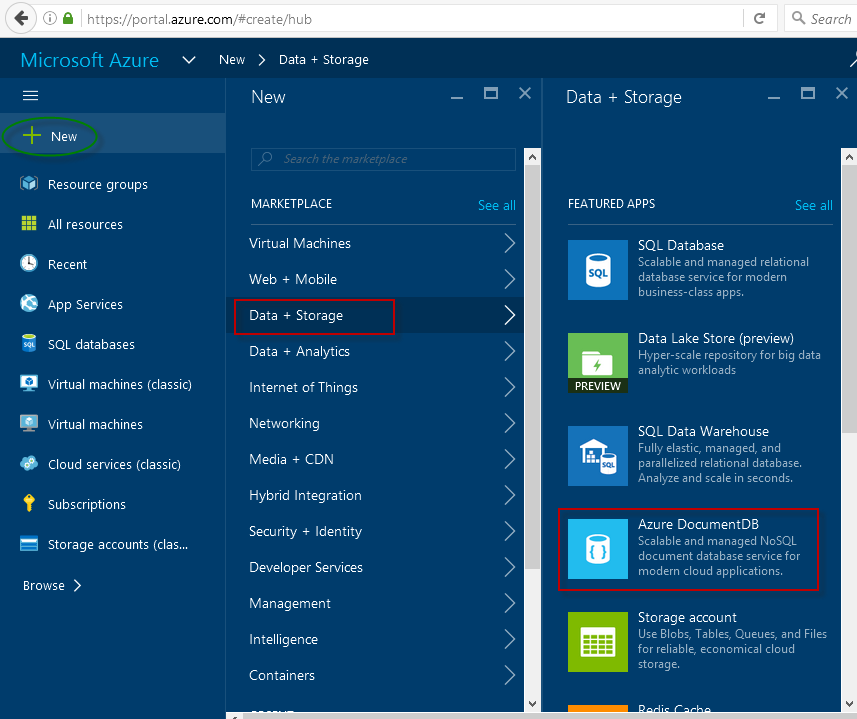
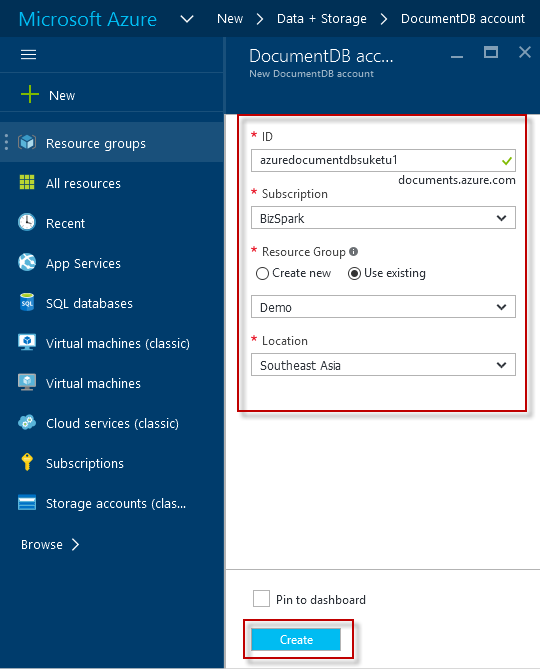
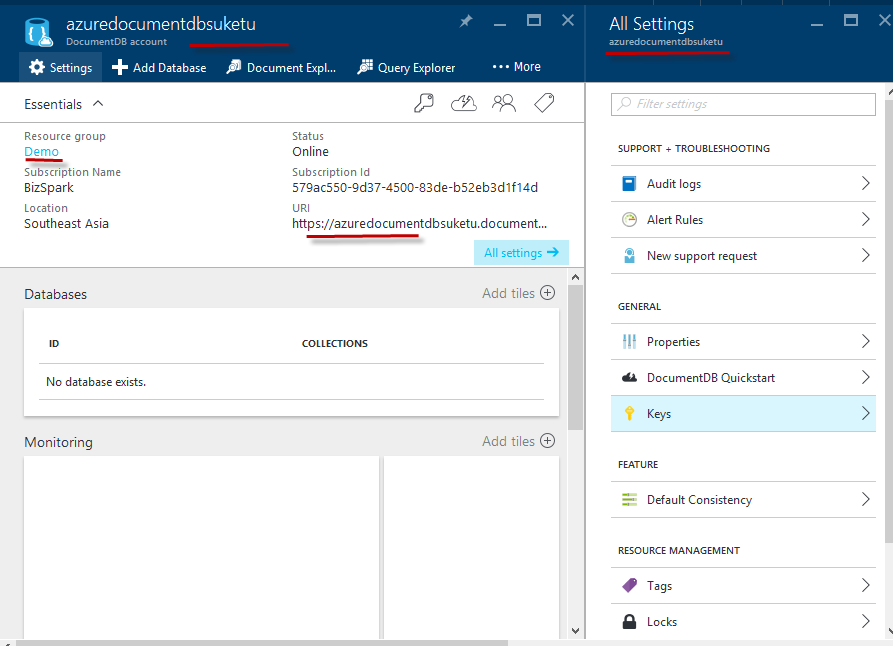
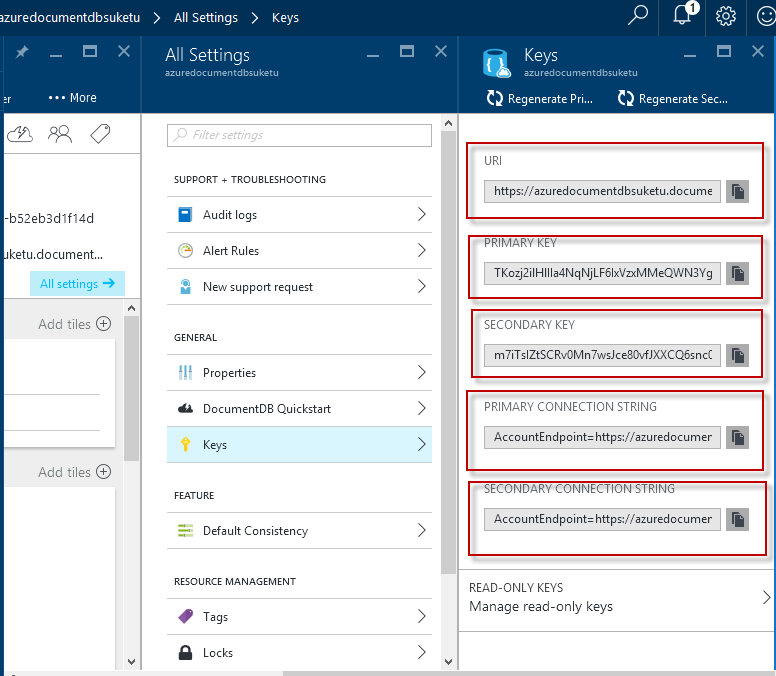
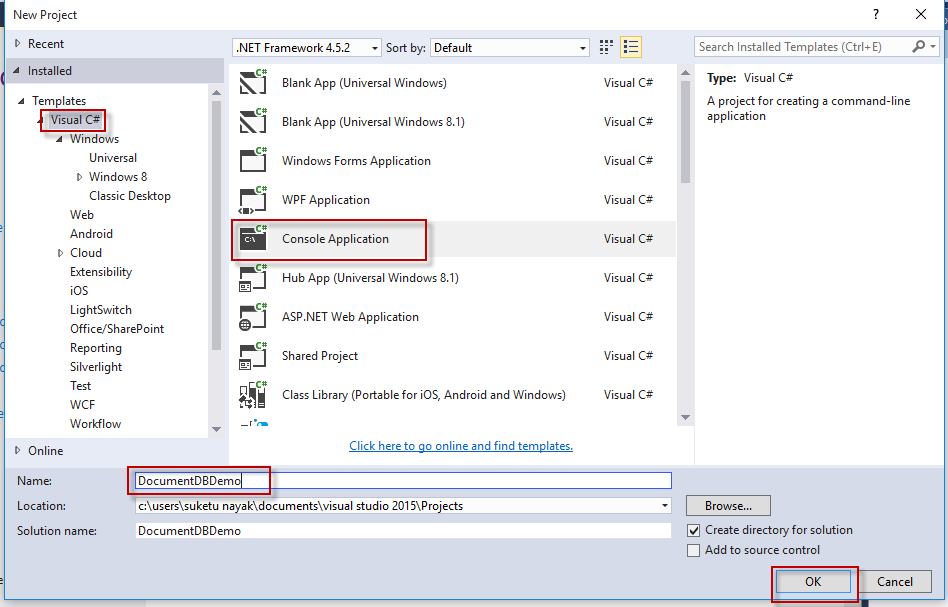
DocumentDB in Azure

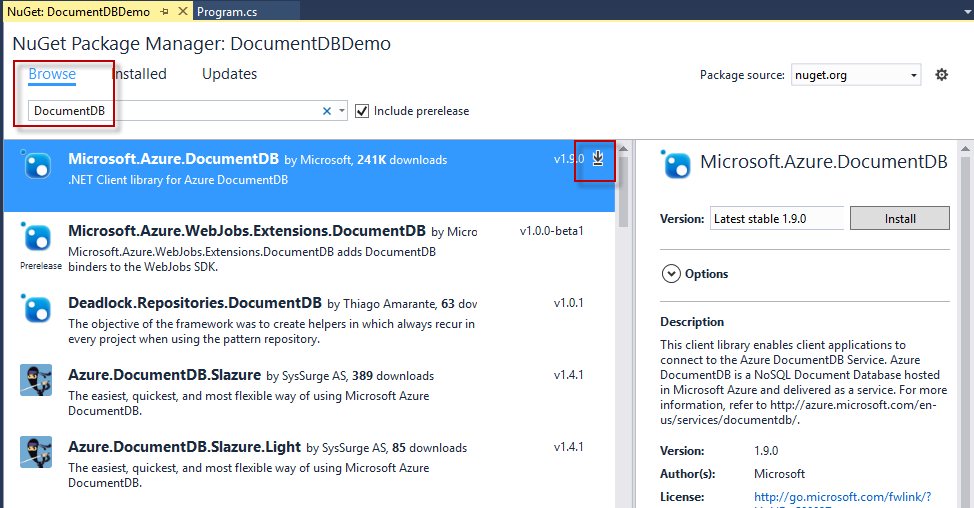


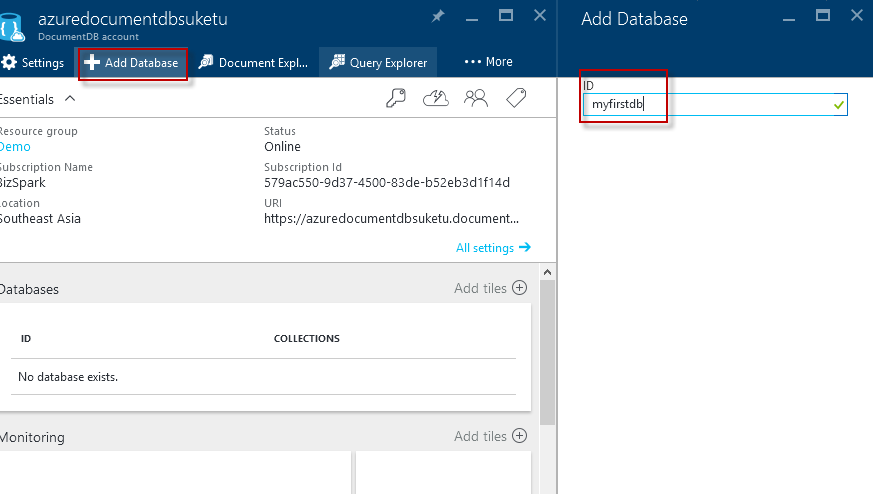












**Program.cs File**

using Microsoft.Azure.Documents;

using Microsoft.Azure.Documents.Client;

using Microsoft.Azure.Documents.Linq;

using Newtonsoft.Json;

namespace DocumentDBDemo {

class Program {

private const string EndpointUrl = "https://azuredocdbdemo.documents.azure.com:443/";

private const string AuthorizationKey = "BBhjI0gxdVPdDbS4diTjdloJq7Fp4L5RO/

StTt6UtEufDM78qM2CtBZWbyVwFPSJIm8AcfDu2O+AfV T+TYUnBQ==";

static void Main(string[] args) {

try {

CreateDocumentClient().Wait();

} catch (Exception e) {

Exception baseException = e.GetBaseException();

Console.WriteLine("Error: {0}, Message: {1}", e.Message, baseException.Message);

}

Console.ReadKey();

}

private static async Task CreateDocumentClient() {

// Create a new instance of the DocumentClient

var client = new DocumentClient(new Uri(EndpointUrl), AuthorizationKey);

{

// Create a Database for DocumentDB Using .Net SDK

await CreateDatabase(client);

// TO LIST DATABASES

GetDatabases(client);

// TO DELETE DATABASE

GetDatabases(client);

await DeleteDatabase(client);

GetDatabases(client);

}

}

}

}

Create a Database for DocumentDB Using .Net SDK

Add this in Program.cs

private async static Task CreateDatabase(DocumentClient client)

{

Console.WriteLine();

Console.WriteLine("\*\*\*\*\*\*\*\* Create Database \*\*\*\*\*\*\*");

var databaseDefinition = new Database { Id = "mynewdb" };

var result = await client.CreateDatabaseAsync(databaseDefinition);

var database = result.Resource;

Console.WriteLine(" Database Id: {0}; Rid: {1}", database.Id, database.ResourceId);

Console.WriteLine("\*\*\*\*\*\*\*\* Database Created \*\*\*\*\*\*\*");

}

TO LIST DATABASES

private static void GetDatabases(DocumentClient client) {

Console.WriteLine();

Console.WriteLine();

Console.WriteLine("\*\*\*\*\*\*\*\* Get Databases List \*\*\*\*\*\*\*\*");

var databases = client.CreateDatabaseQuery().ToList();

foreach (var database in databases) {

Console.WriteLine(" Database Id: {0}; Rid: {1}",

database.Id, database.ResourceId);

}

Console.WriteLine();

Console.WriteLine("Total databases: {0}", databases.Count);

}

TO DELETE DATABASE

private async static Task DeleteDatabase(DocumentClient client) {

Console.WriteLine();

Console.WriteLine("\*\*\*\*\*\*\*\* Delete Database \*\*\*\*\*\*\*\*");

Database database = client

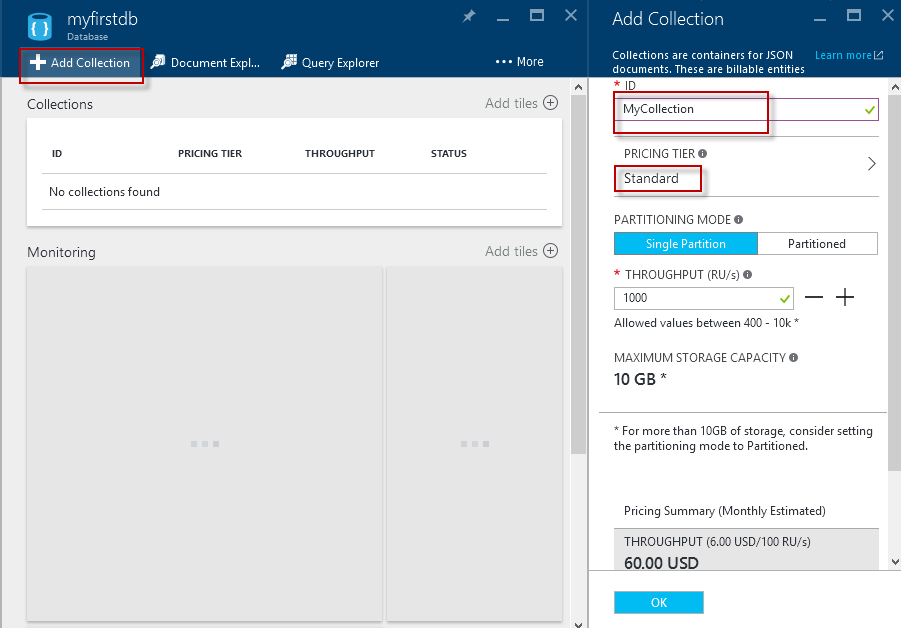
.CreateDatabaseQuery("SELECT \* FROM c WHERE c.id = 'tempdb1'")

.AsEnumerable()

.First();

await client.DeleteDatabaseAsync(database.SelfLink);

}



First Create **MyCollection**  in database **myfirstdb**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Microsoft.Azure.Documents;

using Microsoft.Azure.Documents.Client;

using Microsoft.Azure.Documents.Linq;

using Newtonsoft.Json;

namespace DocumentDBDemo

{

class Program

{

private const string EndpointUrl = "https://azuredocumentdbsuketu.documents.azure.com:443/";

private const string AuthorizationKey = "TKozj2ilHlIIa4NqNjLF6IxVzxMMeQWN3YgFFeQTKnyGqhexjOpENkmvQszw1crBxGKCNSNxlDm1hYBNX8yXzQ==";

private static Database database;

private static DocumentCollection collection;

static void Main(string[] args)

{

try

{

CreateDocumentClient().Wait();

}

catch (Exception e)

{

Exception baseException = e.GetBaseException();

Console.WriteLine("Error: {0}, Message: {1}", e.Message, baseException.Message);

}

Console.ReadKey();

}

private static async Task CreateDocumentClient()

{

// Create a new instance of the DocumentClient

var client = new DocumentClient(new Uri(EndpointUrl), AuthorizationKey);

{

database = client.CreateDatabaseQuery("SELECT \* FROM c WHERE c.id = 'myfirstdb'").AsEnumerable().First();

collection = client.CreateDocumentCollectionQuery(database.CollectionsLink, "SELECT \* FROM c WHERE c.id = 'MyCollection'").AsEnumerable().First();

await CreateDocuments(client);

}

}

private async static Task<Document> CreateDocument(DocumentClient client, object documentObject)

{

var result = await client.CreateDocumentAsync(collection.SelfLink, documentObject);

var document = result.Resource;

Console.WriteLine("Created new document: {0}\r\n{1}", document.Id, document);

return result;

}

private async static Task CreateDocuments(DocumentClient client)

{

Console.WriteLine();

Console.WriteLine("\*\*\*\* Create Documents \*\*\*\*");

Console.WriteLine();

dynamic document1Definition = new

{

name = "Suketu Nayak",

address = new

{

addressType = "Suketu Office",

addressLine1 = "123 Main Street",

location = new

{

city = "Ahmedabad",

stateProvinceName = "Gujarat"

},

postalCode = "380015",

countryRegionName = "India"

},

};

Document document1 = await CreateDocument(client, document1Definition);

Console.WriteLine("Created document {0} from dynamic object", document1.Id);

Console.WriteLine();

}

}

}