



MUHAMMED SARAC

Sql case 2

Praktik Centret

Indholdsfortegnelse

Task 1	2
Forord	2
Tidsplan	2
Tirsdag 08.08.2017	2
Onsdag 09.08.2017	2
Løsning.....	2
Process.....	3
Diagram.....	3
Kode.....	5
Kilder.....	8
Software	8
Konklusion	8
Use case.....	8
Task 2	8
Forord	8
Tidsplan	9
Onsdag 10.08.2017	9
Løsning.....	9
Process.....	9
Kilde	9
Konklusion	9
Task 3	9
Forord	9
Tidsplan	9
Torsdag 10.08.2017	9
Process.....	9
Kilde	10
Konklusion	10
Task 4.....	10
Forord	10
Tidsplan	10
Torsdag 10.08.2017	10
Process.....	10

Kilde	11
Konklusion	11
Task 5	11
Forord	11
Tidsplan	11
Torsdag 10.08.2017	11
Process.....	11
Kilde	11
Konklusion	12
Task 6	12
Forord	12
Tidsplan	12
Torsdag 10.08.2017	12
Process.....	12
Kode.....	12
Konklusion	15
Task 7	15
Task 8.....	15

Task 1

Forord

I dette øvelse skal vi arbejde en gruppe og finde en løsning til tec. Tec har brug for en løsning i deres sikkerheds struktur hvor personfølsomme data bliver gemt, i to forskellige datacentreer. Her skal vi diskutere i gruppen om hvordan vi kan øge sikkerheden og få en godkendelse af datasikkerheden

Tidsplan

Tirsdag 08.08.2017

Diskuter øvelsen, finde løsning, rapport

Onsdag 09.08.2017

Installer server, oprette database og tabeller, få maskiner til samarbejde ,kode i c#, test, rapport

Løsning

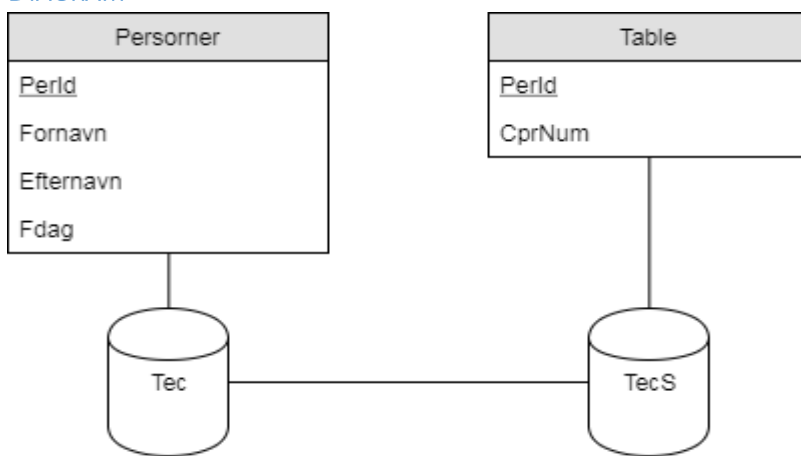
Vi fandt en løsning med oprette to databaser i to forskellige servere og oprette to tabeller. Hvor der er Person id, fornavn, efternavn, fødselsdato. På det andet med person id, Cpr(det sidste fire) og kode en lille program i c# for få dette to database til samarbejde. Programmet vi koder skal kunne tilføje data til

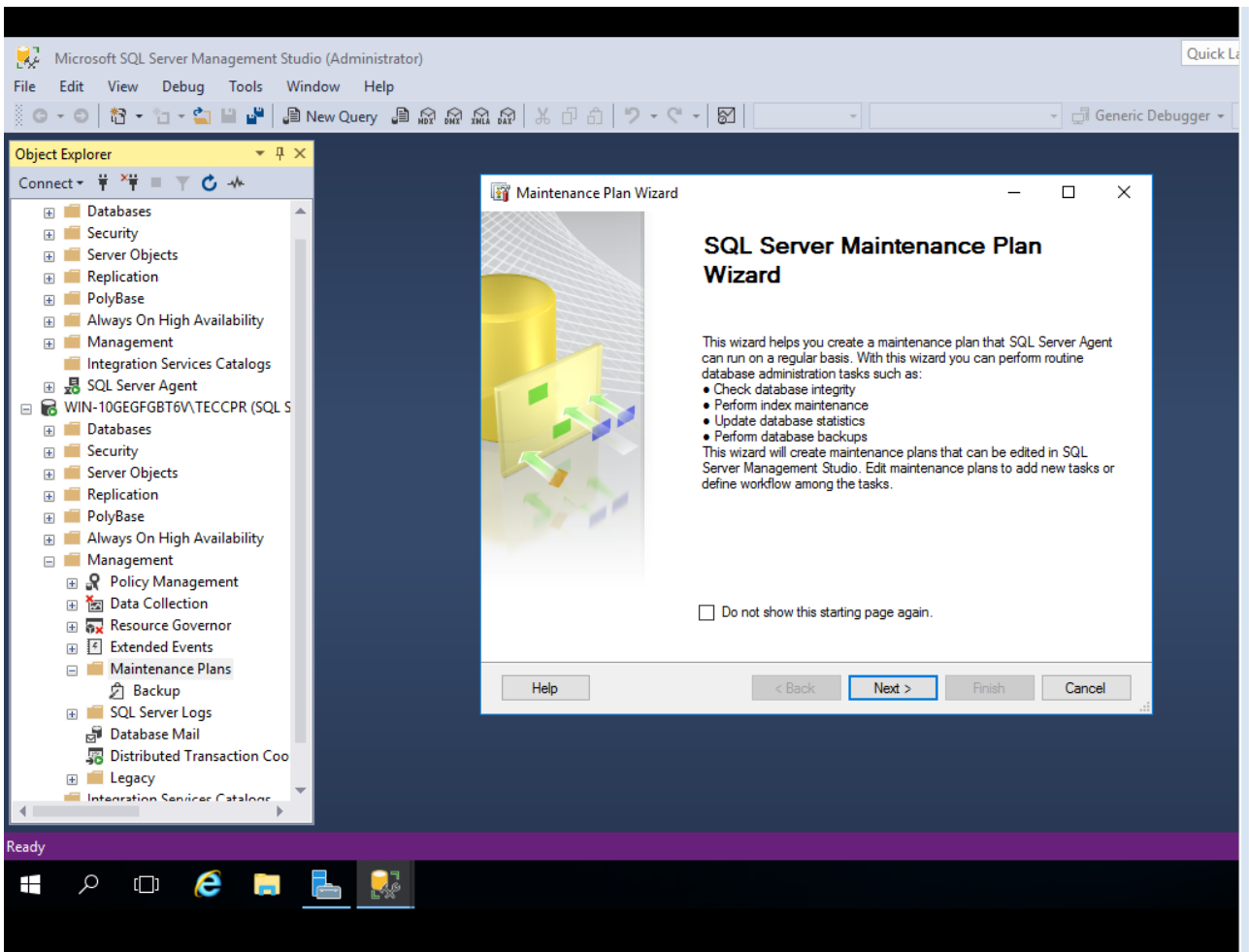
databaserne hvor det så bliver delt op i deres tabeller og bliver gemt hver for sig. Hvis nu en af dem bliver hacked kan hacker ikke få hele cprnr på dette måde kan øge sikkerheden på databaserne. Så skal der i hver server sættes backup så men undgår data tab hvis en af databaserne skulle på en måde miste datta.

Process

Opsætte server og installer MS SQL på hver server, oprette database i hver server, oprette tabel personer med kolonnerne personId, fornavn, efternavn, fødslesdato i en af serveren. Oprette tabel personcpr med kolonnerne personid, cpr på det andet database .Sætte backup op i hver server med maintanance plan wizard hvor du igennem guiden vælger destinationen af backup filen og hvad backup filen skal være, hvor tit der skal tages backup, hvor meget af dataerne skal tages backup på. Her vælger vi full backup og følger guiden ud. Kode i C# og teste om det hele virker som det skal.

DIAGRAM

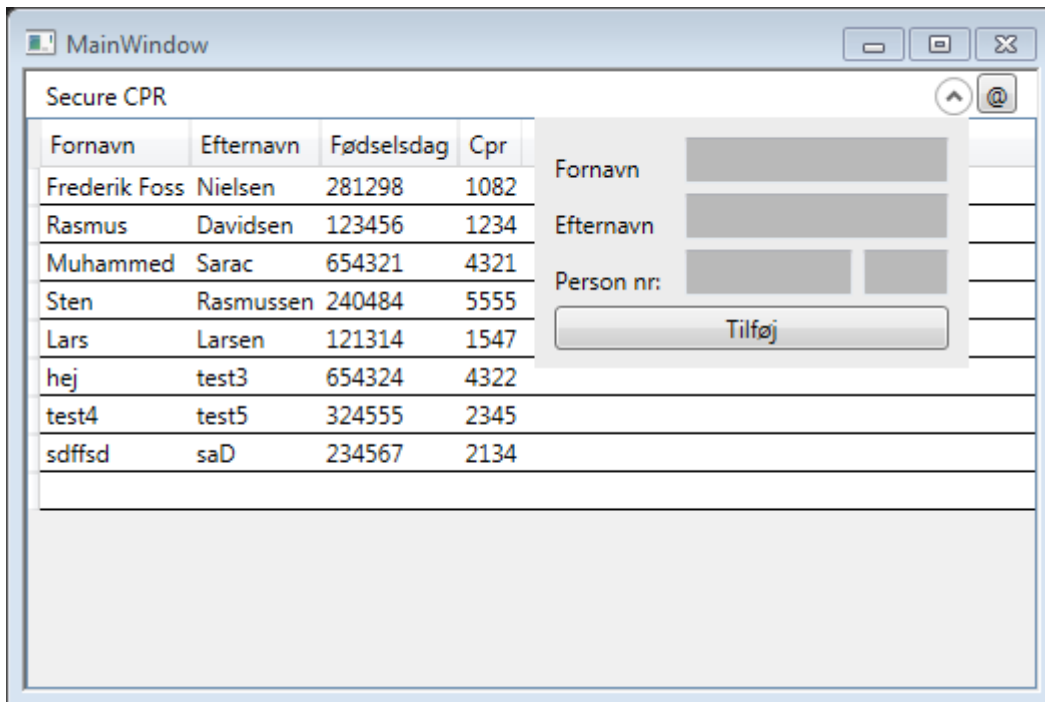




MainWindow

Secure CPR

Fornavn	Efternavn	Fødselsdag	Cpr
Frederik Foss	Nielsen	281298	1082
Rasmus	Davidsen	123456	1234
Muhammed	Sarac	654321	4321
Sten	Rasmussen	240484	5555
Lars	Larsen	121314	1547
hej	test3	654324	4322
test4	test5	324555	2345
sdffsd	saD	234567	2134



Kode

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Navigation;
using System.Windows.Shapes;

namespace SqlServerSome
{
    public partial class MainWindow : Window
    {
        private static CprSql cp = new CprSql("WIN-10GEGFGBT6V\\TECCPR", "Tec", "sa",
"Foss2812", "WIN-10GEGFGBT6V\\SQL", "TecS", "sa", "Foss2812");
        public MainWindow()
        {
            InitializeComponent();
            UpdateGrid();
        }

        private void Button_Click(object sender, RoutedEventArgs e)
    
```

```

    {
        UpdateGrid();
    }

    private void addButton_Click(object sender, RoutedEventArgs e)
    {
        cp.CreateUser(nameBox.Text, lastNameBox.Text, Convert.ToInt32(ageBox.Text),
Convert.ToInt32(cprBox.Text));
        UpdateGrid();
        nameBox.Text = "";
        lastNameBox.Text = "";
        cprBox.Text = "";
        ageBox.Text = "";
    }

    private void UpdateGrid()
    {
        DataGridUsers.ItemsSource = cp.GetUser().DefaultView;
    }
}

class CprSql
{
    private ConnectionAndCmd userCon;
    private ConnectionAndCmd cprCon;

    public CprSql(string serverName, string catalogName, string userName, string
password, string serverName2, string catalogName2, string userName2, string password2)
    {
        userCon = new ConnectionAndCmd(serverName, catalogName, userName, password);
        cprCon = new ConnectionAndCmd(serverName2, catalogName2, userName2, password2);
    }

    public DataTable GetUser()
    {
        DataTable dt = new DataTable();
        dt.Columns.Add("Fornavn", typeof(string));
        dt.Columns.Add("Efternavn", typeof(string));
        dt.Columns.Add("Fødselsdag", typeof(int));
        dt.Columns.Add("Cpr", typeof(int));

        JoinTables(dt, GetData(userCon, "SELECT * FROM Personer"), GetData(cprCon,
"SELECT * FROM PersonNummer"));

        return dt;
    }

    private DataTable JoinTables(DataTable resultDT, DataTable table1, DataTable table2)
    {
        var result = from dataRows1 in table1.AsEnumerable()
                    join dataRows2 in table2.AsEnumerable()
                    on dataRows1.Field<int>("PerId") equals
dataRows2.Field<int>("PerId")

                    select resultDT.LoadDataRow(new object[]
        {
            dataRows1.Field<string>("Fornavn"),
            dataRows1.Field<string>("Efternavn"),
            dataRows1.Field<int>("Fdag"),
            dataRows2.Field<int>("CprNum"),
        }

```

```

        }, false);
        result.CopyToDataTable();

        return resultDT;
    }

    private DataTable GetData(ConnectionAndCmd myCon, string command)
    {
        DataTable dt = new DataTable();

        myCon.OpenConnection();
        dt = myCon.NewQuery(command);
        myCon.CloseConnection();

        return dt;
    }

    public void CreateUser(string name, string lastname, int age, int cpr)
    {
        userCon.OpenConnection();
        int id = userCon.GetId($"INSERT INTO Personer OUTPUT INSERTED.PerId VALUES
('{name}', '{lastname}', {age});");
        userCon.CloseConnection();

        InsertCpr(id, cpr);
    }

    private void InsertCpr(int id, int cpr)
    {
        cprCon.OpenConnection();
        cprCon.NewQuery($"INSERT INTO Personnummer VALUES ({id}, {cpr})");
        cprCon.CloseConnection();
    }
}

class ConnectionAndCmd
{
    SqlConnection con;
    SqlCommand cmd;

    public ConnectionAndCmd(string serverName, string catalogName, string userName,
string password)
    {
        con = new SqlConnection($"Data Source={serverName}; Initial
Catalog={catalogName}; User id={userName}; Password={password}");
        cmd = new SqlCommand();
        cmd.Connection = con;
    }

    public DataTable NewQuery(string command)
    {
        DataTable dt = new DataTable();
        cmd.CommandText = command;
        SqlDataAdapter da = new SqlDataAdapter(cmd);
        da.Fill(dt);
        return dt;
    }
}

```



```
public int GetId(string command)
{
    cmd.CommandText = command;
    return Convert.ToInt32(cmd.ExecuteScalar());
}

public void OpenConnection()
{
    con.Open();
}

public void CloseConnection()
{
    con.Close();
}
}
```

Kilder

<https://www.w3schools.com/sql/>

<https://www.dotnetperls.com/sqlconnection>

<https://docs.microsoft.com/en-us/sql/relational-databases/maintenance-plans/options-in-the-back-up-database-task-for-maintenance-plan>

Software

Windows Server 2012

Ms Sql 2014

Visual studio

Konklusion

Vi har i øvelsen lavede to forskellige server, med en database og en tabel i hver, for gemme dataerne i to del. Så vi kunne øge sikkerheden for dataerne. For få det to server til samarbejde, har vi kodet et lille program. Som ekstra har vi udviklet programmet til en interface program hvor bruger kan indtaste data til databaserne og hente data fra databaserne.

Use case

Dette vil virke med en lille program som man taster data ind og det så bliver splittede og gemt i det to forskellige databaser. Som ekstra kan bruger i interfacen se data og sætte data ind. I enhver situation om hacker angreb eller data tab kan men altid gendanne data via backup.

Task 2

Forord

I dette øvelse skal vi flytte alle odense kommunes data til en google cloud fra databasen.

Tidsplan

Onsdag 10.08.2017

Gør dokumentation klar, snakke om øvelsen, inhente data, gemme i google drive.

Løsning

Vi tager et sql dump i csv fil fra vores database og oploder den til vores ny google cloud og importer dens indhold til Cloud SQL instance så er dataerne flyttede. Så kan vi så hente dataerne fra clouden i xml fil og gemme dem i google drive til bruge som backup.

Process

Vi starter med starte maintenance wizard i Sql server management, i stien som kan ses på billedet i diagramscreenshots. Laver en backup (dump) i en CSV fil der cloud sql kan læse dette fil efter dette oploder vi filen til vores Google cloud og importer indholdet til Cloud SQL. Vi kan så hente data fra Google cloud i xml fil og gemme den i google drive til backup bruge.

Kilde

<https://cloud.google.com/sql/docs/mysql/create-instance>

Konklusion

Vi har i dette øvelse fundet en løsning til Odense kommune om deres data flytning til Google Cloud og lave backup i Google Drive.

Task 3

Forord

På dette øvelse skal vi lave en powershell script hvor vi skal Start PowerShell ISE (x86), opret et nyt PowerShell script
Lav en Variabel i PowerShell script
Lav et array i PowerShell Script
Lav en while i PowerShell Script
Tilføj et share folder som netværks drive

Tidsplan

Torsdag 10.08.2017

Lave powershell script og udføre kommandoerne.

Process

Vi starter med starte powershell ved start og powershell Ise X(86) på windows server.

Vælger new op til højre og starter med kode.

Variable i powershell

```
[string]$a = "sarac"
```

Array I powershell

```
[int[]] $myIntArray = 12,64,8,64,12
```

While loop I powershell

```
while($b -ne 10) { $b++ ; write-Host $b }
```

Share folder som netværkdrev

```
New-Item "C:\Shared" -type directory
New-SMBShare -Name "Shared" -Path "C:\Shared" `
-ContinuouslyAvailable `
-FullAccess domain\admingroup `
-ChangeAccess domain\deptusers `
-ReadAccess "domain\authenticated users"
```

Kilde

<https://www.howtogeek.com/132354/how-to-map-network-drives-using-powershell/>

Konklusion

I dette task har jeg oprettede en powershell script og udført kommandoer og oprettede en sharet folder som netværkdrev.

Task 4

Forord

På dette task skal vi oprette forbindelse til MS SQL ved hjælp af PowerShell.
Vis MS SQL data ved hjælp af PowerShell

Tidsplan

Torsdag 10.08.2017

Lave powershell script og udføre kommandoerne.

Process

Vil her i tasken bruge følgende kode til koble til MS SQL serveren og indhente og se dataerne.

```
$SQLServer = "WIN-10GEGFGBT6\TECCPR"
$SQLDBName = "Tec"
$SQLPassword = "Foss2812"
$SQLUsername = "sa"
$SqlQuery = "select * from Test"
```

```
$SqlConnection = New-Object System.Data.SqlClient.SqlConnection
$SqlConnection.ConnectionString = "Server = $SQLServer;Database = $SQLDBName;User ID = $SQLUsername;Password = $SQLPassword"
```

```
$SqlCommand = New-Object System.Data.SqlClient.SqlCommand
$SqlCommand.CommandText = $SqlQuery
$SqlCommand.Connection = $SqlConnection
```

```
$SqlAdapter = New-Object System.Data.SqlClient.SqlDataAdapter
$SqlAdapter.SelectCommand = $SqlCommand
```

```
$DataSet = New-Object System.Data.DataSet  
$SqlAdapter.Fill($DataSet)
```

```
$SqlConnection.Close()
```

```
$DataSet.Tables[0]
```

Kilde

<https://stackoverflow.com/questions/25682703/connect-to-sql-server-database-from-powershell>

Konklusion

I dette task har jeg kodet script til koble til serveren og vise data i powershell.

Task 5

Forord

I dette task skal vi ekporter data fra database til xml med RAW, AUTO, PATH og forklare hvorfor man kan bruge XML til andre Databaser.

Tidsplan

Torsdag 10.08.2017

Lave ny query i database og udføre kommandoerne.

Process

Eksporter nogle data fra Databasen til XML med RAW

```
SELECT * from Tec.dbo.Personer
```

```
FOR XML RAW
```

Eksporter nogle data fra databasen til XML med AUTO

```
SELECT * from Tec.dbo.Personer
```

```
FOR XML AUTO
```

Eksporter nogle data fra databasen til XML med PATH

```
SELECT * from Tec.dbo.Personer
```

```
FOR XML PATH
```

Kilde

https://blogs.msdn.microsoft.com/saurabh_singh/2010/05/11/export-sql-table-records-to-xml-form/

Konklusion

Jeg har i serveren/ new query kørt de forskellige kommandoer og lavet nogle xml filler af de forskellige kommandoer.

Task 6

Forord

I dette øvelse skal vi oprette forbindelse til server via c# og det har vi lavede i task 1 det vi fik to server til samarbejde med et c# program og som ekstra har udvidet den. Vi tilføjer en login til den nu.

Tidsplan

Torsdag 10.08.2017

Kode i c#.

Process

Vi laver et login side til vores tidlige program i c# hvor vi henviser textbox et som bruger i koden og password til password i koden.

Kode

```
namespace SqlServerSome
{
    /// <summary>
    /// Interaction logic for SqlServerBrowser.xaml
    /// </summary>
    public partial class SqlServerBrowser : Page
    {
        CprSql cp;

        public SqlServerBrowser(string username, string password)
        {
            try
            {
                InitializeComponent();
                cp = new CprSql("WIN-10GEGFGBT6V\\TECCPR", "Tec", username, password, "WIN-10GEGFGBT6V\\SQL", "TecS", username, password);
                UpdateGrid();
            }
            catch { ToLogin(); }
        }

        private void Button_Click(object sender, RoutedEventArgs e)
        {
            UpdateGrid();
        }

        private void addButton_Click(object sender, RoutedEventArgs e)
        {
            try
            {
                cp.CreateUser(nameBox.Text, lastNameBox.Text, Convert.ToInt32(ageBox.Text),
                Convert.ToInt32(cprBox.Text));
                UpdateGrid();
                nameBox.Text = "";
                lastNameBox.Text = "";
            }
        }
    }
}
```

```

        cprBox.Text = "";
        ageBox.Text = "";
    }
    catch { ToLogin(); }
}

private void UpdateGrid()
{
    try
    {
        DataGridUsers.ItemsSource = cp.GetUser().DefaultView;
    }
    catch { ToLogin(); }
}

private void ToLogin()
{
    Window.GetWindow(this).Content = new Login();
}
}
}

class CprSql
{
    private ConnectionAndCmd userCon;
    private ConnectionAndCmd cprCon;

    public CprSql(string serverName, string catalogName, string userName, string password,
string serverName2, string catalogName2, string userName2, string password2)
    {
        userCon = new ConnectionAndCmd(serverName, catalogName, userName, password);
        cprCon = new ConnectionAndCmd(serverName2, catalogName2, userName2, password2);
    }

    public DataTable GetUser()
    {
        DataTable dt = new DataTable();
        dt.Columns.Add("Fornavn", typeof(string));
        dt.Columns.Add("Efternavn", typeof(string));
        dt.Columns.Add("Fødselsdag", typeof(int));
        dt.Columns.Add("Cpr", typeof(int));

        JoinTables(dt, GetData(userCon, "SELECT * FROM Personer"), GetData(cprCon, "SELECT *
FROM PersonNummer"));

        return dt;
    }

    private DataTable JoinTables(DataTable resultDT, DataTable table1, DataTable table2)
    {
        try
        {
            var result = from dataRows1 in table1.AsEnumerable()
                        join dataRows2 in table2.AsEnumerable()
                        on dataRows1.Field<int>("PerId") equals
dataRows2.Field<int>("PerId")

                        select resultDT.LoadDataRow(new object[]
            {
                dataRows1.Field<string>("Fornavn"),

```

```

        dataRows1.Field<string>("Efternavn"),
        dataRows1.Field<int>("Fdag"),
        dataRows2.Field<int>("CprNum"),
    }, false);
    result.CopyToDataTable();
}
catch { }
return resultDT;
}

private DataTable GetData(ConnectionAndCmd myCon, string command)
{
    DataTable dt = new DataTable();

    myCon.OpenConnection();
    dt = myCon.NewQuery(command);
    myCon.CloseConnection();

    return dt;
}

public void CreateUser(string name, string lastname, int age, int cpr)
{
    userCon.OpenConnection();
    int id = userCon.GetId($"INSERT INTO Personer OUTPUT INSERTED.PerId VALUES
('{name}', '{lastname}', {age});");
    userCon.CloseConnection();

    InsertCpr(id, cpr);
}

private void InsertCpr(int id, int cpr)
{
    cprCon.OpenConnection();
    cprCon.NewQuery($"INSERT INTO Personnummer VALUES ({id}, {cpr})");
    cprCon.CloseConnection();
}
}

class ConnectionAndCmd
{
    SqlConnection con;
    SqlCommand cmd;

    public ConnectionAndCmd(string serverName, string catalogName, string userName, string
password)
    {
        con = new SqlConnection($"Data Source={serverName}; Initial Catalog={catalogName};
User id={userName}; Password={password}");
        cmd = new SqlCommand();
        cmd.Connection = con;
    }

    public DataTable NewQuery(string command)
    {
        DataTable dt = new DataTable();
        cmd.CommandText = command;
        SqlDataAdapter da = new SqlDataAdapter(cmd);
    }
}

```

```
        da.Fill(dt);
        return dt;
    }

    public int GetId(string command)
    {
        cmd.CommandText = command;
        return Convert.ToInt32(cmd.ExecuteScalar());
    }

    public void OpenConnection()
    {
        con.Open();
    }

    public void CloseConnection()
    {
        con.Close();
    }
}
```

Konklusion

Vi har opgraderet vores program med logind.

Task 7

Codeplex er en open source platform hvor der bliver delt projekt koder til forbedring fejl søgning og opgradering eller for bare dele sine færdig skrevet projekt til folk kan bruge den. Så alle som vil dele sine koder og kan dele alle finde løsninger til det spurgte eller følge bestemte former for det forskellige projekter. Den lukker snart der jeg tænker at konkrenten github bliver mere brugt og mere flexible end den.

Github kan men de samme i codeplex og mere som du kan installer github i din pc og sykroniser den med din github konti og gør den mere brugbar og flexible. Tænk og lave en projekt med en grubbe, hvor alle kan rediger i kodet og den kan styr dem alle og synkroniser samme tid. Hvor den så gemmer den forige tilstand af koden, så man alle tid kan se forskel på gamle og ny eller vende tilbage.

Task 8

Jeg har præsenteret Task 4 til de andre i grubben.