

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ReportBackups
{
    class Application
    {
        static void Main(string[] args)
        {
            // Connection string
            // The server and the credentials will need to be changed in order for the program to work
            // with another server and / or db.

            var connstr = @"server=TestServer;database=ReportServer; Trusted_Connection=True;";

            // SQL Query
            // SQL Query used to take the underlying data to generate the underlying SSRS reporting in
            terms of
            // content / XML title and folder position on the network.

            var query = "SELECT C.NAME + ' ' + convert(varchar, getdate(), 104) as [NAME], SUBSTRING
(
REPLACE(REPLACE(C.Path, C.Name, ''), '/', '\\'), 1, 500) + '\\\ ' as [PATH], CONVERT(NVARCHAR(MAX),
CONVERT(XML, CONVERT(VARBINARY(MAX), C.CONTENT))) AS REPORTXML FROM
[ReportServer].[dbo].[Catalog] C WHERE C.CONTENT IS NOT NULL AND C.TYPE = 2 and ModifiedDate
>= dateadd(DD, -1, cast(getdate() as date))";

            var da = new SqlDataAdapter(query, connstr);
            var dt = new DataTable();

```

```

da.Fill(dt);

Console.WriteLine("ReportBackups 1.15");

Console.WriteLine("Christopher Lee - Senior Information Analyst
(Christopher.Lee@Nottshc.nhs.uk");

// Recursively move through the reports generated in the underlying SQL query.
// Checking folders (generating folders as necessary) and generating underlying RDL files to
then populate
// these folders.

foreach (DataRow item in dt.Rows)
{
// Create variables for the folder / path of the reports and the file names for the reports.

var path = @"H:\" + item["PATH"].ToString();
var fname = item["NAME"].ToString() + ".rdl";

// Check whether underlying folder / paths exists to store report file via boolean check.
// If the underlying folder / path doesn't exist then the folder is created.

bool exists = System.IO.Directory.Exists(path);

if (!exists)
    System.IO.Directory.CreateDirectory(path);

path += item["NAME"].ToString() + ".rdl";

// Delete the file if it exists.

// Because the file name is appended with the data it was generated, this process means
that the file

```

```
// cannot be generated twice on the same day.

if (File.Exists(path))
{

    File.Delete(path);
}

using (FileStream fs = File.Create(path))
{
    Byte[] info = new UTF8Encoding(true).GetBytes(item["REPORTXML"].ToString());

    fs.Write(info, 0, info.Length);
}

Console.WriteLine(fname + " report completed.");

}
}
}
}
```