# StudentModulePuntApp upgrade

## Doelstelling

Vertrekkend van de modeloplossing van vorige week (éénmalig krijgt u deze modeloplossing) zullen we deze aanpassen zodat gebruik wordt gemaakt van resources en styles. Dit is een voorbereiding op het labo van volgende week. Volgende week krijgt u de modeloplossing van deze week niet. De webservice toepassing is dezelfde als deze van vorige week.

## Opdracht: voorbereiding op styles en resources

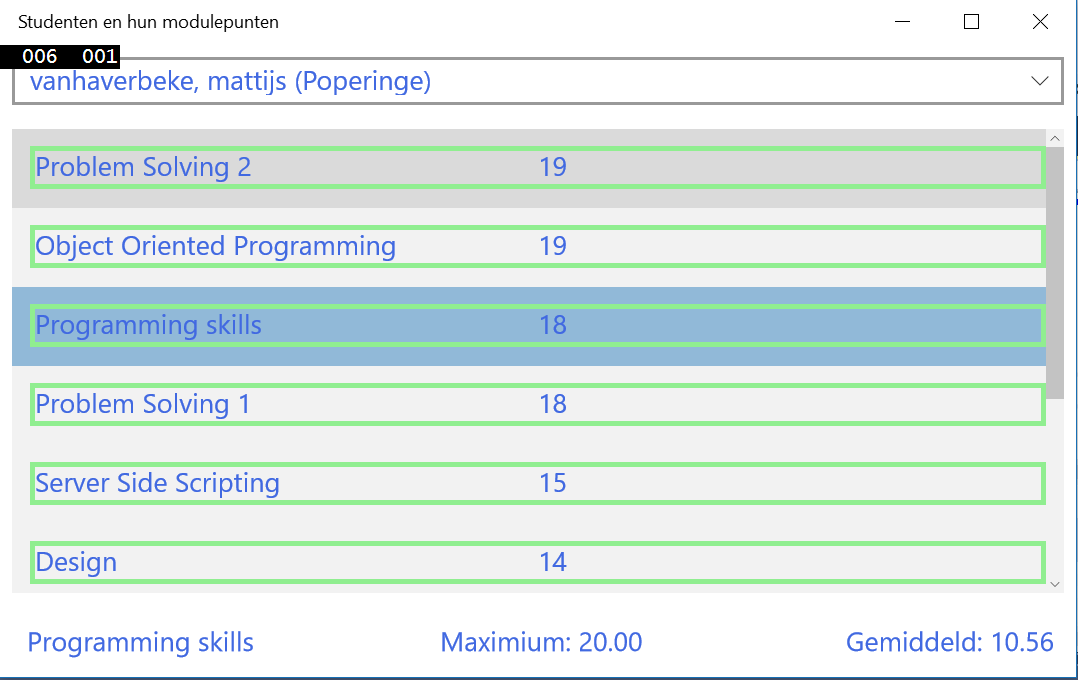
Indien u de Run klasse er op nakijkt (plaats in Visual Studio de cursor op een Run- tag en klik F1) dan merkt u dat de Run- klasse geen style property kent. We kunnen dan ook geen opmaak via styles instellen. Omdat dit een oefening omtrent styles is zullen we de RichTextblock vervangen door een gewone TextBlock (die wel Styles ondersteunt).

Voor wie de oefening niet maakte: ctrl-f helpt u de Run instanties te vinden

## Ontwikkel een default TextBlock style

* De default TextBlock style heeft volgende kenmerken:
  + FontSize: 18 (probeer ook eens 88 om te verifiëren dat de fontsize wordt toegepast)
  + Foreground: koningsblauw
  + Fontfamily: Showcard Gothic (dit enkel ter illustratie: het opvallend zijn van deze font maakt dat het duidelijk is dat hij wordt toegepast)
* Plaats deze style in de Styles- folder (zelf te maken) in de Styles resourcedictionary file (zelf te maken)
* De default TextBlock style zal door de ganse toepassing dezelfde zijn. De enige goede plaats om deze te definiëren is dan ook in App.xaml
  + De tekst [ResourceDictionary.MergedDictionaries property](https://msdn.microsoft.com/en-us/library/windows/apps/windows.ui.xaml.resourcedictionary.mergeddictionaries.aspx) beschrijft netjes hoe u een merged resource directory aanmaakt. Deze verwijst naar de Styles- file die zonet werd aangemaakt
* De default fontsize van een tekstvoorstelling zal ook voor andere controls, en dus voor andere stijlen dezelfde te zijn. Omdat we niet telkens elke stijl willen aanpassen indien de default fontsize wijzigt voeren we volgende aanpassing door:
  + Maak een double resource waarin de default fontsize wordt gedefinieerd
  + Gebruik deze resource om de fontsize van de stijlen te definiëren
* De achtergrond van elke ctlStudentModulePunt wordt via C# code ingesteld. Verwijder deze code, we zullen deze logica vervangen door styles:
  + Ontwikkel een geslaagd en gebuisd style. Beide slides hebben als TargetType Grid (omdat we een Grid zullen voorzien van een achtergrondkleur)
  + Omdat deze styles enkel gebruikt zullen worden in de ctlStudentModulePunt usercontrol definiëren we de styles in de Resources van de usercontrol
  + Geslaagd zal de achtergrondkleur van de Grid waarop hij wordt toegepast lichtgroen kleuren, Gebuisd gebruikt de kleur Coral
  + Het instellen van de style gebeurt (jammer maar helaas) via C# code:
    - Op basis van het behaalde punt wordt de juiste style opgehaald (uit de this.Resources) en ingesteld op de Grid in de usercontrol
  + Wijzig de geslaagd style:
    - We passen de background niet langer aan
    - We voorzien een border die lichtgroen is met een borderthickness van 3
  + Vervang de drie tekstblokken onderaan de mainpage door textboxen
    - Kijk na in hoeverre we kunnen verder bouwen op de TextBlock styles
    - Maak een TextBox style (met key) die dezelfde fontfamily, fontsize en foreground gebruikt als de default textblocken, zonder de border te tonen

Het voordeel bij het gebruiken van stijl over de C# aansturing ligt vooral in de flexibele aanpasbaarheid en het vlotte delen van properties (via de BaseOn style definities).

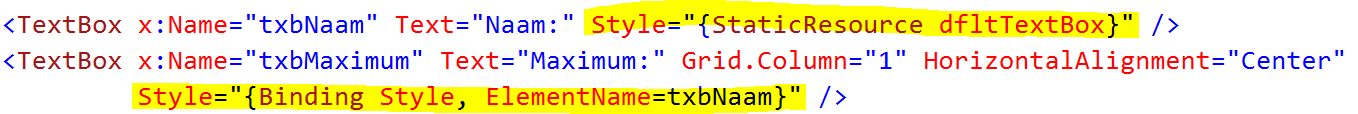


### Opmerking:

Indien u de tekst zomaar volgde zijn in de Styles file twee styles met eenzelfde hardgecodeerde kleur. Maak van deze kleur een resource die in de styles gebruikt wordt.

### Opmerking:

Het instellen van een (niet default) style gebeurt in deze oefeningen momenteel door het gebruik van een static resource:



De tweede TextBox bindt zijn style aan deze van de eerste TextBox (volgende week zien we Binding). Het voordeel hiervan is dat indien meerdere controls dezelfde style gebruiken slechts de style keuze van de eerste control moet aangepast worden om de aanpassing ook in de andere controls terug te zien keren.

### Opmerking:

Het selecteren van een style voor een willekeurige control op basis van randvoorwaarden kan niet in XAML geschreven worden, we dienen dit via C# code in te stellen. Een uitzondering hierop zullen de items in een ListBox control zijn die op basis van de ItemTemplateSelector property hun template (niet hun style!) krijgen toegewezen.

### Opmerking:

uw toepassing start vlot (er zijn wat aanpassingen nodig: zie adaptive UI’s) op een windows mobile device (screenshot: Mobile emulator, 5inch, 1 GB)

