**TABLE STORAGE – SE GUARDA EN Cosmos DB**

tableStorage es una buena opcion para nuevo proyectos donde necesitemos almacenar informacion no tan relacional.

**1) Crear un recurso de Cosmos DB**

**=============================**

Click en Add

Buscar Azure Cosmos DB

Elegir el grupo de recurso,

Nombrar la cuenta de CosmosDB

Elegir la location

Las opciones Geo-redundancy y multi-region es recomendada dejarlas disable

Table Storage con CosmosDB

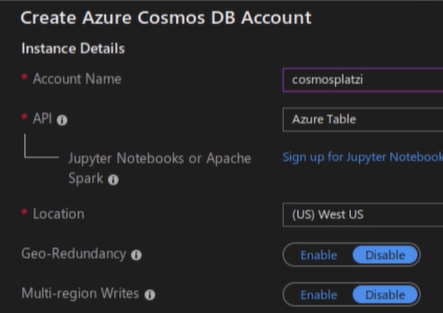
Para ello crear un recurso de Cosmos DB

En el Portal, en la cuenta,

Click en Add (+)

Buscar : Azure Cosmos DB

Create



Al crearlo asigne

- Grupo de recurso,

- Nombre: la cuenta de CosmosDB

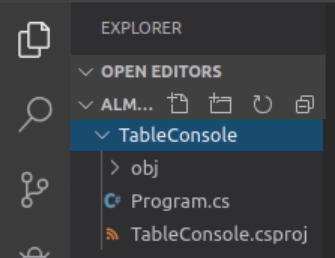
- Location

- Las opciones Geo-redundancy y multi-region es recomendada dejarlas disable

**2) .Net Core**

==========

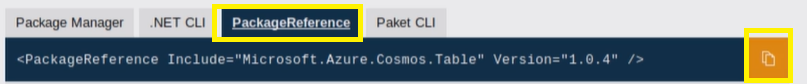
dotnet new console -n TableConsole



Agregue la seguridad, appsettings.json

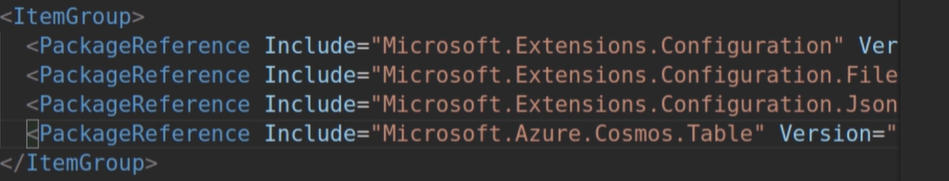
Instalar paquete: Microsoft.Azure.Cosmos.Table

Revise version no beta



**<PackageReference Include=“Microsoft.Azure.Cosmos.Table” Version=“1.0.7” />**

Y en vez de correr voy al archive <projecto>.csproj y agrego la referencia



En archivo Programa.cs

using System.IO;

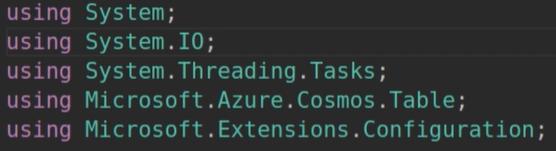
using System;

using System.Threading.Tasks;

using Microsoft.Azure.Cosmos.Table;

using Microsoft.Extensions.Configuration;

using Microsoft.Extensions.Configuration.FileExtensions;



using Microsoft.Extensions.Configuration;

using Microsoft.Extensions.Configuration.Json;

static void Main(string[] args)

{

var builder = new ConfigurationBuilder()

.SetBasePath(Directory.GetCurrentDIrectory() )

.AddJsonFIle("appsettings.json");

IConfiguration config = new ConfigurationBuilder()

.AddJsonFile("appsettings.json",true, true)

.Build();

//En portal tome la cadena de string de Cosmos, del creado y lo asigna a appsettings.json

//Llama a la funcion con

CreateTableAsync("test01", config["connectionString"]).Wait();

}

private static async Task<CloudTable> CreateTableAsync(string TableName, string Conn)

{

CloudTableClient tableClient = storageAccount.CreateCloudTableClient(new TableClientConfiguration());

CloudTable table = tableClient.GetTableReference(tableName);

if (await table.CreateIfNotExistsAsync())

Console.Writeline("Tabla creada {0} ",tableName);

else

Console.Writeline("Tabla ya existe: {0} ",tableName);

return table;

}

Connection string

