Programming with the .NET SDK



Leonard Lobel
CTO, SLEEK TECHNOLOGIES
lennilobel.wordpress.com

Client Development

Build web-scale applications

Uses REST/HTTP

Platform SDKs

.NET / .NET Core
Java
Node.js
Python

...



Create a CosmosClient instance

Supply connection information (endpoint and key)

Task Parallel Library (TPL)

Simplified asynchronous programming
Use async/await keywords with Task objects

Invoke methods to access resources

Create, modify, and delete resources
Use POCOs or dynamics for document objects



Synchronous code

```
private void Main()
{
    DoSomething();
}

private void DoSomething()
{
    // do some work
}
```

Asynchronous code

```
private async void Main()
{
    await DoSomething();
}

private async Task DoSomething()
{
    // do some asynchronous work
}
```

access resources

delete resources for document objects

Task Parallel Library (TPL)

Simplified asynchronous programming
Use async/await keywords with Task objects



Synchronous code

```
private void Main()
{
    var result = GetSomething();
}

private string GetSomething()
{
    // do some work
    return "Hello";
}
```

Asynchronous code

```
private async void Main()
{
    var result = await GetSomething();
}

private async Task<string> GetSomething()
{
    // do some asynchronous work
    return "Hello";
}
```

access resources

delete resources for document objects

Task Parallel Library (TPL)

Simplified asynchronous programming
Use async/await keywords with Task objects



Create a CosmosClient instance

Supply connection information (endpoint and key)

Task Parallel Library (TPL)

Simplified asynchronous programming
Use async/await keywords with Task objects

Invoke methods to access resources

Create, modify, and delete resources
Use POCOs or dynamics for document objects

LINQ provider

Automatically translates LINQ queries to SQL





Getting started with the .NET SDK





Working with databases





Working with containers





Creating documents





Querying for documents





Stateful and stateless paging





Streaming iterators





Querying with LINQ



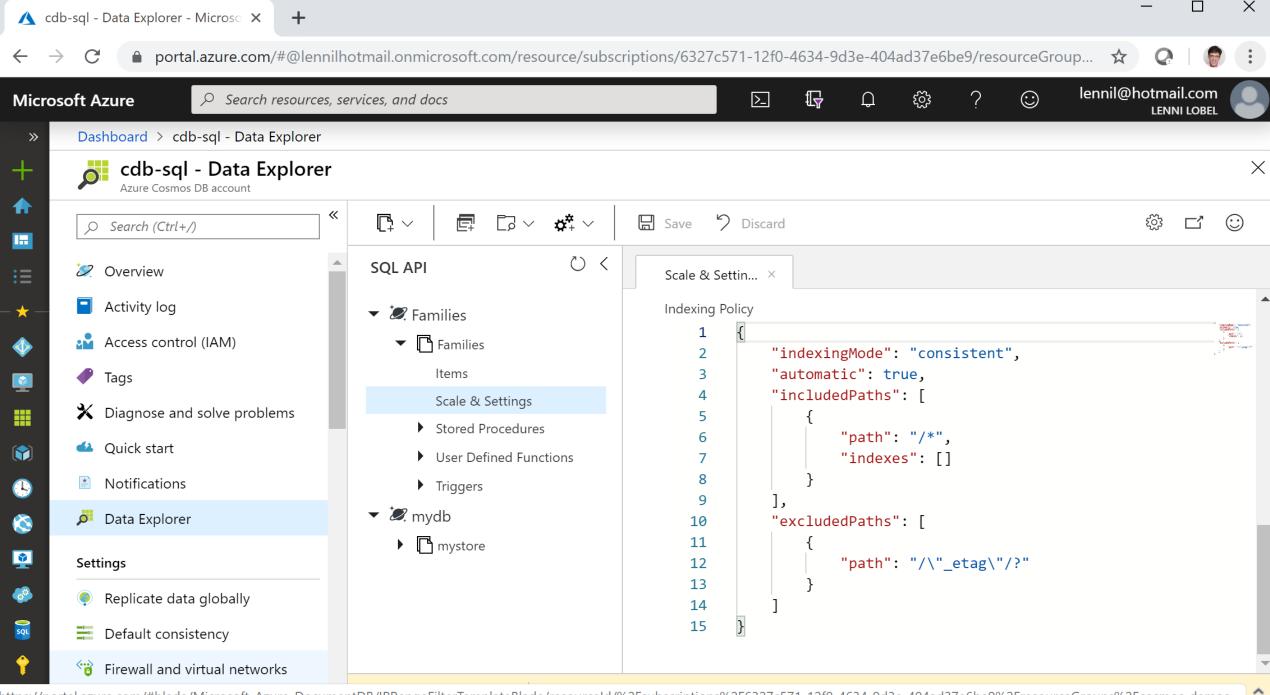


Replacing and deleting documents



Indexing Policies





Indexing Policies

Range index

Equality, range, ORDER BY

Selective indexing

Exclude selected property paths

Spatial index

GeoJSON points, line strings, and polygons

Composite indexes

Enable ORDER BY on multiple properties





Custom indexing



Summary



.NET SDK

- CosmosClient
- Databases
- Containers
- Create documents
- SQL & LINQ queries
- Replace/delete documents
- Stateful and stateless paging
- Streaming iterators
- Custom indexing policies

