

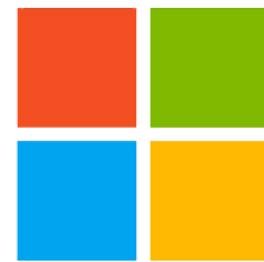


AZURE DAY

SQL Server su Azure:
DBaaS, PaaS, IaaS



Thanks to



Microsoft



avanade



REPLY
CLUSTER

Capgemini

plexin.
CON NOI È SEMPLICE



F A B R I K S



L O B R A

6TH
SENSE



PLURALSIGHT



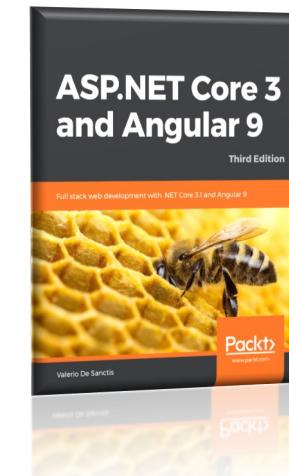
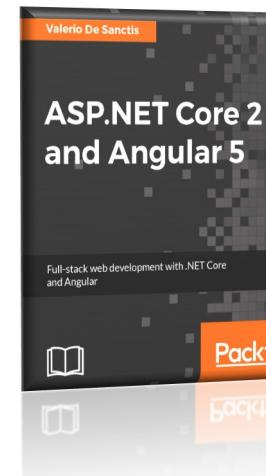
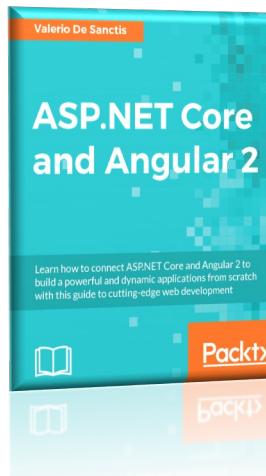
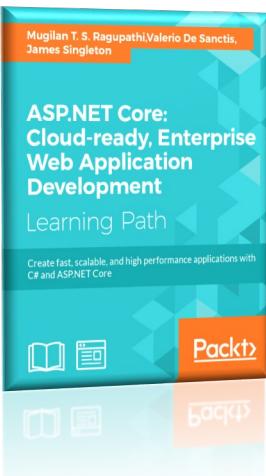
DotNetCode.IT



AZURE DAY

CHI SONO

- Responsabile IT, Sviluppo Software e Information Security presso *AssirecreGroup*
- Fondatore e responsabile di redazione del sito www.ryadel.com
- Coordinatore e Dev Team Manager della community di sviluppo *Ryadel*
- Microsoft MVP for Developer Technologies
- Azure Security Engineer (certificazione Microsoft)
- Data Protection & Security Officer (certificazione TUV Italia)
- Autore della serie di libri *ASP.NET Core and Angular* (*Packt Publishing*)
- Appassionato di sviluppo software, IT architecture design, DBMS e sicurezza



Award Categories
Developer Technologies





INDICE

- DBMS: *on-premise* vs *cloud*
- Azure SQL Database (*DBaaS*)
- Azure SQL Managed Instance (*PaaS*)
- SQL Server su Azure VM (*IaaS*)
- Azure SQL Database: esempio di creazione
- Connessione via SSMS
- Connessione via .NET Core Web App
- Deploy .NET Core Web App su Azure App Services
- Question Time

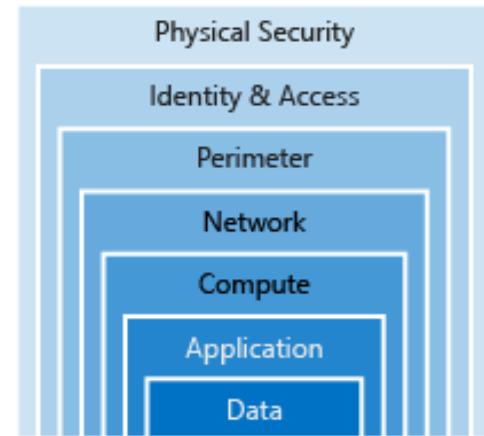


AZURE DAY

DBMS: *on-premise* vs *cloud*

“perché abbandonare una soluzione *on-premise* per mettere i nostri *dati strutturati* in *cloud*? ”

- **Data Protection & Security**
 - Rischi *interni* ed *esterni*
 - Confidenzialità (*Confidentiality*)
 - Integrità (*Integrity*)
 - Disponibilità (*Availability*)
- **Scaling delle risorse**
 - Prestazioni, Spazio, Ridondanza
 - Elasticità e adattabilità
- **Rapidità nel deployment**
 - Modifiche strutturali (CED)
 - Variazioni ai contratti di servizio (fornitori)
 - Acquisti e installazioni



Responsabilità	Locale	IaaS	PaaS	SaaS
Gestione della governance dei dati e				
Endpoint client				
Gestione degli account e degli accessi				
Infrastruttura di identità e directory				
Applicazione				
Controlli di rete				
Sistema operativo				
Host fisici				
Rete fisica				
Centro dati fisico				

Legend:

- Microsoft (Dark Blue)
- Cliente (Light Blue)



AZURE DAY

Azure SQL Database (*DBaaS*)

- **Completamente gestito (fully managed)**
 - Aggiornamenti automatici (OS e DB Engine)
 - Business Continuity
 - Backup automatici
 - Advanced Threat Protection
 - 99.99% uptime (garantito da SLA)
- **Tre modalità di deploy**
 - Database singolo
 - Elastic pool
 - SQL Server logico
- **Tre modelli di acquisto**
 - DTU-based
 - vCore-based
 - General Purpose
 - Hyperscale
 - Business Critical
 - Serverless





AZURE DAY

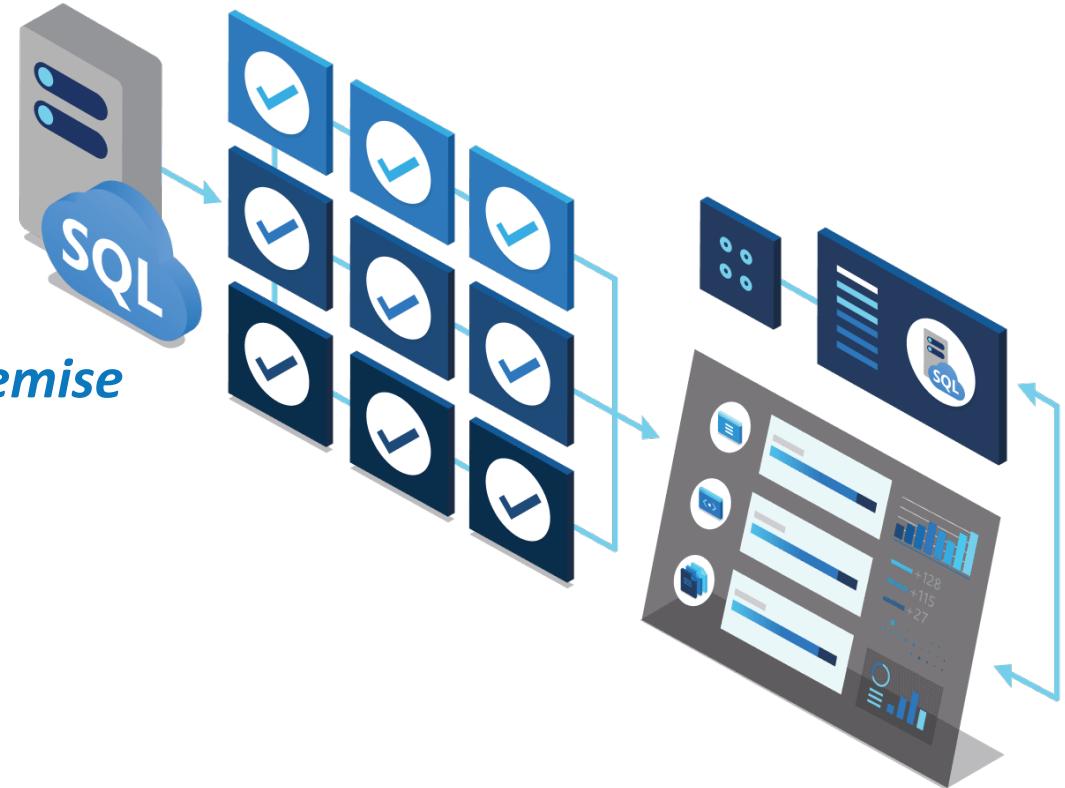
Azure SQL Managed Instance (PaaS)

- **Completamente gestito (*fully managed*)**

- Aggiornamenti automatici (OS e DB Engine)
- Business Continuity
- Backup automatici
- Advanced Threat Protection
- 99.99% uptime (garantito da SLA)

- **Funzionalità simili a una istanza SQL Server on-premise**

- Gestione manuale di BACKUP e RESTORE
- Funzionalità di logging e monitoring (CDC)
- Configurazione collation, charset e timezone
- Common Language Runtime
- Cross-database query
- Cross-database transaction
- Linked Server
- OPENDATASOURCE e OPENQUERY
- DbMail
- Viste Partizionate
- SQL Server Agent & SQL Server Auditing





AZURE DAY

SQL Server su Azure VM (IaaS)

- **Istanza reale di SQL Server 2019**

- Massima versatilità
- Configurabilità OS-level
- Pieno controllo sul Database Engine
 - Start/stop del servizio
 - Attività di maintenance e patching
 - Scelta del Recovery Model
- Piena compatibilità per migrazioni da sistemi *on-premise*

- **Versatilità a livello di versione (SQL Engine)**

- Supporto di tutte le versioni di SQL Server da 2008 a 2019

- **Versatilità a livello di sistema operativo**

- Windows (SQL Server 2008-2016)
- Windows, Linux e Container Docker (da SQL Server 2017-2019)

- **Due licensing model**

- Gestita da Microsoft: Free (*Developer, Express*) o a pagamento (*Web, Standard, Enterprise*)
- Bring Your Own License (BYOL)



Red Hat



SUSE



ubuntu



Windows
Server



AZURE DAY

Azure SQL Database: esempio di creazione

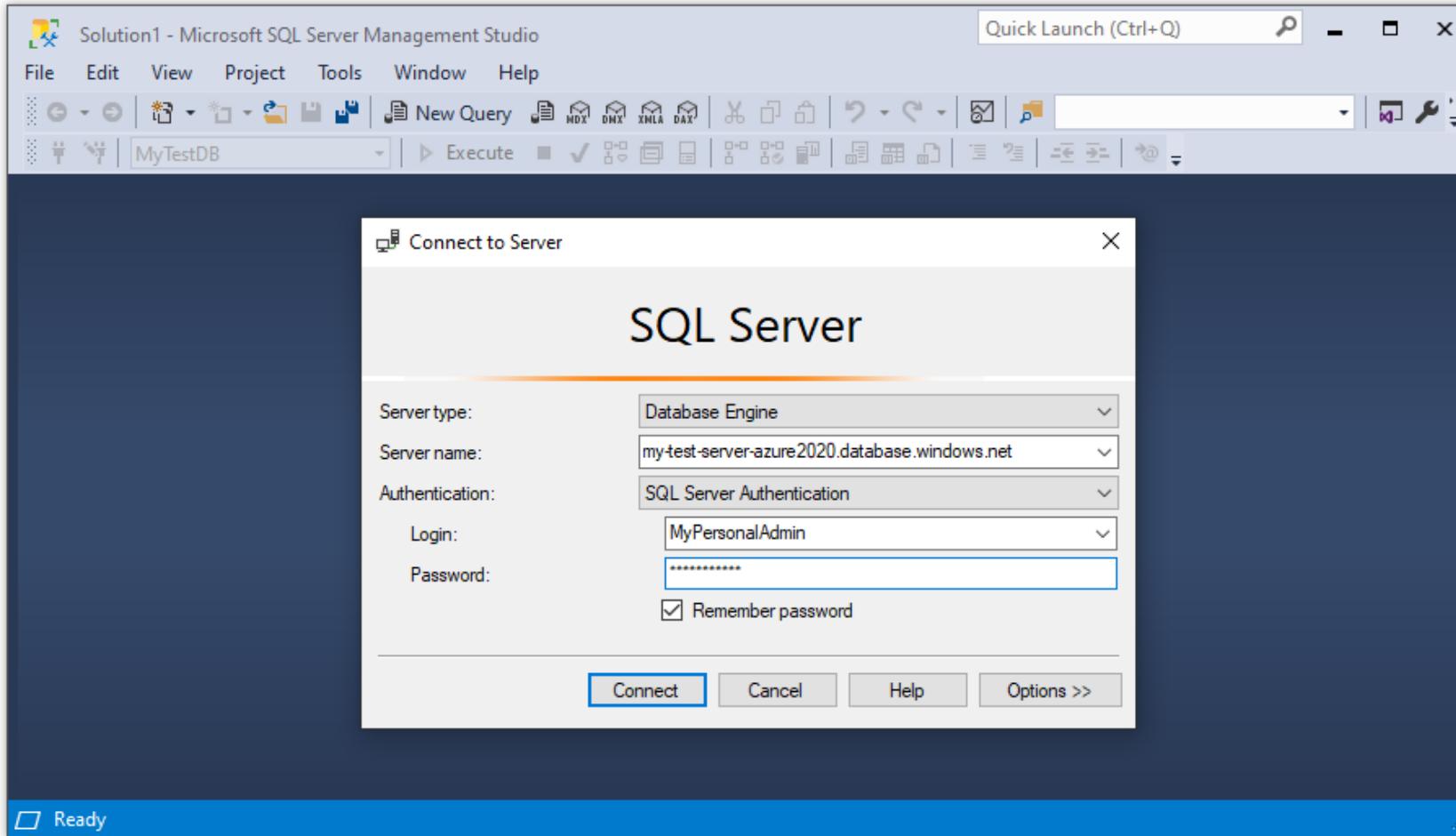




AZURE DAY

Azure SQL DB: connessione via SSMS - 1

1 – Inserire le credenziali di accesso al Database Server Logico

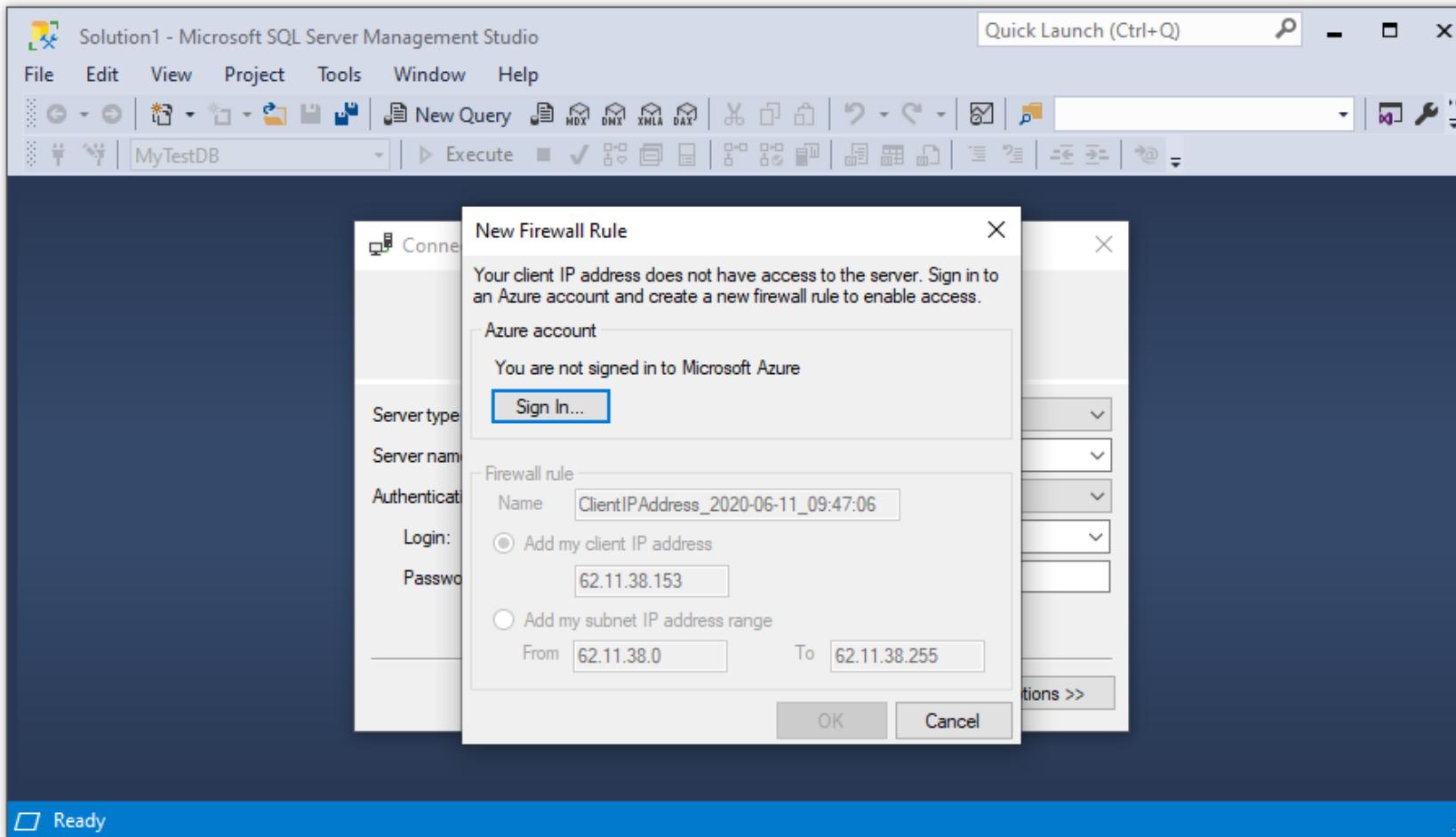




AZURE DAY

Azure SQL DB: connessione via SSMS - 2

2 – Effettuare il login a Microsoft Azure

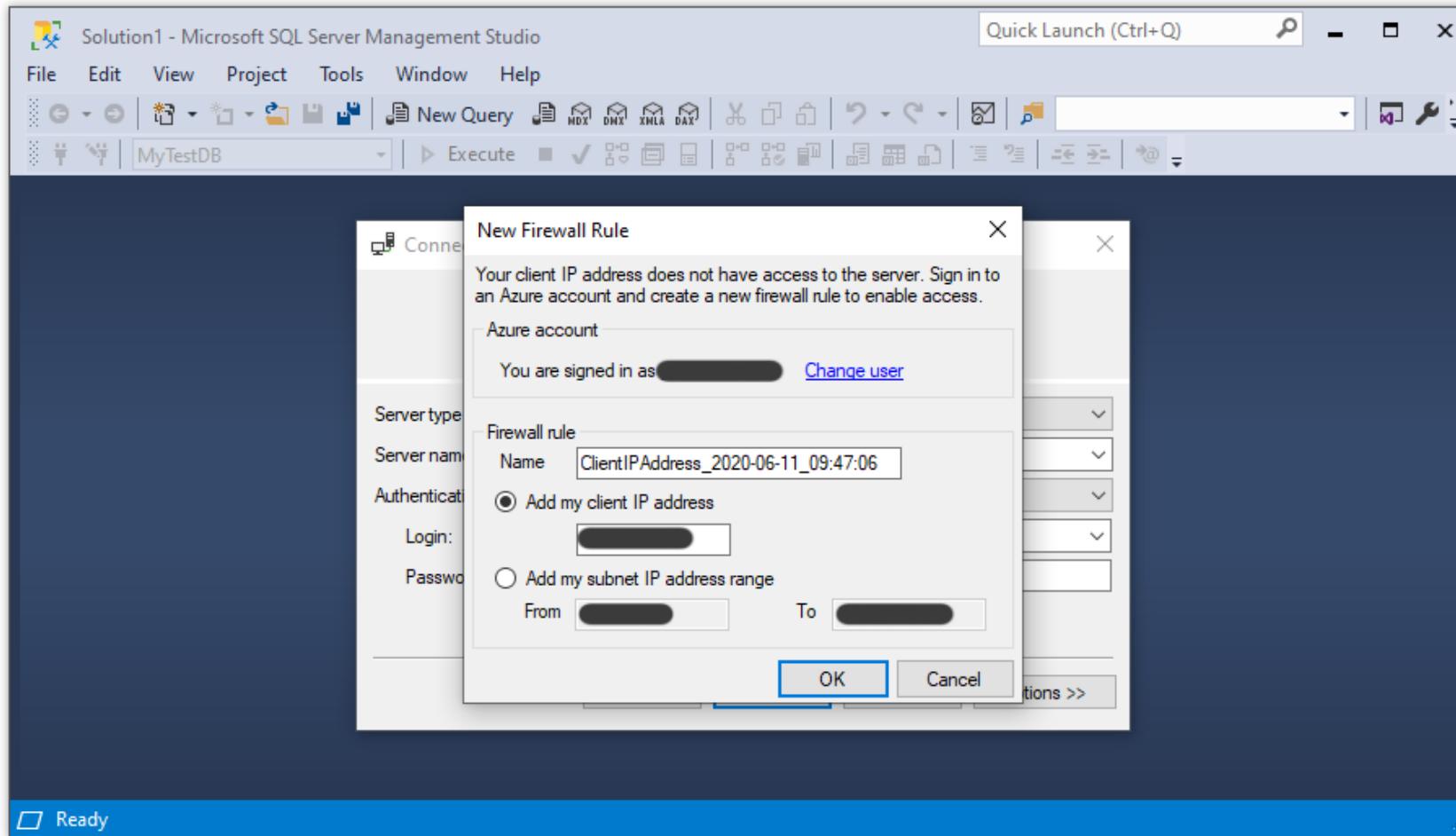




AZURE DAY

Azure SQL DB: connessione via SSMS - 3

3 – Autorizzare la creazione di una regola Firewall per consentire l'accesso





AZURE DAY

Azure SQL DB: connessione via SSMS - 4

4 – Utilizzare l’Object Explorer di SSMS per accedere ai dati

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. The title bar indicates the connection is to `my-test-server-azure2020.database.windows.net.MyTestDB` as MyPersonalAdmin (82). The main window is divided into several panes:

- Object Explorer** (left pane): Shows the database structure. The `SalesLT.Customer` table is selected and highlighted.
- SQLQuery2.sql** (top right pane): Displays a T-SQL script:

```
***** Script for SelectTopNRows command from SSMS *****
SELECT TOP (1000) [CustomerID]
    ,[NameStyle]
    ,[Title]
    ,[FirstName]
```
- Results** (bottom right pane): Shows the output of the query as a grid. The first 10 rows of the `SalesLT.Customer` table are displayed, with the `SalesLT.Customer` table name also visible in the results.
- Status Bar** (bottom): Provides connection information and the number of rows affected: `Query... | my-test-server-azure2020.da... | MyPersonalAdmin (82) | MyTestDB | 00:00:00 | 847 rows`.



Azure SQL DB: connessione via SSMS - 5

AZURE DAY

Microsoft Azure Cerca risorse, servizi e documentazione (G+/-)

Home page > Azure SQL >

my-test-server-azure2020 | Firewall e reti virtuali

SQL Server

Cerca (CTRL+/-) Salva Rimuovi + Aggiungi IP client

Nega l'accesso alla rete pubblica ⓘ Sì No

L'impostazione su Sì consente le connessioni solo tramite l'endpoint privato approvato e disabilita tutte le regole del firewall esistenti. [Altre informazioni.](#)

Versione minima TLS ⓘ > 1.0 > 1.1 > 1.2

Si sta impostando la proprietà Versione minima TLS per tutti i database Database SQL e SQL Data Warehouse associati al server. Eventuali tentativi di accesso dai client che usano una versione TLS inferiore alla Versione minima TLS verranno rifiutati.

Sicurezza

- Sicurezza dei dati avanzata
- Controllo
- Firewall e reti virtuali**
- Connessioni endpoint privato
- Transparent Data Encryption

Accesso da risorse Azure:
NON CONSENTITO

my-test-server-azure2020 | Firewall e reti virtuali

SQL Server

Cerca (CTRL+/-) Salva Rimuovi + Aggiungi IP client

Criteri di connessione ⓘ Predefinito Proxy Reindirizzamento

Consenti alle risorse e ai servizi di Azure di accedere a questo server Sì No

Le connessioni dagli indirizzi IP specificati sotto forniscono l'accesso a tutti i database in my-test-server-azure2020.

Indirizzo IP client [blurred]

Nome regola	Indirizzo IP iniziale	Indirizzo IP finale
ClientIPAddress_2020-06-11...	[blurred]	[blurred]

Ottimizzazione automatica

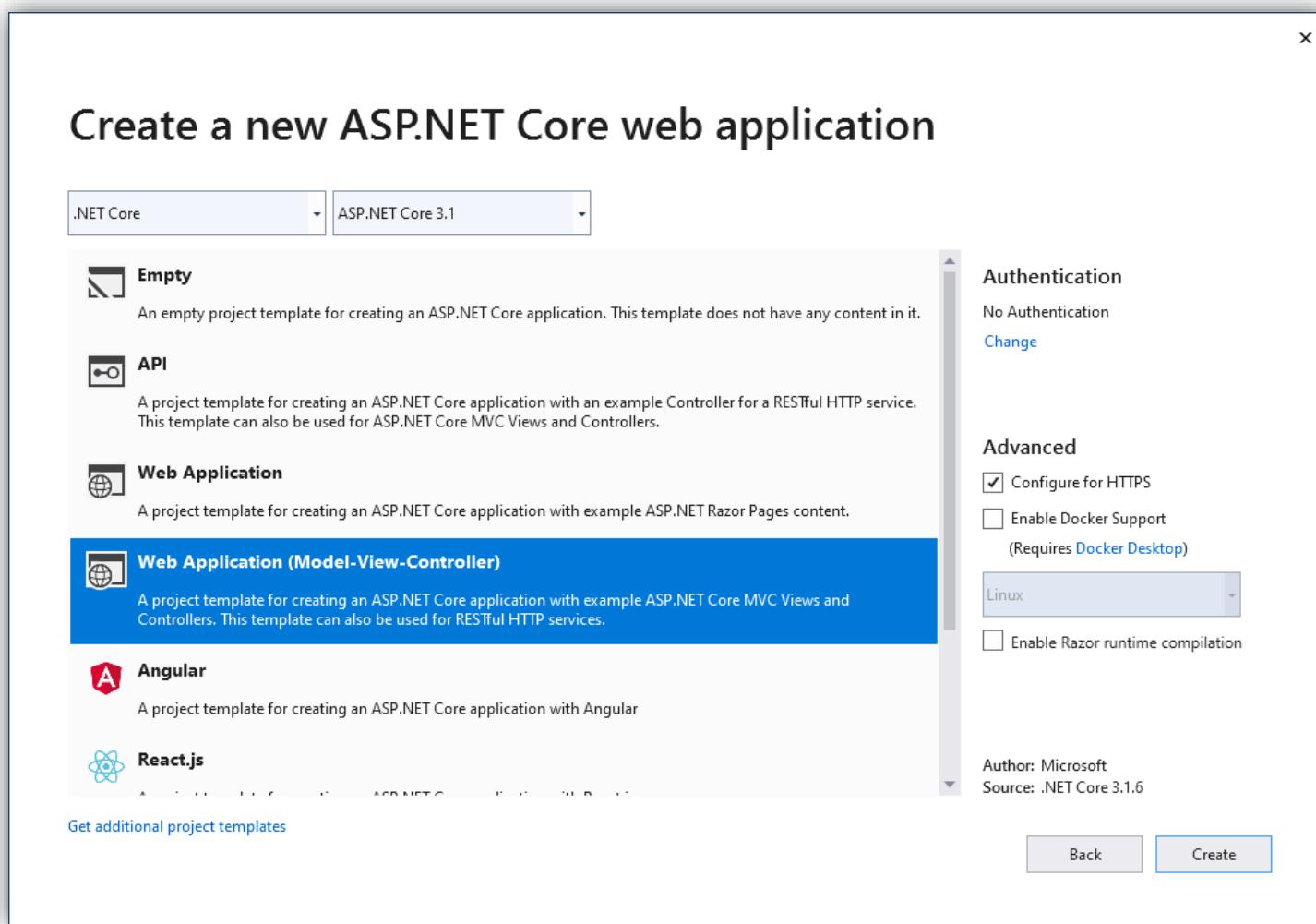
Regola ad-hoc per l'accesso ai Database

Accesso da rete pubblica:
CONSENTITO (con regole ad-hoc)



AZURE DAY

Azure SQL DB: connessione via .NET Core - 1



GITHUB REPO:

- <https://github.com/Darkseal/NETCore-AzureSQL>

NUGET PACKAGES UTILIZZATI:

- Microsoft.EntityFrameworkCore
- Microsoft.EntityFrameworkCore.SqlServer
- Microsoft.Azure.Services.AppAuthentication



Azure SQL DB: connessione via .NET Core - 2

AZURE DAY

/Models/Customer.cs

```
Customer.cs
1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel.DataAnnotations;
4  using System.ComponentModel.DataAnnotations.Schema;
5  using System.Linq;
6  using System.Threading.Tasks;
7
8  namespace NETCore_AzureSQL.Models
9  {
10     [Table("Customer", Schema = "SalesLT")]
11     public class Customer
12     {
13         public int CustomerID { get; set; }
14
15         [Display(Name = "First Name")]
16         [DataType(DataType.Text)]
17         [MaxLength(50)]
18         [Required]
19         public string FirstName { get; set; }
20
21         [Display(Name = "Last Name")]
22         [DataType(DataType.Text)]
23         [MaxLength(50)]
24         [Required]
25         public string LastName { get; set; }
26
27         // TODO: add other properties to map the corresponding database fields
28     }
29
30 }
```

The Solution Explorer shows the project structure:

- Solution 'NETCore-AzureSQL' (1 of 1 project)
 - NETCore-AzureSQL
 - Connected Services
 - Dependencies
 - Properties
 - wwwroot
 - Controllers
 - CustomersController.cs
 - HomeController.cs
 - Data
 - MyDbContext.cs
 - Models
 - Customer.cs
 - ErrorViewModel.cs
 - Views
 - Customers
 - Details.cshtml
 - Index.cshtml
 - Home
 - Index.cshtml
 - Privacy.cshtml
 - Shared
 - _Layout.cshtml
 - _ValidationScriptsPartial.cshtml
 - Error.cshtml
 - _ViewImports.cshtml
 - _ViewStart.cshtml
 - appsettings.json
 - Program.cs
 - Startup.cs



Azure SQL DB: connessione via .NET Core - 3

AZURE DAY

/Data/MyDbContext.cs

The screenshot shows the Microsoft Visual Studio IDE interface. The title bar indicates the project name is "NETCore-AzureSQL". The code editor displays the file "MyDbContext.cs" which contains the following C# code:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Threading.Tasks;
5  using Microsoft.EntityFrameworkCore;
6
7  namespace NETCore_AzureSQL.Data
8  {
9      public class MyDbContext : DbContext
10     {
11         public MyDbContext(DbContextOptions<MyDbContext> options)
12             : base(options)
13         {
14         }
15
16         public DbSet<Models.Customer> Customer { get; set; }
17     }
18 }
```

The Solution Explorer on the right side of the interface lists the project structure. A red arrow points from the Solution Explorer to the "MyDbContext.cs" file in the code editor. The Solution Explorer shows the following files and folders:

- Solution 'NETCore-AzureSQL' (1 of 1 project)
 - NETCore-AzureSQL
 - Connected Services
 - Dependencies
 - Properties
 - wwwroot
 - Controllers
 - CustomersController.cs
 - HomeController.cs
 - Data
 - MyDbContext.cs
 - Models
 - Customer.cs
 - ErrorViewModel.cs
 - Views
 - Customers
 - Details.cshtml
 - Index.cshtml
 - Home
 - Index.cshtml
 - Privacy.cshtml
 - Shared
 - _Layout.cshtml
 - _ValidationScriptsPartial.cshtml
 - Error.cshtml
 - _ViewImports.cshtml
 - _ViewStart.cshtml
 - appsettings.json
 - Program.cs
 - Startup.cs



Azure SQL DB: connessione via .NET Core - 4

AZURE DAY

/Startup.cs

```
13  namespace NETCore_AzureSQL
14  {
15      public class Startup
16      {
17          public Startup(IConfiguration configuration)
18          {
19              Configuration = configuration;
20          }
21
22          public IConfiguration Configuration { get; }
23
24          // This method gets called by the runtime. Use this method to add services to the container.
25          public void ConfigureServices(IServiceCollection services)
26          {
27              services.AddControllersWithViews();
28
29              // Add the DbContext
30              services.AddDbContext<Data.MyDbContext>(options =>
31                  options.UseSqlServer(Configuration.GetConnectionString("MyConnectionString")));
32
33
34          // This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
35          public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
36          {
37              if (env.IsDevelopment())
38              {
39                  app.UseDeveloperExceptionPage();
40              }
41          }
42      }
43  }
```

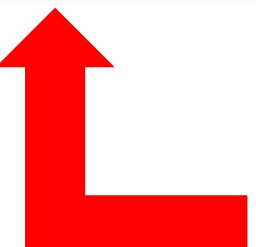


AZURE DAY

Azure SQL DB: connessione via .NET Core - 5

/appsettings.json

```
Schema: https://json.schemastore.org/appsettings
1  {
2    "Logging": {
3      "LogLevel": {
4        "Default": "Information",
5        "Microsoft": "Warning",
6        "Microsoft.Hosting.Lifetime": "Information"
7      }
8    },
9    "AllowedHosts": "*",
10   "ConnectionStrings": {
11     "MyDbConnection": "(PASTE-YOUR-CONNECTION-STRING-HERE)"
12   }
13 }
14 }
```



Microsoft Azure > Home page >

MyTestDB (my-test-server-azure2020/MyTestDB) | Stringhe di connessione

Cerca risorse, servizi e documentazione (G+/)

Power Platform

Power BI (anteprima)

Power Apps (anteprima)

Power Automate (anteprima)

Impostazioni

Configura

Replica geografica

Stringhe di connessione

Sincronizza con altri database

ADO.NET JDBC ODBC PHP Go

ADO.NET (autenticazione SQL)

```
Server=tcp:my-test-server-azure2020.database.windows.net,1433;Initial Catalog=MyTestDB;Persist Security Info=False;User ID=MyPersonalAdmin;Password=(your_password);MultipleActiveResultSets=False;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;
```

Cerca (CTRL+)

Scarica il driver ADO.NET per SQL server



AZURE DAY

Azure SQL DB: connessione via .NET Core - 6

/Controllers/CustomerController.cs

```
13  1 reference
14  public class CustomersController : Controller
15  {
16      private readonly MyDbContext _context;
17
18      O references
19      public CustomersController(MyDbContext context)
20      {
21          _context = context;
22      }
23
24      // GET: Customers
25      O references
26      public async Task<IActionResult> Index()
27      {
28          return View(await _context.Customer.ToListAsync());
29      }
30
31      // GET: Customers/Details/5
32      O references
33      public async Task<IActionResult> Details(int? id)
34      {
35          if (id == null)
36          {
37              return NotFound();
38          }
39
40          var customer = await _context.Customer
41              .FirstOrDefaultAsync(m => m.CustomerID == id);
42          if (customer == null)
43          {
44              return NotFound();
45          }
46
47          return View(customer);
48      }
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
139
140
141
142
143
144
145
```

/Views/Customers/Index.html

```
1  @model IEnumerable<NETCore_AzureSQL.Models.Customer>
2
3  @{
4      ViewData["Title"] = "Index";
5  }
6
7  <h1>Customers</h1>
8
9  <p><a href="#">asp-action="Create">Create New</a></p>
10 <table class="table">
11     <thead>
12         <tr>
13             <th>@Html.DisplayNameFor(model => model.CustomerID)</th>
14             <th>@Html.DisplayNameFor(model => model.FirstName)</th>
15             <th>@Html.DisplayNameFor(model => model.LastName)</th>
16         </tr>
17     </thead>
18     <tbody>
19         @foreach (var item in Model) {
20             <tr>
21                 <td>@Html.DisplayFor(modelItem => item.CustomerID)</td>
22                 <td>@Html.DisplayFor(modelItem => item.FirstName)</td>
23                 <td>@Html.DisplayFor(modelItem => item.LastName)</td>
24             </tr>
25         }
26     </tbody>
27 </table>
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
```

/Views/Customers/Details.html

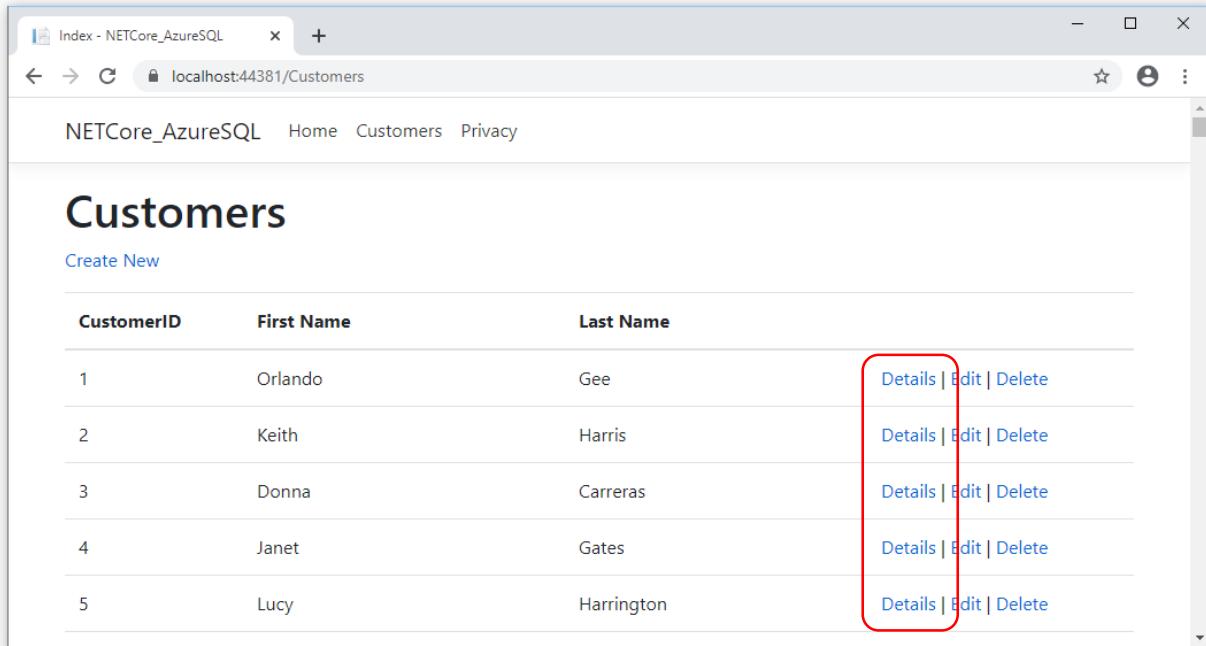
```
1  @model NETCore_AzureSQL.Models.Customer
2
3  @{
4      ViewData["Title"] = "Details";
5  }
6
7  <h1>Customer Details</h1>
8
9  <div>
10     <dl class="row">
11         <dt class = "col-sm-2">@Html.DisplayNameFor(model => model.CustomerID)</dt>
12         <dd class = "col-sm-10">@Html.DisplayFor(model => model.CustomerID)</dd>
13     </dl>
14     <dt class = "col-sm-2">@Html.DisplayNameFor(model => model.FirstName)</dt>
15     <dd class = "col-sm-10">@Html.DisplayFor(model => model.FirstName)</dd>
16     <dt class = "col-sm-2">@Html.DisplayNameFor(model => model.LastName)</dt>
17     <dd class = "col-sm-10">@Html.DisplayFor(model => model.LastName)</dd>
18
19     <div>
20         <dt class = "col-sm-2"><a href="#">asp-action="Edit" asp-route-id="@Model.CustomerID">Edit</a> |</dt>
21         <dt class = "col-sm-2"><a href="#">asp-action="Delete" asp-route-id="@Model.CustomerID">Delete</a></dt>
22     </div>
23
24
25
26
27
28
29
30
31     <div>
32         <dt class = "col-sm-2"><a href="#">asp-action="Edit" asp-route-id="@Model.CustomerID">Edit</a></dt>
33         <dt class = "col-sm-2"><a href="#">asp-action="Index">Back to List</a></dt>
34     </div>
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
109
110
111
112
113
114
115
116
117
118
119
119
120
121
122
123
124
125
126
127
128
129
129
130
131
132
133
134
135
136
137
138
139
139
140
141
142
143
144
145
```



AZURE DAY

Azure SQL DB: connessione via .NET Core - 7

Master view



Index - NETCore_AzureSQL

localhost:44381/Customers

NETCore_AzureSQL Home Customers Privacy

Customers

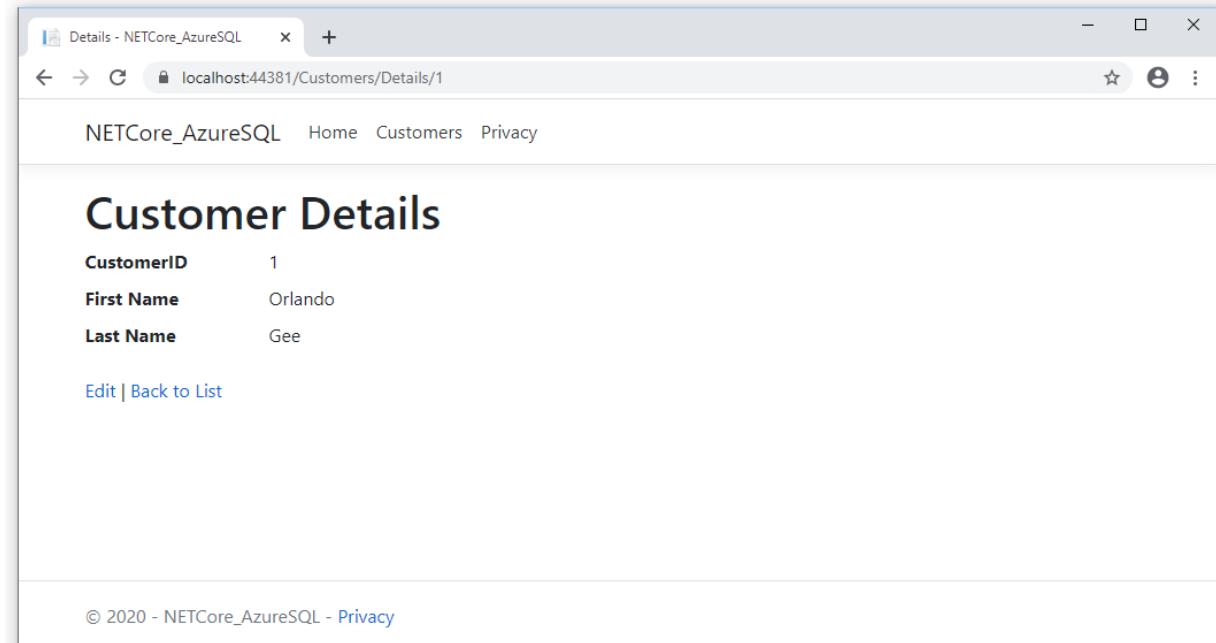
Create New

CustomerID	First Name	Last Name	
1	Orlando	Gee	Details Edit Delete
2	Keith	Harris	Details Edit Delete
3	Donna	Carreras	Details Edit Delete
4	Janet	Gates	Details Edit Delete
5	Lucy	Harrington	Details Edit Delete



https://localhost:<random_port>
(development environment)

Detail view



Details - NETCore_AzureSQL

localhost:44381/Customers/Details/1

NETCore_AzureSQL Home Customers Privacy

Customer Details

CustomerID	1
First Name	Orlando
Last Name	Gee

[Edit](#) | [Back to List](#)

© 2020 - NETCore_AzureSQL - [Privacy](#)



AZURE DAY

Azure SQL DB: connessione via .NET Core - 8

METODO ALTERNATIVO

1 – Creare un nuovo utente nella Azure Active Directory

The screenshot shows the Microsoft Azure portal interface for creating a new user. At the top, there's a navigation bar with the Microsoft Azure logo, a search bar, and various icons. Below it, the breadcrumb navigation shows 'Home page > Utenti | Tutti gli utenti >'. The main title is 'Nuovo utente' (New User), followed by the name 'Ryadel'. A note says 'Sono disponibili commenti?' (Comments are available). There are two options: 'Crea utente' (Create user) and 'Invita l'utente' (Invite user). The 'Crea utente' option is selected. It describes creating a new user in the organization and provides an example email address: 'alice@ryadel.com'. A link 'Si vogliono creare in blocco utenti' (If you want to create multiple users) is provided. The 'Invita l'utente' option describes inviting a guest user and sending an invitation. A link 'Si vogliono invitare utenti guest in blocco' (If you want to invite multiple guest users) is provided. Below these options, there's a section titled 'Suggerimenti per la scelta' (Suggestions for selection) which lists 'Identità' (Identity), 'Nome utente' (User name), 'Nome' (Name), and 'Nome' (Name again). The 'Nome utente' field contains 'MyDBA' and 'ryadel.com', with a note 'Il nome di dominio necessario non è mostrato qui' (The required domain name is not shown here). The 'Nome' field contains 'My Database|Administrator'.



AZURE DAY

Azure SQL DB: connessione via .NET Core - 9

METODO ALTERNATIVO

2 – Aggiungere l’utente appena creato come amministratore del Database Server Logico

The screenshot shows the Microsoft Azure portal interface for managing Azure SQL databases. On the left, the 'Azure SQL' blade is open, displaying a list of databases: 'my-test-server-azure2020', 'MyTestDB (my-test-server-azure2020...)', 'netcore3-angular8', and 'WorldCities (netcore3-angular8/WorldC...'. The database 'my-test-server-azure2020' is highlighted with a red box. On the right, the detailed view for 'my-test-server-azure2020' is shown, titled 'my-test-server-azure2020 | amministratore di Active Directory'. The top navigation bar includes 'Microsoft Azure', a search bar, and various icons. Below the title, there's a 'Panoramica' section with a 'Cerca (CTRL+/' button. In the top right, there are buttons for 'Imposta amministratore' (highlighted with a red box), 'Rimuovi amministratore', and 'Salva'. A descriptive text states: 'Azure Active Directory authentication permette di gestire in modo centralizzato le identità e l'accesso a Azure SQL Database V12.' At the bottom, it shows 'amministratore di Active Directory' (status: Nessun amministratore di Active Directory) and a 'Altre informazioni' link.

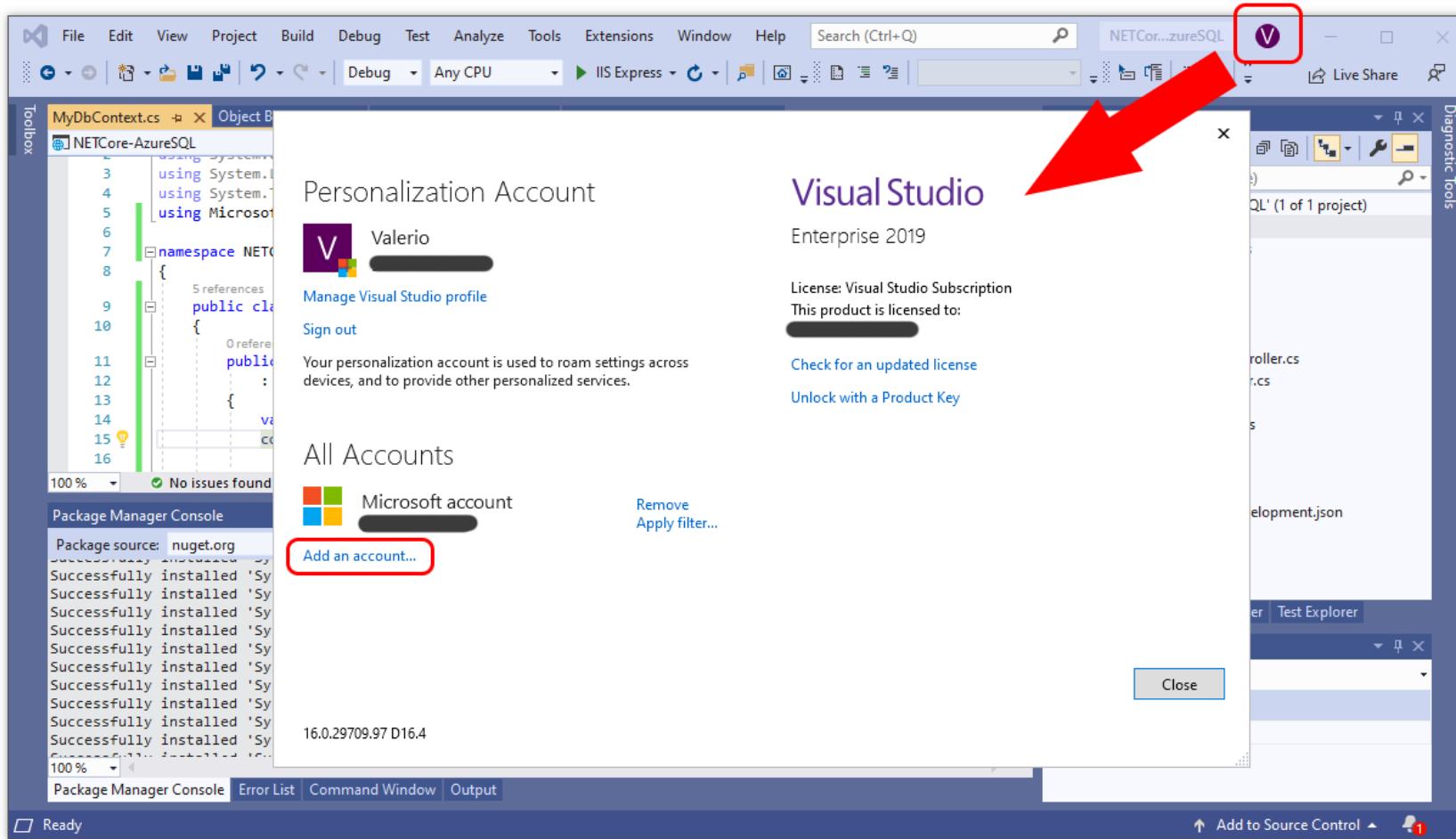


AZURE DAY

Azure SQL DB: connessione via .NET Core - 10

METODO ALTERNATIVO

3 – Aggiungere l’utente Azure AD appena creato a Visual Studio



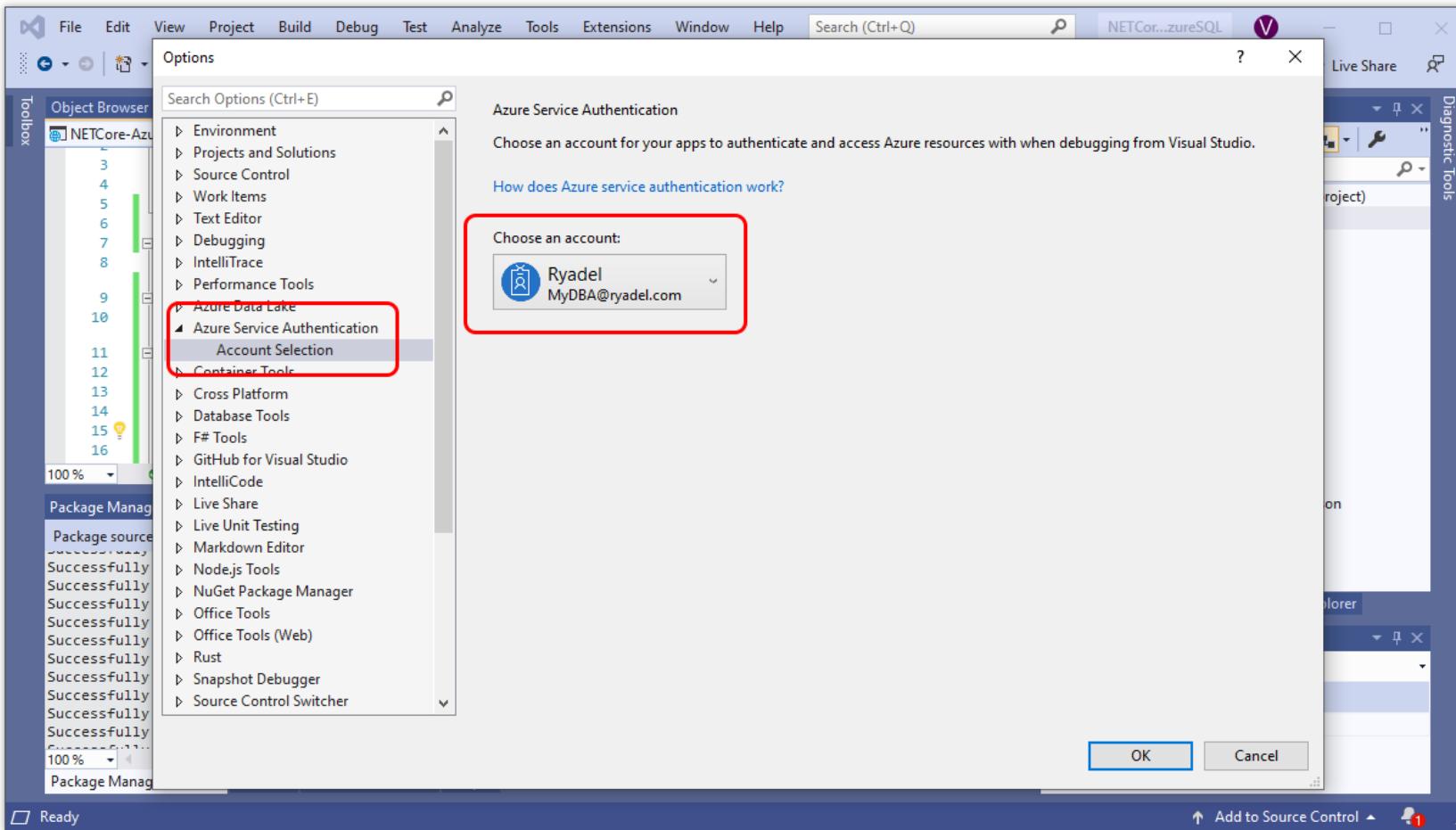


AZURE DAY

Azure SQL DB: connessione via .NET Core - 11

METODO ALTERNATIVO

4 – Impostare l’utente Azure AD come account di accesso alle risorse Azure per il debug





AZURE DAY

Azure SQL DB: connessione via .NET Core - 12

METODO ALTERNATIVO

5 – Aggiungere e configurare il TokenProvider per servizi Azure all'interno del DbContext

The screenshot shows the Visual Studio IDE interface with the following details:

- MyDbContext.cs:** The code defines a `MyDbContext` class that inherits from `DbContext`. It uses `DbContextOptions<MyDbContext>` for configuration. A specific constructor is shown where the `AccessToken` is obtained via the `AzureServiceTokenProvider`.

```
// Add and configure AzureServiceTokenProvider to securely access the SQL Database
var conn = (Microsoft.Data.SqlClient.SqlConnection)Database.GetConnectionString();
conn.AccessToken = (new Microsoft.Azure.Services.AppAuthentication.AzureServiceTokenProvider())
    .GetAccessTokenAsync("https://database.windows.net/").Result;
```

- Solution Explorer:** Shows the project structure for `NETCore-AzureSQL`, including `Connected Services`, `Dependencies`, `Properties`, `wwwroot`, `Controllers`, `Data` (containing `MyDbContext.cs` and `Models`), `Views`, `appsettings.json`, `Program.cs`, and `Startup.cs`.
- Properties:** The `MyDbContext.cs` file properties are set to `Copy to Output` and `Do not copy`.
- Command Window:** Empty.
- Package Manager Console:** Ready.
- Error List:** No issues found.
- Output:** No output.



AZURE DAY

Azure SQL DB: connessione via .NET Core - 13

METODO ALTERNATIVO

6 – Rimuovere le credenziali di autenticazione dalla *Connection String* nel file *appsettings.json*

```
1 {
2     "Logging": {
3         "LogLevel": {
4             "Default": "Information",
5             "Microsoft": "Warning",
6             "Microsoft.Hosting.Lifetime": "Information"
7         }
8     },
9     "AllowedHosts": "*",
10    "ConnectionStrings": {
11        "MyDbConnection": "Server=tcp:my-test-server-azure2020.database.windows.net,1433;Initial Catalog=MyTestDB;"}
12    }
13 }
```

NUOVA CONNECTION STRING:

Server=tcp:my-test-server-azure2020.database.windows.net,1433;Initial Catalog=MyTestDB;



AZURE DAY

Azure SQL DB: .NET Core App - Deploy - 1

1 – Creare un Servizio App su MS Azure





AZURE DAY

Azure SQL DB: .NET Core App - Deploy - 2

2 – Attivare l'identità gestita (assegnata dal sistema)

Screenshot of the Microsoft Azure portal showing the configuration of an Azure App Service identity.

The page title is **NetCore-AzureSQL | Identità**. The left sidebar shows navigation links: Home page, Slot di distribuzione, Centro distribuzione, Impostazioni, Configurazione, Autenticazione/Autorizzazione, Application Insights, **Identità** (highlighted with a red box), Backup, Domini personalizzati, Impostazioni TLS/SSL, and Rete.

The main content area shows two tabs: **Assegnata dal sistema** (selected) and **Assegnata dall'utente**. A descriptive text explains that a system-assigned managed identity allows Azure resources to authenticate with cloud services like Azure Key Vault without storing credentials in the code. It notes that the identity's lifecycle depends on the resource's lifecycle and provides a link to "Altre informazioni sulle identità gestite".

At the bottom, there are buttons for **Salva** (Save), **Rimuovi** (Delete), **Aggiorna** (Update), and a comment section. The status section shows the current state as **Disattivata** (Disabled) and a button to **Attivata** (Enable).

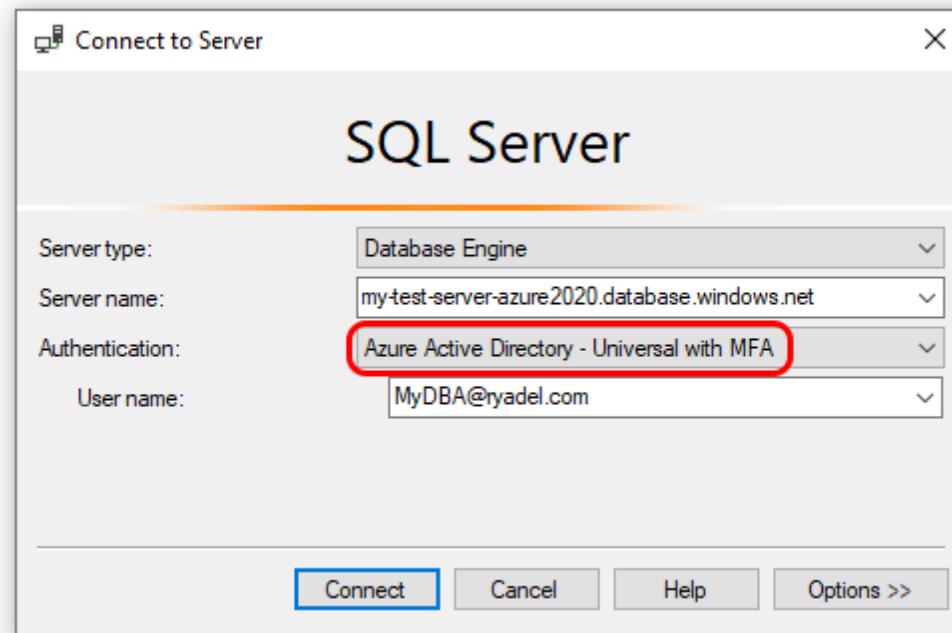


AZURE DAY

Azure SQL DB: .NET Core App - Deploy - 3

3 – Accedere al Database Server logico tramite SSMS utilizzando una delle seguenti modalità di autenticazione:

Azure Active Directory – Universal with MFA
Azure Active Directory - Password



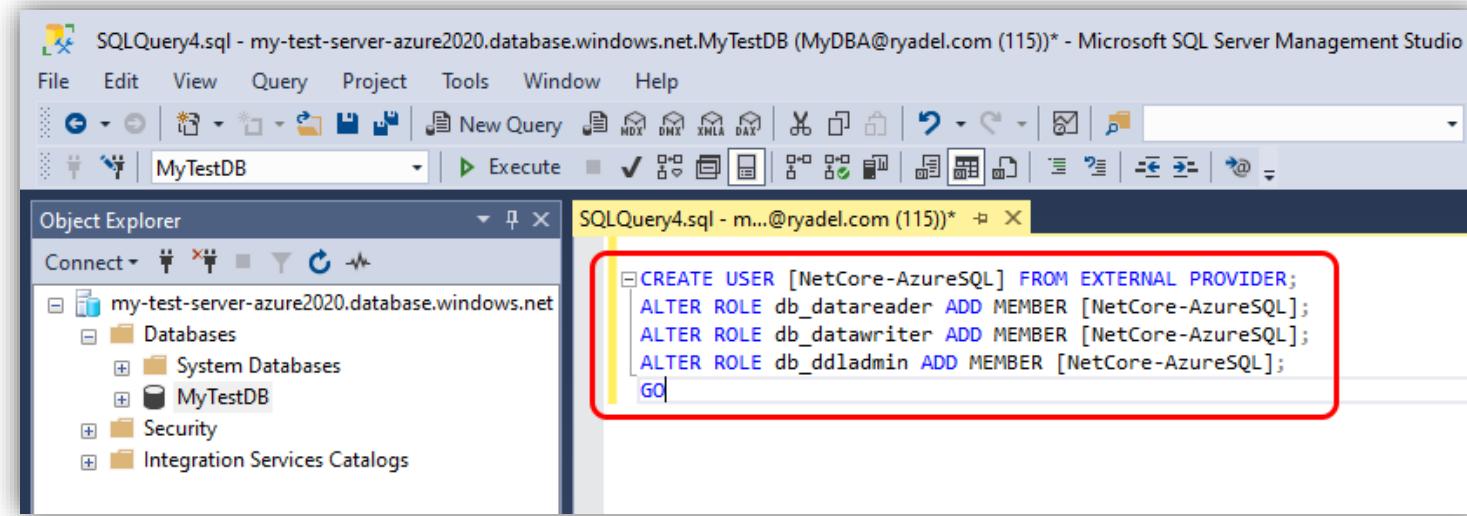
Inserire l'indirizzo e-mail
dell'utente creato all'interno
di Azure Active Directory



AZURE DAY

Azure SQL DB: .NET Core App - Deploy - 4

4 – Autorizzare l'accesso dell'identità gestita sul Database con la seguente query:



```
CREATE USER [NetCore-AzureSQL] FROM EXTERNAL PROVIDER;
ALTER ROLE db_datareader ADD MEMBER [NetCore-AzureSQL];
ALTER ROLE db_datawriter ADD MEMBER [NetCore-AzureSQL];
ALTER ROLE db_ddladmin ADD MEMBER [NetCore-AzureSQL];
GO
```

```
CREATE USER [NetCore-AzureSQL] FROM EXTERNAL PROVIDER;
ALTER ROLE db_datareader ADD MEMBER [NetCore-AzureSQL];
ALTER ROLE db_datawriter ADD MEMBER [NetCore-AzureSQL];
ALTER ROLE db_ddladmin ADD MEMBER [NetCore-AzureSQL];
GO
```

NOTE IMPORTANTI:

- In caso di identità gestita, il nome dell'utente da creare è lo stesso del servizio App
- Per creare l'utente è necessario accedere con un utente creato su Azure Active Directory



AZURE DAY

Azure SQL DB: .NET Core App - Deploy - 5

5 – Abilitare l’accesso al Database Server logico da parte delle risorse Azure

Microsoft Azure Cerca risorse, servizi e documentazione (G+/-)

Home page > my-test-server-azure2020 | Firewall e reti virtuali

SQL Server

Cerca (CTRL+/) Salva Rimuovi Aggiungi IP client

L'impostazione su Sì consente le connessioni solo tramite l'endpoint privato approvato e disabilita tutte le regole del firewall esistenti. [Altre informazioni.](#)

Versione minima TLS > 1.0 > 1.1 > 1.2

Si sta impostando la proprietà Versione minima TLS per tutti i database Database SQL e SQL Data Warehouse associati al server. Eventuali tentativi di accesso dai client che usano una versione TLS inferiore alla Versione minima TLS verranno rifiutati.

Criteri di connessione Predefinito Proxy Reindirizzamento

Consenti alle risorse e ai servizi di Azure di accedere a questo server **Sì** No

Le connessioni dagli indirizzi IP specificati sotto forniscono l'accesso a tutti i database in my-test-server-azure2020.

Indirizzo IP client 62.11.38.153

Blocchi Esporta modello Sicurezza Sicurezza dei dati avanzata Controllo Firewall e reti virtuali Connessioni endpoint privato Transparent Data Encryption Prestazioni intelligenti Ottimizzazione automatica Raccomandazioni Monitoraggio Log



AZURE DAY

Azure SQL DB: .NET Core App - Deploy - 6

6 – Configurare il deploy della Web Application su Azure

The screenshot illustrates the deployment process for a .NET Core application named 'NETCore-AzureSQL'. In the Solution Explorer, the project is selected. A red box highlights the 'Publish...' option in the context menu, which is then expanded to show options like 'Overview' and 'Scope to This'. A large red arrow points from the 'Publish...' step to the 'Pick a publish target' dialog.

Pick a publish target

- App Service** (highlighted with a red box)
- App Service Linux
- Azure Virtual Machines
- IIS, FTP, etc
- Folder

Azure App Service
Fully managed, and highly scalable cloud environment

Create New
 Select Existing (highlighted with a red box)

Advanced...

Import Profile... Create Profile Cancel



AZURE DAY

Azure SQL DB: .NET Core App - Deploy - 7

7 – Effettuare il deploy della Web Application su Azure

The screenshot illustrates the deployment process for a .NET Core application to Azure. On the left, the Azure portal's App Service blade is shown, displaying basic settings like Subscription (Microsoft MVP) and View (Resource group). A red arrow points from this blade to the right-hand side of the screen, which shows the Visual Studio interface.

In the Visual Studio interface, the project "NETCore-AzureSQL" is open. The "Publish" tab is selected in the "Web Publish Activity" section. A publish profile named "NetCore-AzureSQL - Web Deploy" is listed. The "Publish" button, located at the top right of the "Publish" section, is highlighted with a red border. The "Actions" sidebar on the right provides links for previewing changes, managing in Cloud Explorer, editing app service settings, attaching a snapshot debugger, and opening a troubleshooting guide.

Key details visible in the "Publish" section:

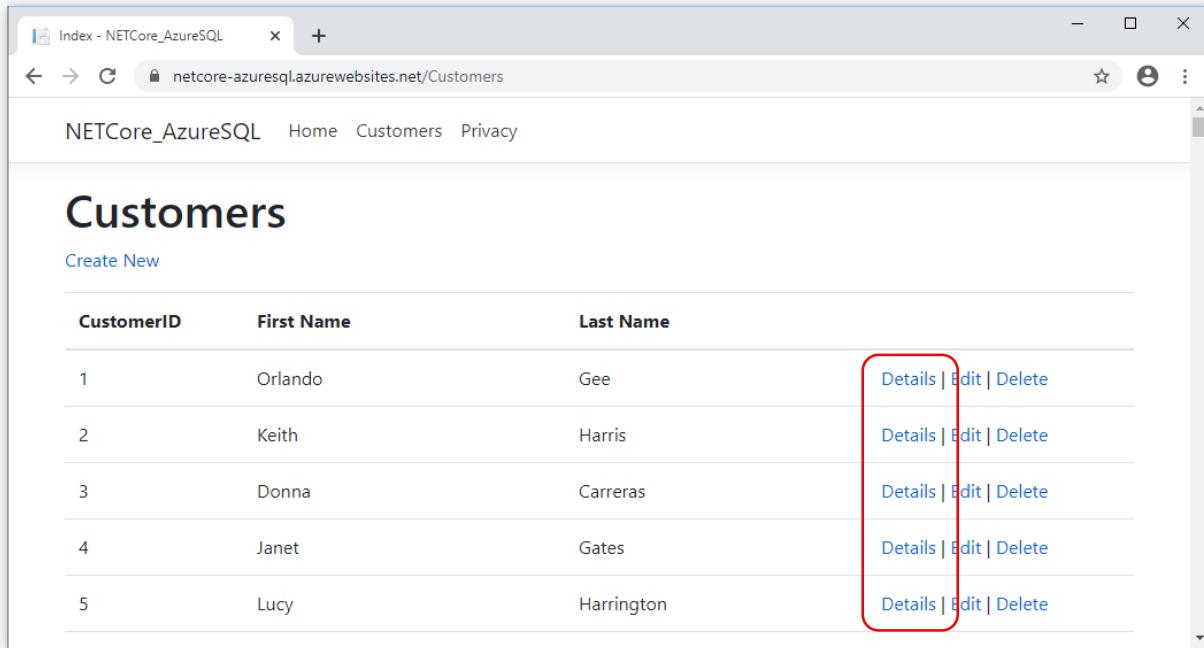
- Overview:** Deploy your app to a folder, IIS, Azure, or another destination. [More info](#)
- Service References:** None listed.
- Publish:** NetCore-AzureSQL - Web Deploy
- Summary:**
 - Site URL: <http://netcore-azuresql.azurewebsites.net>
 - Resource Group: AzureDay2020
 - Configuration: Release
 - Target Framework: netcoreapp3.1
 - Deployment Mode: Framework-Dependent
- Actions:**
 - Preview changes
 - Manage in Cloud Explorer
 - Edit Azure App Service settings
 - Attach Snapshot Debugger
 - Open troubleshooting guide
- Dependencies:** No dependencies currently configured, please click 'Add' to connect to additional services.



AZURE DAY

Azure SQL DB: .NET Core App - Test su WWW

Master view

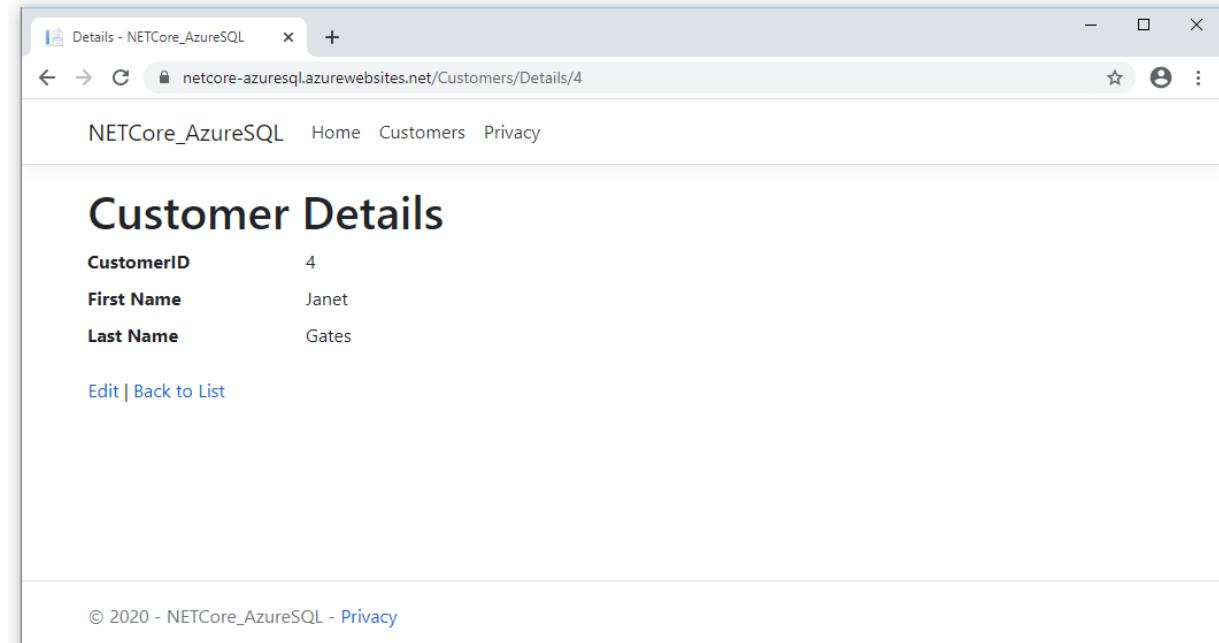


CustomerID	First Name	Last Name	
1	Orlando	Gee	Details Edit Delete
2	Keith	Harris	Details Edit Delete
3	Donna	Carreras	Details Edit Delete
4	Janet	Gates	Details Edit Delete
5	Lucy	Harrington	Details Edit Delete



<https://netcore-azuresql.azurewebsites.net>
(production environment)

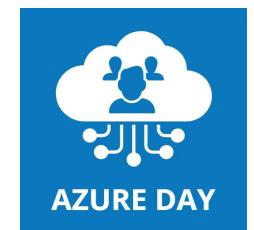
Detail view



CustomerID	4
First Name	Janet
Last Name	Gates
Edit Back to List	
© 2020 - NETCore_AzureSQL - Privacy	



QUESTION TIME



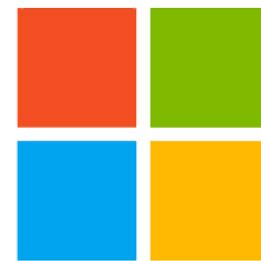


AZURE DAY

Thank You!!!



Thanks to



Microsoft



avanade



Capgemini

plexin.
CON NOI È SEMPLICE

FABRICARIS

LLOBRA

6TH
SENSE



PLURALSIGHT



DotNetCode.IT