

<pre>public class AccountInfoCollectionReader { public static IAccountInfoCollection Read(string filename) { AccountInfoCollection accountInfoCollection = new AccountInfoCollection(); accountInfoCollection.AccountInfos = new List<AccountInfo>(); StreamReader inputStream = null; try { inputStream = File.OpenText(filename); string line = inputStream.ReadLine(); if (!line.StartsWith("Name:")) { throw new ParseException(\$"{filename} does not start with *Name:*"); } string[] nameTitle = line.Split(':'); string name = nameTitle[1]; accountInfoCollection.Name = name; line = inputStream.ReadLine(); while (line != null) { AccountInfo accountInfo = new AccountInfo(); string[] input = line.Split(','); accountInfo.Title = input[0]; accountInfo.Username = input[1]; accountInfo.Password = input[2]; accountInfo.Notes = input[3]; string[] fileDate = input[4].Split('/'); DateTime date = new DateTime(Convert.ToInt32(fileDate[2]), Convert.ToInt32(fileDate[1]), Convert.ToInt32(fileDate[0])); accountInfo.Expiration = date; accountInfoCollection.AccountInfos.Add(accountInfo); line = inputStream.ReadLine(); } } catch (FileNotFoundException ex) { MessageBox.Show(ex.Message); } catch (ParseException ex) { MessageBox.Show(ex.Message); } finally { if (inputStream != null) { inputStream.Close(); } } return accountInfoCollection; } }</pre>	<p>Testen of de eerste regel van het bestand start met...</p> <p>ParseException word geworpen indien niet</p> <p>Eerste regel opdelen en splitsen op : String instellen op 2de element van array Naam van collection instellen op string.</p> <p>Volgende regel inlezen</p> <p>Loop doorheen het bestand</p> <p>Nieuwe accountinfo maken -> nodig om te kunnen doorgeven Ingelezen bestand splitsen op ',' Data uit array toevoegen aan de aangemaakte instantie Datum, splitsen op '/' Nieuwe DateTime aanmaken met de gesplitste array </p> <p>Aangemaakte accountinfo toevoegen aan de collectie</p> <p>Volgende lijn inlezen</p> <p>Reader altijd afsluiten in Finally</p> <p>Aangemaakte collectie teruggeven</p>
---	--

```
private void LeesBestand_Click(object sender, RoutedEventArgs e)
{
    string folder = Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments);
    string woorden = System.IO.Path.Combine(folder, "woorden.txt");

    try
    {
        StreamReader reader = File.OpenText(woorden);

        while (!reader.EndOfStream)
        {
            String huidigWoord = reader.ReadLine().Trim();

            if (huidigWoord.Length > 2 && huidigWoord.Length <= 6)
            {
                woordenLijst.Add(reader.ReadLine());
            }
        }
    }
    catch (FileNotFoundException)
    {
        foutmeldingTextBlock.Text = @"De directory 'Mijn documenten' bevat geen bestand woorden.txt";
    }
    catch (FileLoadException)
    {
        foutmeldingTextBlock.Text = "het gegeven bestand is leeg";
    }
}
```

Folder instellen
In folder het bestand woorden.txt open

Trim haalt spaties weg

Woord toevoegen aan woordenlijst

```
private void button2_Click(object sender, System.EventArgs e)
{
    SaveFileDialog saveFileDialog1 = new SaveFileDialog();
    saveFileDialog1.Filter = "Jpeg Image|*.jpg|Bitmap Image|*.bmp|Gif Image|*.gif";
    saveFileDialog1.Title = "Save an Image File";
    saveFileDialog1.ShowDialog();
    if(saveFileDialog1.FileName != "")

    {

        System.IO.FileStream fs =
            (System.IO.FileStream)saveFileDialog1.OpenFile();
        switch(saveFileDialog1.FilterIndex)
        {
            case 1 :
                this.button2.Image.Save(fs,
                    System.Drawing.Imaging.ImageFormat.Jpeg);
                break;

            case 2 :
                this.button2.Image.Save(fs,
                    System.Drawing.Imaging.ImageFormat.Bmp);
                break;

            case 3 :
                this.button2.Image.Save(fs,
                    System.Drawing.Imaging.ImageFormat.Gif);
                break;
        }

        fs.Close();
    }
}
```

Zie OpenFileDialog voor een overzicht van de filters

Als SaveFileDailog1 niet leeg is, doe volgende

```
public static void Write(IList<IToDoItem> toDoItems, string filePath)
{
    IList<IToDoItem> write = toDoItems;
    string[] words = new string[2];

    StreamWriter outputStream = File.CreateText(filePath);

    for (int i = 0; i < write.Count; i++)
    {
        if (write[i] is Task)
        {
            string line = write[i].Serialize();
            line = line.Remove(0, 6);
            line = line.Trim();
            outputStream.Write(line);
            outputStream.Write(Environment.NewLine);
        }
        else
        {
            outputStream.Write(write[i].DisplayText);
            outputStream.Write(Environment.NewLine);
        }
    }

    outputStream.Close();
}
```

IList aanmaken en vullen met meegegeven IList
Nieuwe array aanmaken

StreamWriter openen met met de meegegeven string

Als de IList op positie i een Task is doe volgende

Serialize is een zelfgemaakte methode om strings achter mekaar te zetten
Characters wegdoen van startpositie over 6 spaces
Trim verwijderd voorgaande en volgende whitespaces
String wegschrijven
Naar nieuwe lijn gaan

Indien geen Task

Methode DisplayText aanroepen, welke een strings achter mekaar zet
Naar nieuwe lijn gaan

Writer sluiten

```

private void OpenMenuItemClick(object sender, RoutedEventArgs e)
{
    lastPath = GetPathFromSelectedFile();
    _accountInfoCollection = AccountInfoCollectionReader.Read(lastPath);

    if (_accountInfoCollection.Name != null)
    {
        accountInfoListBox.ItemsSource = _accountInfoCollection.AccountInfos;

        this.Title = "PixelPass - " + _accountInfoCollection.Name;
        newAccountInfoButton.IsEnabled = true;
        _currentAccountInfo = (AccountInfo)accountInfoListBox.SelectedItem;
        titleTextBlock.Text = _currentAccountInfo.Title;
        usernameTextBlock.Text = _currentAccountInfo.Username;
        notesTextBlock.Text = _currentAccountInfo.Notes;
        expirationTextBlock.Text = _currentAccountInfo.Expiration.ToString();
    }
}

private string GetPathFromSelectedFile()
{
    OpenFileDialog openFileDialog = new OpenFileDialog();
    string startDir = Environment.GetFolderPath(Environment.SpecialFolder.Desktop);
    openFileDialog.InitialDirectory = startDir;
    openFileDialog.Filter = "All files (*.*)|*.*";

    if (openFileDialog.ShowDialog() == true)
    {
        return openFileDialog.FileName.ToString();
    }
    else
    {
        return "";
    }
}

```

Itemssource van listbox instellen

Titelbalk instellen

Dialog openen
 Startpath instellen

Filteren op bestanden ->	Word Documents(*.doc) *.doc
	Excel Worksheets(*.xls) *.xls
	PowerPoint Presentations(*.ppt) *.ppt
	Office Files(*.doc;*.xls;*.ppt) *.doc;*.xls;*.ppt
	All Files(*.*) *.*
	Image Files(*.BMP;*.JPG;*.GIF) *.BMP;*.JPG;*.GIF

Word Documents|*.doc|Excel Worksheets|*.xls|PowerPoint Presentations|*.ppt|Office Files|*.doc;
 .xls;.ppt|All Files|*.*

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

```
{
    if (!_currentAccountInfo.IsExpired)
    {

        expirationProgressBar.Value = expirationProgressBar.Maximum;
        _timer = new DispatcherTimer();
        Clipboard.SetText(_currentAccountInfo.Password);
        _timer.Tick += Timer_Tick;
        _timer.Interval = TimeSpan.FromSeconds(1);
        _timer.Start();
    }
}
```

```
private void Timer_Tick(object sender, EventArgs e)
```

```
{
    expirationProgressBar.Value -= expirationProgressBar.Maximum * 0.2;
    if (expirationProgressBar.Value == 0)
    {
        Clipboard.Clear();
        MessageBox.Show("Tijd is om");
        _timer.Stop();
    }
}
```

ProgressBar enkel doen indien instantie niet vervallen is

Beginnen op maximum

Tekst in het geheugen zetten

Timer_Tick methode aanroepen = wat er moet gebeuren met een tick van de timer
Interval van de timer instellen tussen de ()

Wat er moet gebeuren op een tick van de timer

In dit geval de waarde van de ProgressBar verminderen

Wanneer de ProgressBar de 0 bereikt wordt het geheugen gewist, de timer gestopt en de messagebox getoont.

```
private DateTime expirationSliderDate;

public CreateAccountInfoWindow()
{
    InitializeComponent();
    expirationSliderDate = DateTime.Today.AddDays(1);
    createDateSlider();
    expirationSlider.ValueChanged += ExpirationSlider_OnValueChanged;
}

private void createDateSlider()
{
    expirationSlider.Minimum = expirationSliderDate.Day;
    expirationSlider.Value = expirationSlider.Minimum;
    expirationDateTextBlock.Text = expirationSliderDate.ToShortDateString();
}

private void ExpirationSlider_OnValueChanged(object sender, RoutedEventArgs<double> e)
{
    expirationSliderDate = new DateTime(expirationSliderDate.Year, expirationSliderDate.Month, (int)
expirationSlider.Value);
    expirationDateTextBlock.Text = expirationSliderDate.ToShortDateString();
}
```

Object aanmaken welk de slider moet vertegenwoordigen
In dit geval een DateTime

Object instellen op Datum van morgen
Methode createDateSlider aanroepen wanneer het window geopend wordt
Wanneer de slider beweegt, roep methode ExpirationSlider_OnValueChanged aan

Waarde van minimum instellen. In dit geval de dag van morgen.
Hier kan ook de maximum ingesteld worden, maar kan ook in XAML
ToShortDateString geeft automatisch het formaat DD/MM/YYYY terug

Methode voor wat er gebeurt als de slider verschoven wordt

Nieuwe DateTime aanmaken met de waarden van de slider.

Huidige selectie weergeven in het textvak

```
private IAccountInfoCollection _accountInfoCollection;  
private AccountInfo _currentAccountInfo;
```

```
accountInfoListBox.ItemsSource = _accountInfoCollection.AccountInfos;  
this.Title = "PixelPass - " + _accountInfoCollection.Name;  
_currentAccountInfo = (AccountInfo)accountInfoListBox.SelectedItem;  
titleTextBlock.Text = _currentAccountInfo.Title;  
usernameTextBlock.Text = _currentAccountInfo.Username;  
notesTextBlock.Text = _currentAccountInfo.Notes;  
expirationTextBlock.Text = _currentAccountInfo.Expiration.ToString();
```

```
private void AccountInfoListBox_OnSelectionChanged(object sender, SelectionChangedEventArgs e)  
{  
    newAccountInfoButton.IsEnabled = true;  
    AccountInfo account = (AccountInfo)((ListBox)sender).SelectedItem;  
    titleTextBlock.Text = account.Title;  
    usernameTextBlock.Text = account.Username;  
    copyButton.IsEnabled = !account.IsExpired;  
    notesTextBlock.Text = account.Notes;  
    expirationTextBlock.Text = account.Expiration.ToShortDateString();
```

```
    if (!copyButton.IsEnabled)  
    {  
        detailsCanvas.Background = new SolidColorBrush(Colors.LightCoral);  
        expirationProgressBar.Value = 0;  
    }  
    else  
    {  
        expirationProgressBar.Value = expirationProgressBar.Maximum;  
        detailsCanvas.Background = new SolidColorBrush(Colors.White);  
    }  
    _currentAccountInfo = account;  
}
```

Bron instellen van de listBox
Titel van het window instellen
Eerste item dat geselecteerd is -> in XAML SelectedIndex op 0 zetten, anders error
Data uit de listBox invullen in textvakken

Wanneer een ander item geselecteerd wordt

Om een button te enablen
Object aanmaken en geselecteer item van de listBox hieraan toewijzen
Object vullen met data

Copy button enkel enabled indien account niet vervallen is. IsExpired is een methode van het Object

Wanneer button niet enabled is

Achtergrond van het canvas op de gewenste kleur zetten
ProgressBar op 0

Anders is de waarde de ingegeven maximum in XAML
Achtergrond canvas op wit

Object instellen om terug te geven.


```
public class MyFileNotFoundException : Exception
{
}
```

Als klasse
Aanroepen met Throw new classname

```
static double SafeDivision(double x, double y)
{
    if (y == 0)
        throw new System.DivideByZeroException();
    return x / y;
}
```

Met throw

```
try
{
    result = SafeDivision(a, b);
    Console.WriteLine("{0} divided by {1} = {2}", a, b,
result);
}
catch (DivideByZeroException e)
{
    Console.WriteLine("Attempted divide by zero.");
}
```

Met try catch