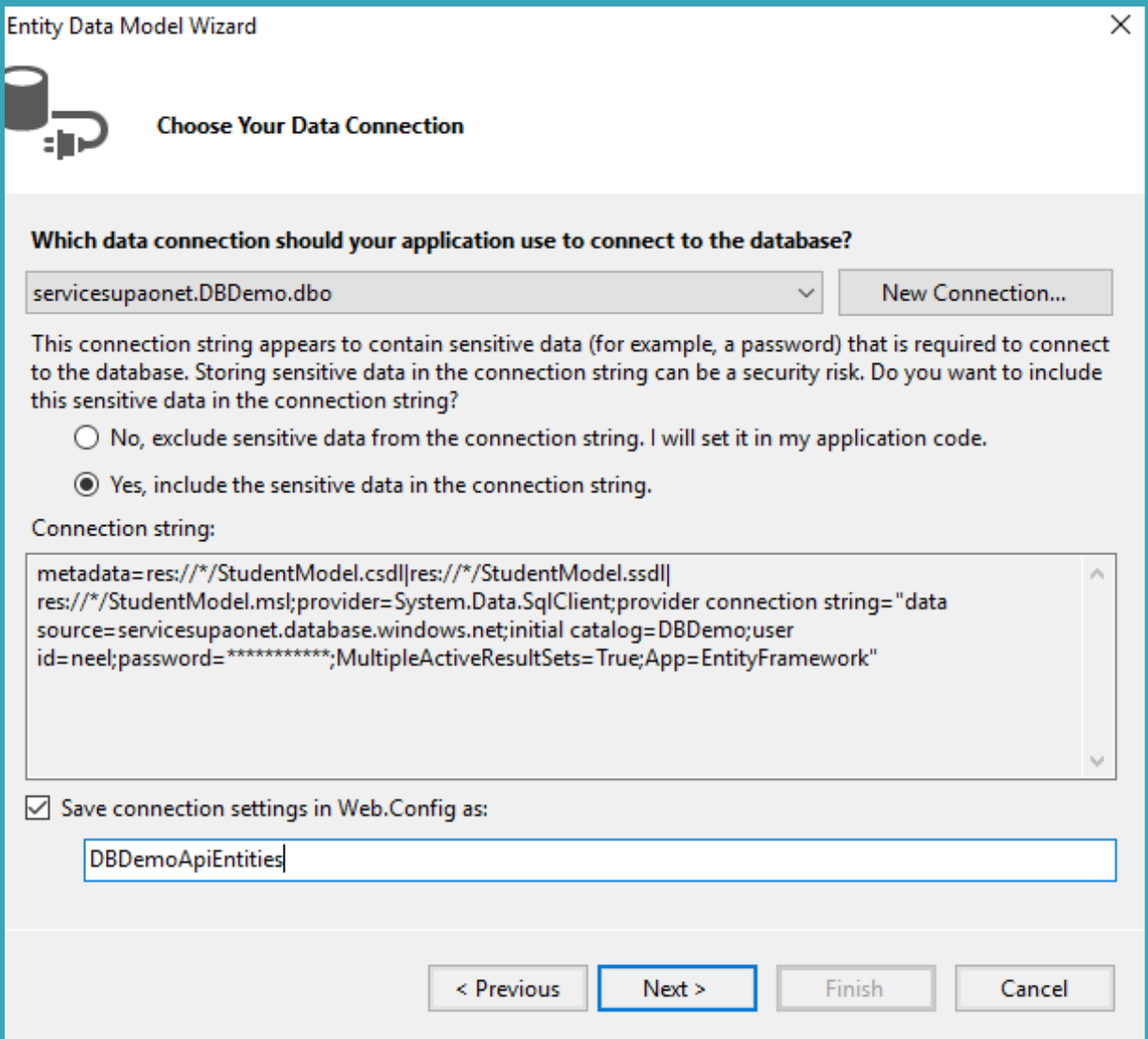
You don't know about your data connection of azure? Get in here, then copy and paste in our Connection Properties of the Page 8

The screenshot displays the Microsoft Azure portal interface. The left sidebar shows the 'SQL databases' resource selected (1). The main pane shows the 'DBDemo - Properties' page. The 'SQL databases' list on the left shows 'DBDemo' selected (2). The 'Properties' tab is selected in the left sidebar (3). The 'Connection strings' section is expanded, showing the 'SERVER NAME' as 'serverwcf.database.windows.net' (4) and the 'SERVER ADMIN LOGIN' as 'neel' (5).

We need do this:

1- this allows add a string connection in our web.config.

2 - this allows add a string connection in our web.config and allows connect our azure database.



The image shows a screenshot of the 'Entity Data Model Wizard' window, specifically the 'Choose Your Data Connection' step. The window has a title bar with the text 'Entity Data Model Wizard' and a close button. Below the title bar is a header area with a database icon and the text 'Choose Your Data Connection'. The main content area contains a question: 'Which data connection should your application use to connect to the database?'. Below this question is a dropdown menu showing 'servicesupaonet.DBDemo.dbo' and a 'New Connection...' button. A text block explains that the connection string may contain sensitive data and asks if it should be included. There are two radio buttons: 'No, exclude sensitive data from the connection string. I will set it in my application code.' and 'Yes, include the sensitive data in the connection string.' The 'Yes' option is selected. Below the radio buttons is a text box labeled 'Connection string:' containing a long string of metadata and connection details. At the bottom, there is a checkbox labeled 'Save connection settings in Web.Config as:' which is checked. Below this checkbox is a text box containing 'DBDemoApiEntities'. At the very bottom of the window are four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'.

Entity Data Model Wizard

Choose Your Data Connection

Which data connection should your application use to connect to the database?

servicesupaonet.DBDemo.dbo

New Connection...

This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

☐ No, exclude sensitive data from the connection string. I will set it in my application code.

☒ Yes, include the sensitive data in the connection string.

Connection string:

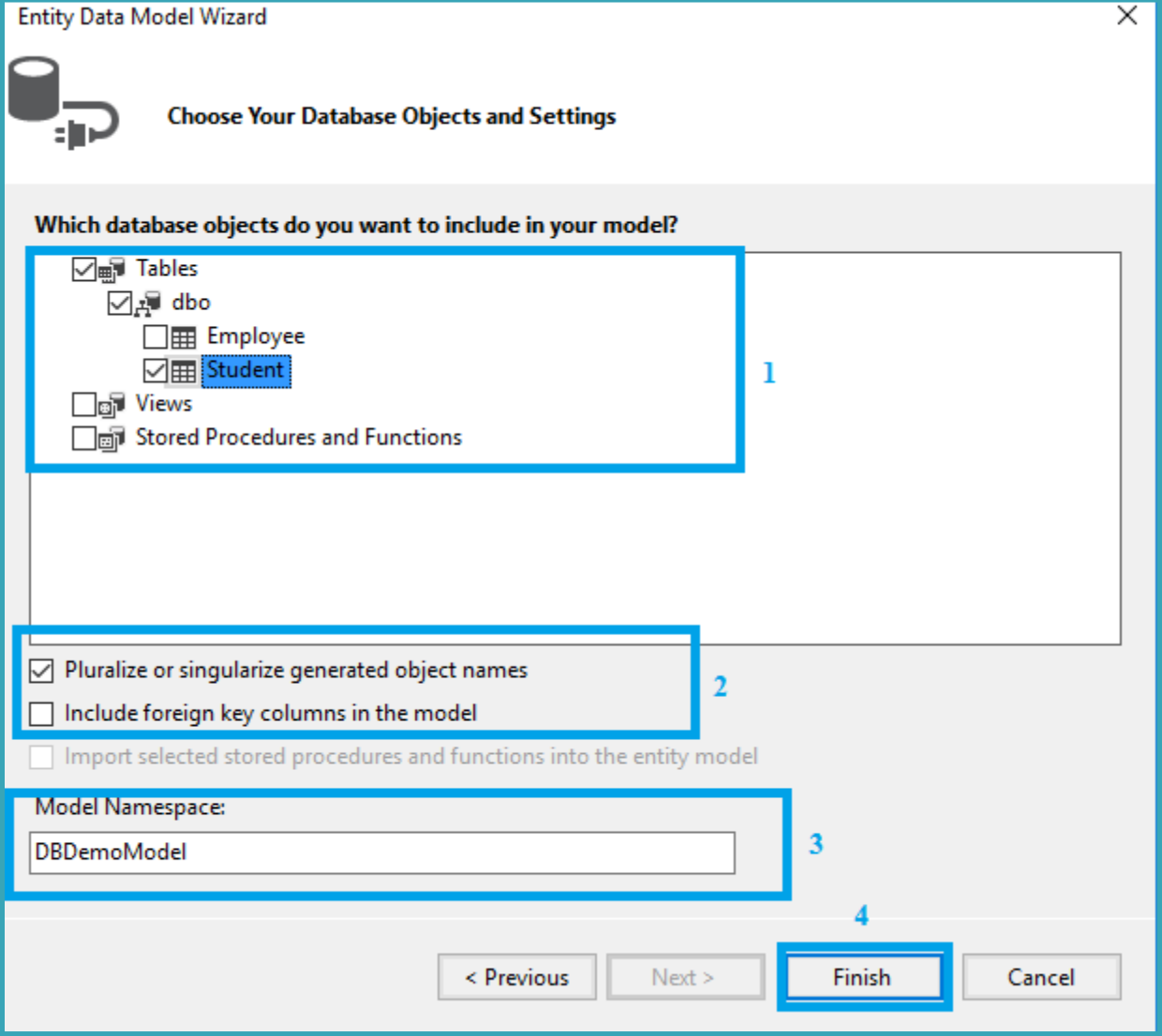
metadata=res://*/StudentModel.csdl|res://*/StudentModel.ssdl|
res://*/StudentModel.msl;provider=System.Data.SqlClient;provider connection string="data
source=servicesupaonet.database.windows.net;initial catalog=DBDemo;user
id=neel;password=*****;MultipleActiveResultSets=True;App=EntityFramework"

☒ Save connection settings in Web.Config as:

DBDemoApiEntities

< Previous Next > Finish Cancel

- We need add our tables of our database
- Usually, you want entity names singular and entity set names plural.
- This is necessary specify If you have a database relational



The image shows the 'Entity Data Model Wizard' dialog box, titled 'Choose Your Database Objects and Settings'. It contains several sections and options:

- Which database objects do you want to include in your model?**
 - ☒ Tables
 - ☒ dbo
 - ☐ Employee
 - ☒ Student
 - ☐ Views
 - ☐ Stored Procedures and Functions
- ☒ Pluralize or singularize generated object names
- ☐ Include foreign key columns in the model
- ☐ Import selected stored procedures and functions into the entity model
- Model Namespace:**
DBDemoModel
- Navigation buttons: < Previous, Next >, **Finish**, Cancel

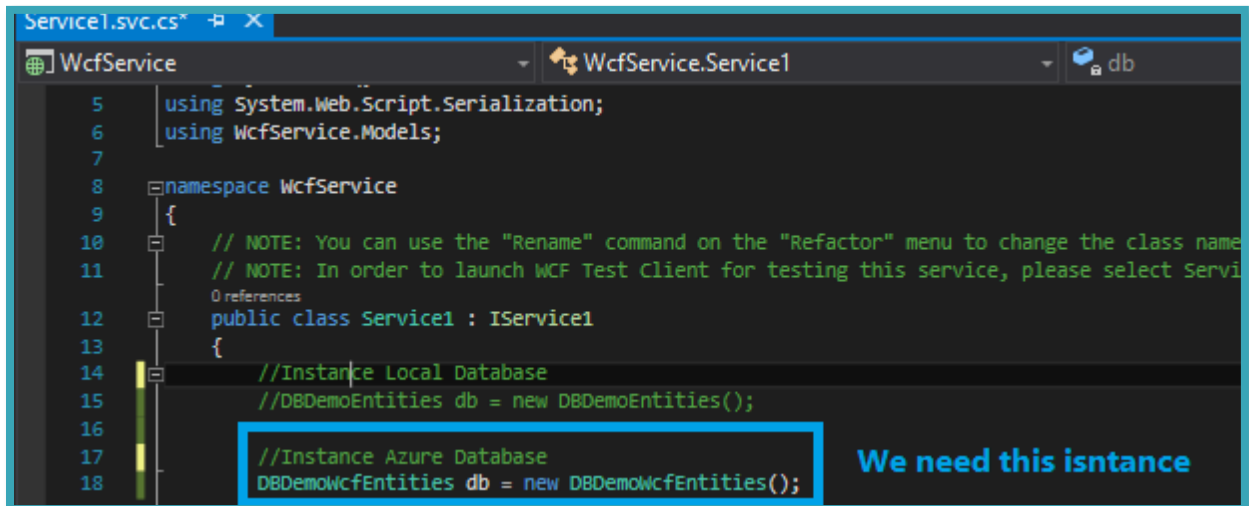
Blue boxes and numbers 1 through 4 highlight specific areas: 1 points to the 'Tables' section, 2 points to the 'Pluralize or singularize generated object names' checkbox, 3 points to the 'Model Namespace' text box, and 4 points to the 'Finish' button.

Verify if our connection string exists in our "web.config".

```
<connectionStrings>
  <!--Connection String Local-->
  <!--<add name="DBDemoRestEntities" connectionString="metadata=res://*/StudentModel.csdl|res://*/StudentModel.ssdl|res://*/StudentModel.msl" provider="System.Data.SqlClient" providerInvariant="true" /-->

  <!--Connection string of Azure-->
  <add name="DBDemoApiEntities" connectionString="metadata=res://*/StudentModel.csdl|res://*/StudentModel.ssdl|res://*/StudentModel.msl" provider="System.Data.SqlClient" providerInvariant="true" /-->
</connectionStrings>
```

Then need get in our Service1.svc and add this new instance



```
Service1.svc.cs
WcfService
WcfService.Service1
db

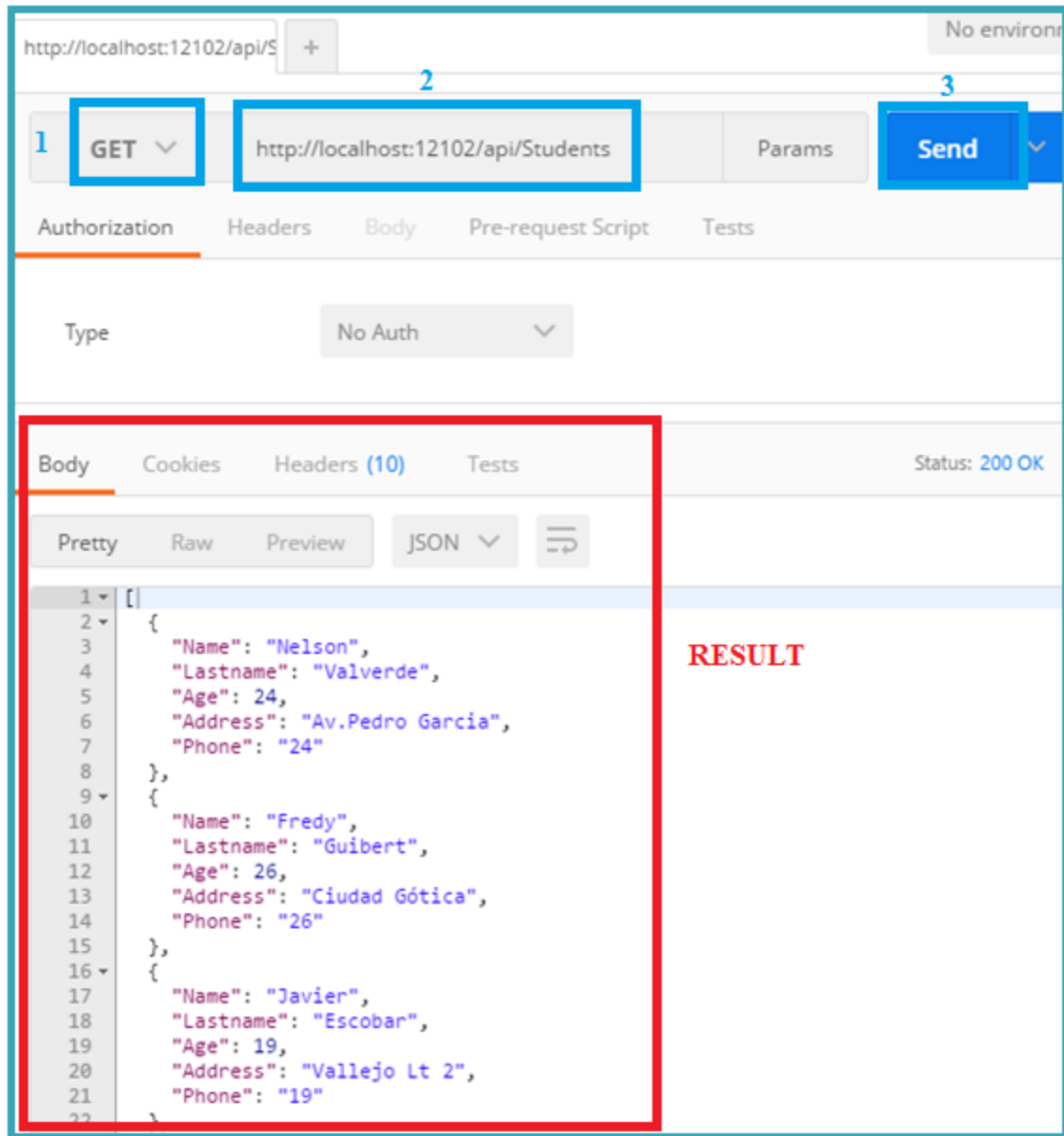
5 using System.Web.Script.Serialization;
6 using WcfService.Models;
7
8 namespace WcfService
9 {
10     // NOTE: You can use the "Rename" command on the "Refactor" menu to change the class name
11     // NOTE: In order to launch WCF Test Client for testing this service, please select Servi
12     0 references
13     public class Service1 : IService1
14     {
15         //Instance Local Database
16         //DBDemoEntities db = new DBDemoEntities();
17         //Instance Azure Database
18         DBDemoWcfEntities db = new DBDemoWcfEntities();
19     }
20 }
```

We need this instance

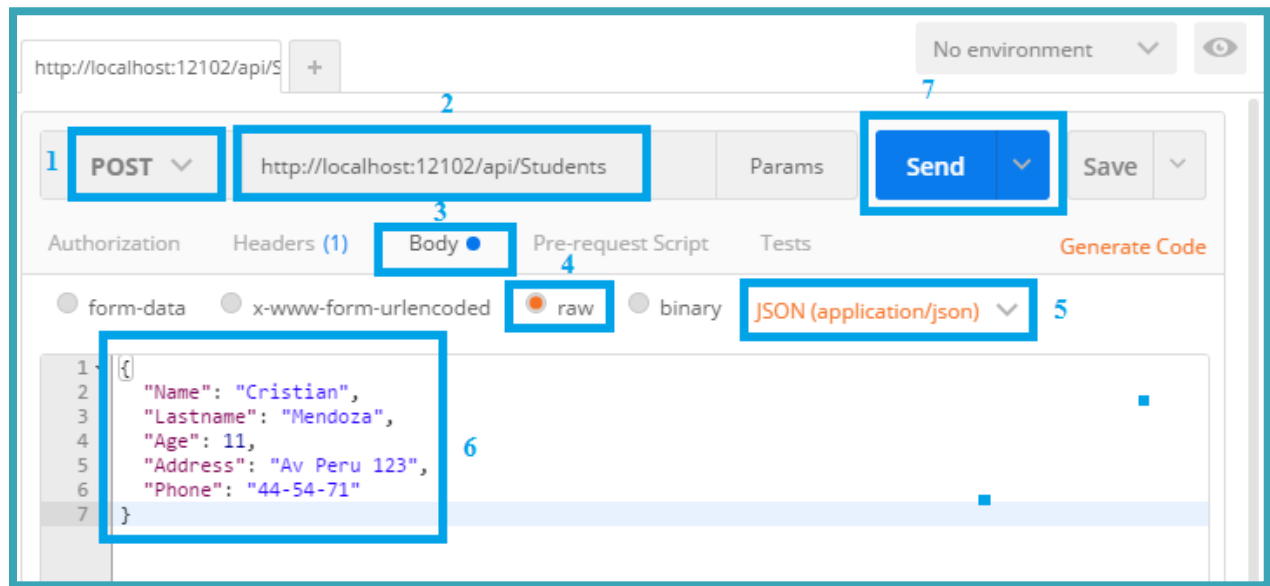
TEST

We need Postman and do this

- 1) This is a simple test of our **List Students**



2) We create our first student in our database



3) We search a student with your id

