

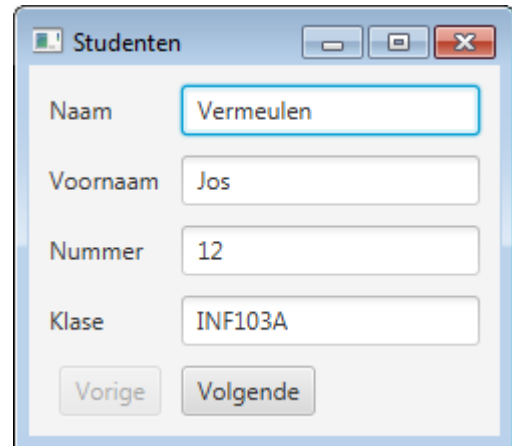
STUDENTEN

We bouwen een applicatie om een lijst van studenten (opgeslagen in een bestand) te raadplegen.

De voornaamste klassen die we voor deze oefening nodig hebben zijn:

- [`java.io.InputStream`](#)
- [`java.io.InputStreamReader`](#)
- [`java.io.BufferedReader`](#)
- [`java.io.IOException`](#)
- [`java.util.StringTokenizer`](#)

Raadpleeg in eerste instantie de [Java documentatie](#) als je ergens vast zit!



1 WIREFRAME

De GUI van deze oefening is reeds uitgewerkt. Het wireframe mag je als oefening maken met behulp van een tool naar keuze.

2 HOOFDSCHERM AANMAKEN – MVP

De *model* klasse is **StudentAdministratie**. Een andere belangrijke model klasse is **Student**. We vullen het model aan in punt 3.

De *view* klasse is **StudentView**. Deze klasse is volledig uitgewerkt.

De *presenter* klasse is **Presenter**. Deze klasse is volledig uitgewerkt.

De **Main** klasse is gegeven.

3 MODEL

In de klasse **StudentAdministratie** werken we de methode **readStudenten** uit.

3.1 De methode readStudenten

- Maak een **InputStreamReader** aan. Aan de constructor geef je als parameter het attribuut **inputStream** mee. (Raadpleeg de constructor van **StudentAdministratie** om te zien hoe dit attribuut geïnitieerd werd!)
- Maak een **BufferedReader** aan op basis van de aangemaakte **InputStreamReader**.
- Gebruik de methode [`readLine`](#) van de klasse **BufferedReader** om de verschillende lijnen van het bronbestand in te lezen. Je doet dit **zolang** de laatst ingelezen lijn niet **null** is.
- Voor elke ingelezen lijn doe je het volgende:
 - Maak een nieuwe **StringTokenizer** aan op basis van de lijn in kwestie.
 - Gebruik de **vier** maal de methode [`nextToken`](#) om de *naam*, *voornaam*, *nummer* en *klas* uit de lijn op te halen.
Het scheidingsteken is puntkomma ";" !

- Maak een **student** object aan op basis van de verkregen gegevens en voeg deze student toe aan de lijst genaamd **studenten**.
Let op: het nummer moet je eerst nog omzetten naar een **int**.
- Vang eventuele **IOExceptions** op en druk de stacktrace af.
- Zorg er voor dat de **BufferedReader** in alle gevallen correct wordt gesloten.
(Dit kan je doen met behulp van *try-with-resources* indien dit concept al behandeld is geweest in de les.)

4 UI OPBOUWEN

De view is volledig uitgewerkt voor deze opdracht.

5 AFHANDELEN EVENTS

De presenter is volledig uitgewerkt voor deze opdracht.