22- WCF -2 DOTNET

1. Write WCF Service Task, CRUD operations on database table.

[ServiceContract]

public interface IEmployeeService

{

[OperationContract]

bool AddEmployee(Employee emp);

[OperationContract]

List<Employee> RetreiveEmployees();

[OperationContract]

Employee RetreiveEmployeeByID(int empId);

[OperationContract]

bool UpdateEmployee(Employee emp);

[OperationContract]

bool DeleteEmployee(int empId);

}

1. Service implementation

public class EmployeeService : IEmployeeService

{

DBCommunicator dbObj = new DBCommunicator();

public bool AddEmployee(Employee emp)

{

try

{

return dbObj.AddEmployee(emp);

}

catch (Exception ex)

{

throw ex;

}

}

public bool DeleteEmployee(int empId)

{

try

{

return dbObj.DeleteEmployee(empId);

}

catch (Exception ex)

{

throw ex;

}

}

public Employee RetreiveEmployeeByID(int empId)

{

try

{

return dbObj.EmployeeDetails(empId);

}

catch (Exception ex)

{

throw ex;

}

}

public List<Employee> RetreiveEmployees()

{

try

{

return dbObj.EmployeeList();

}

catch (Exception ex)

{

throw ex;

}

}

public bool UpdateEmployee(Employee emp)

{

try

{

return dbObj.UpdateEmployee(emp);

}

catch (Exception ex)

{

throw ex;

}

}

}

1. Self-hosted in console app

ServiceHost emphost = new ServiceHost(typeof(EmployeeService.EmployeeService));

try

{

emphost.Open();

Console.WriteLine($"Employee Service Host Started @ {DateTime.Now}");

}

catch (Exception ex)

{

Console.WriteLine(ex.Message);

}

Console.ReadLine();

emphost.Close();

1. Configuration settings

<system.serviceModel>

<services>

<service behaviorConfiguration="mexBehavior" name="EmployeeService.EmployeeService">

<endpoint address="EmpService" binding="basicHttpBinding" name="usingBasicHttp"

contract="EmployeeService.IEmployeeService" />

<endpoint address="mex" binding="mexHttpBinding" name="usingMex"

contract="IMetadataExchange" />

<host>

<baseAddresses>

<add baseAddress="http://localhost:2385/" />

</baseAddresses>

</host>

</service>

</services>

<behaviors>

<serviceBehaviors>

<behavior name="mexBehavior">

<serviceMetadata httpGetEnabled="true"/>

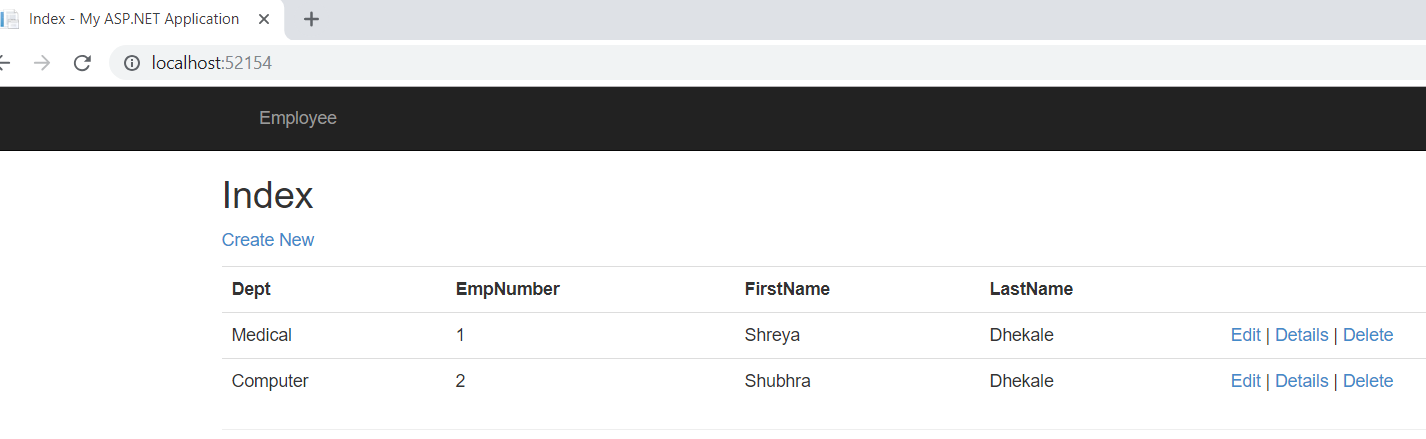
</behavior>

</serviceBehaviors>

</behaviors>

</system.serviceModel>

1. Created MVC application to consume above created service.



Weather service:

1. Service Contract

[ServiceContract]

public interface IWeatherInfo

{

[OperationContract]

Double celciustofarenheit(double temp);

[OperationContract]

Double farenheittocelcius(double temp);

}

1. Service Implementation

public class WeatherInfo : IWeatherInfo

{

public double celciustofarenheit(double temp)

{

double converter = temp \* 1.8 + 32;

return converter;

}

public double farenheittocelcius(double temp)

{

double converter = (temp - 32) / 1.8;

return converter;

}

}

1. Service Host in Windows application

Sd

public partial class WindowsSvcHost : ServiceBase

{

ServiceHost host;

public WindowsSvcHost()

{

InitializeComponent();

}

protected override void OnStart(string[] args)

{

host = new ServiceHost(typeof(WeatherService.WeatherInfo));

host.Open();

}

protected override void OnStop()

{

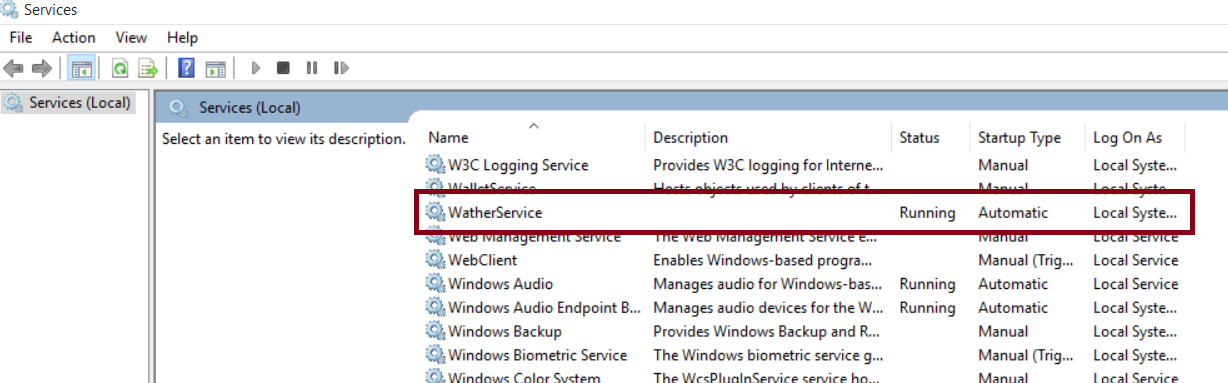
if (host != null)

host.Close();

}

}

1. Install windows service using installutil.exe -i appending wither service name.



1. Console application to consume weather service.

static void Main(string[] args)

{

string userinput = "Y";

while (userinput.Trim().ToUpper() == "Y")

{

TempConvertor();

Console.WriteLine("Do you Want to Continue? Y/N");

userinput = Console.ReadLine();

}

}

static void TempConvertor()

{

try

{

WeatherService.WeatherInfoClient \_client = new WeatherService.WeatherInfoClient();

Console.WriteLine("Temprature Convertor");

Console.WriteLine("Select Conversion");

Console.WriteLine("1. celcius To Farenheit");

Console.WriteLine("2. Farenheit To celcius ");

int option = int.Parse(Console.ReadLine());

if (option == 1)

{

Console.WriteLine("Enter Temp in Clcius");

double temp = double.Parse(Console.ReadLine());

var response = \_client.celciustofarenheit(temp);

Console.WriteLine("Temp in Farenheit : " + response);

}

else if (option == 2)

{

Console.WriteLine("Enter Temp in Farenheit ");

double temp = double.Parse(Console.ReadLine());

var response = \_client.farenheittocelcius(temp);

Console.WriteLine("Temp in Clcius : " + response);

}

else

{

Console.WriteLine("Enter correct Option");

}

}

catch (Exception ex)

{

Console.WriteLine("Something wrong with Service/ Option selection " + ex.Message);

}

}

1. 