

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Runtime.Serialization;
5 using System.ServiceModel;
6 using System.ServiceModel.Web;
7 using System.Text;
8 using System.Data.SqlClient;
9 using System.Web;
10 using System.Data;
11 using System.Xml.Linq;
12 using System.Data.Common;
13
14 // REMARQUE : vous pouvez utiliser la commande Renommer du menu Refactoriser pour changer le nom de classe "Service" dans le code, le fichier svc et le fichier de configuration.
15 public class Service : IService
16 {
17     public string GetData(int value)
18     {
19         return string.Format("You entered: {0}", value);
20     }
21
22     // Serveur SQL en réseau
23     //String sConnexionString = "Server=srv-sql;Database=StationMeteo;User Id=UserMeteo ;Password=MeteoUser;";
24     // BD local
25     String sConnexionString2 = "Data Source=(LocalDB)\\MSSQLLocalDB;" +
26         "" +
27         "AttachDbFilename=C:\\3iL_Cours\\I3_IG\\2023_24_TP\\TP2\_WCFService\\TP2\_WCFService\\App\_Data\\DataBase.mdf;" +
28         "Integrated Security=True;";
29     public string Meteo_GetTemperature()
30     {
31         String sSql = "SELECT * FROM [T_Data] WHERE [DateReleve] = \(SELECT MAX\(\[DateReleve\]\) FROM \[T\_Data\]\)";
32         try
33         {
34             SqlConnection sc = new SqlConnection(sConnexionString2);
35             sc.ConnectionString = sConnexionString2;
36             SqlCommand cmd = new SqlCommand(sSql, sc);
37             sc.Open();
38             SqlDataReader dr = cmd.ExecuteReader();
39             String sTemp = "";
40             while (dr.Read())
41             {
42                 sTemp = dr.GetValue(dr.GetOrdinal("Temp")).ToString();
43             }
44             sc.Close();
45             return sTemp;
46         }
47         catch (Exception ex)
```

```

48     return "Error";
49 }
50
51 private bool ISDate(String sDate)
52 {
53     try
54     {
55         DateTime dt = Convert.ToDateTime(sDate);
56         return true;
57     }
58     catch { return false; }
59     return false;
60 }
61
62 public string Meteo_GetTemperatureByDate(string sDate, out string sDateReleve)
63 {
64     sDateReleve = "";
65     if (!ISDate(sDate)) { return "Error"; } // pour des raisons de
66     // sDate 24/09/2023 10:00 doit être formatée en 09/24/2023
67     // 10:00 car les date sont stockées sous format anglais
68     sDate = sDate.Substring(3, 2) + "/" + sDate.Substring(0, 2) +
69     sDate.Substring(5);
70     //      09          /
71     //      24          /2023 ...
72     String sSql = "SELECT * FROM [T_Data] WHERE [DateReleve] =
73     (SELECT MAX([DateReleve]) FROM [T_Data] " +
74     "WHERE DateReleve < '" + sDate + "')";
75
76     try
77     {
78         SqlConnection sc = new SqlConnection(sConnexionString2);
79         sc.ConnectionString = sConnexionString2;
80         SqlCommand cmd = new SqlCommand(sSql, sc);
81         sc.Open();
82         SqlDataReader dr = cmd.ExecuteReader();
83         String sTemp = "";
84         while (dr.Read())
85         {
86             sTemp = dr.GetValue(dr.GetOrdinal("Temp")).ToString
87             (); // Lecture direct en String
88             // ou
89             Double dTemp = dr.GetDouble(dr.GetOrdinal("Temp")); //
90             Lit la température en double
91             sTemp = dTemp.ToString("0.00"); // Convertit Double en
92             String avec 2 chiffres après la virgule
93
94             sDateReleve = dr.GetValue(dr.GetOrdinal
95             ("DateReleveVC")).ToString();
96         }
97         sc.Close();
98         return sTemp;
99     }
100 }

```

```
91     catch (Exception ex)
92     { }
93     return "Error";
94 }
95
96 public DataSet Meteo_GetTemperatureByDay(string sDate)
97 {
98     sDate = sDate.Substring(3, 2) + "/" + sDate.Substring(0, 2) +
99         sDate.Substring(5);
100     DataSet ds = new DataSet();
101     try
102     {
103         sDate = sDate.Substring(0, 10);
104         String sSql = "SELECT * FROM [T_Data] WHERE [DateReleve] > " +
105             sDate +
106             "' AND [DateReleve] < DateAdd(d, 1, '" + sDate
107             + "')";
108         // SELECT * FROM [T_Data] WHERE [DateReleve] > '23/09/2023'
109         // AND AND [DateReleve] < DateAdd(d, 1, '23/09/2023')
110
111         using (SqlConnection sc = new SqlConnection
112             (sConnexionString2))
113         {
114             sc.Open();
115             SqlDataAdapter da = new SqlDataAdapter(sSql, sc);
116             da.Fill(ds);
117         }
118         return ds;
119     }
120     catch
121     {
122         return null;
123     }
124 }
125
126 //public CompositeType GetDataUsingDataContract(CompositeType
127 //    composite)
128 //{
129 //    if (composite == null)
130 //    {
131 //        throw new ArgumentNullException("composite");
132 //    }
133 //    if (composite.BoolValue)
134 //    {
135 //        composite.StringValue += "Suffix";
136 //    }
137 //    return composite;
138 }
```