

OOP2 – Practicumopdracht

Week 1

Voor de practicumopdracht gaan we ons deze week focussen op de M van MVC, de *models*.

Verzin een onderwerp waarbij je twee classes nodig hebt met een één-op-meer relatie (Master-Detail constructie), voorbeelden zijn:

- TODO-lijst
 - Master: TodoLijst
 - Detail: TodoRegel
- Boodschappenlijst / Verlanglijst
 - Master: Winkel / Webshop
 - Detail: Artikel
- Cijferlijst
 - Master: Vak
 - Detail: Resultaat
- Adresboek
 - Master: Groep
 - Detail: Contact

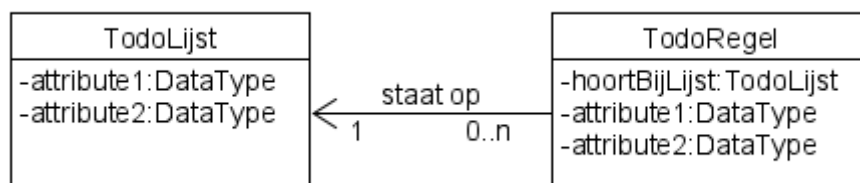
Bedenk bij elke class een aantal nuttige attributen met verschillende datatypes. Elke class moet minimaal twee attributen hebben. Zorg er daarnaast voor dat je alle datatypes in de lijst hieronder minimaal één keer gebruikt:

- LocalDate
- String
- boolean
- double
- int

Daarnaast moet je Detail-class een attribuut hebben met je Master-class als datatype om de één-op-meer relatie mogelijk te maken, bijvoorbeeld:

`TodoRegel` heeft een *private* `TodoLijst` `hoortBijLijst`;

Dit attribuut telt overigens niet mee voor het eerder gestelde minimum aantal attributen.

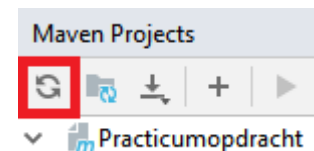


Afbeelding: Voorbeeld van twee classes met een één-op-meer relatie.

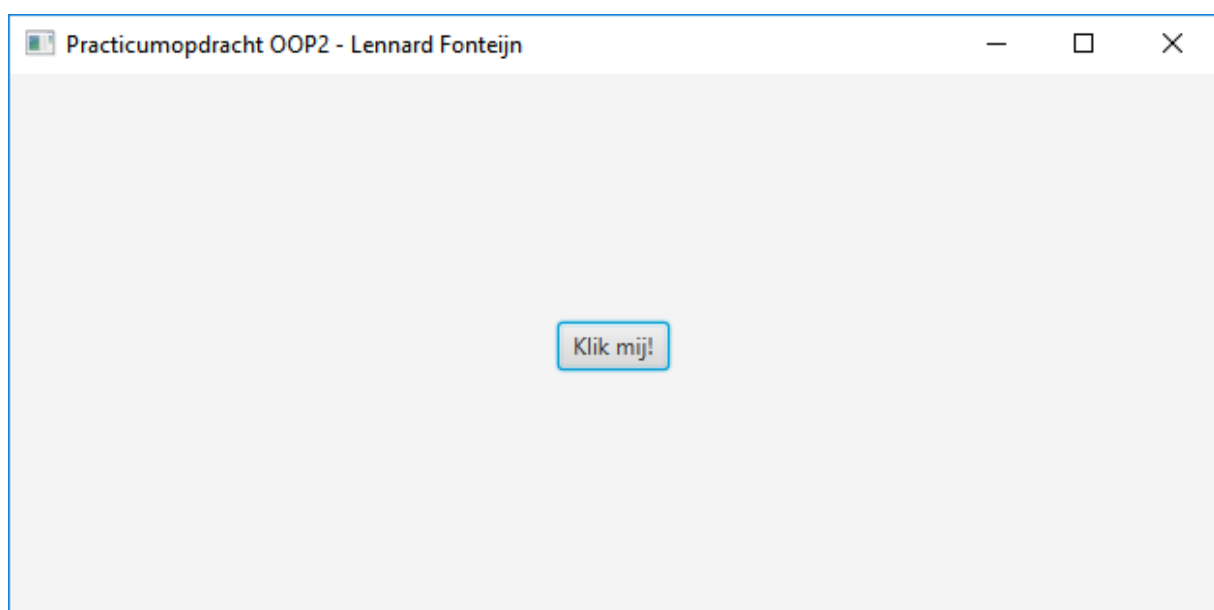
Met een onderwerp in gedachte, volg deze stappen:

1. Maak een project op de HvA GitLab (<https://gitlab.fdmci.hva.nl/>) en nodig je docent daarvoor uit als “Developer”.
2. *Clone* het GitLab-project met *Fork* (<https://git-fork.com/>) op een handige plek.
3. Download de ZIP met het startproject van de DLO en pak deze uit in de map van het gekloonde GitLab-project.
4. Download JDK 13 (<https://www.oracle.com/technetwork/java/javase/downloads/jdk13-downloads-5672538.html>) en installeer deze, als je deze versie nog niet had.
5. Open het project in IntelliJ IDEA en test of je het project correct opstarten vanuit de Main-class. Deze staat in de *practicumopdracht*-package onder *src-map*.

- De *practicumopdracht*-package onder de *src-map* is de plek waar je vanaf nu al je code plaatst, noot erbuiten!
- Bij het opstarten krijg je mogelijk de popup “Maven projects need to be imported”. Als dit het geval is, kies dan voor “Enable Auto-Import”.
- Het kan voorkomen dat je project niet wil opstarten en foutmeldingen geeft. Om dat op te lossen, open dan het “Maven Projects”-scherm vanuit het menu “View > Tool Windows” en klik op de “Refresh”-knop.



6. Pas de Main-class nu zodanig aan, dat deze in het midden van het scherm een knop toont. Maak hiervoor gebruik van het voorbeeld uit hoofdstuf 14.4 van het boek (*ButtonInPane.java*).
7. Maak een package met de naam *models* in de *practicumopdracht*-package van je IntelliJ-project en maak daarin de classes die je eerder hebt verzonnen. Een *constructor*, *getters* en *setters* zijn nog niet nodig. Zorg er wel voor dat de classes geen compile-errors bevatten en dat het project nog steeds opstart.
8. *Commit* all wijzigingen en *push* het resultaat met Fork.



Afbeelding: Voorbeeld van een opstartscherm na het afronden van alle stappen.

Checklist

- ☐ HvA GitLab project gemaakt en docent uitgenodigd als “Developer”.
- ☐ Twee classes bij onderwerp verzonnen met minimaal twee attributen elk. Alle datatypes uit de volgende lijst zijn daarbij gebruikt:
 - ☐ LocalDate
 - ☐ String
 - ☐ boolean
 - ☐ double
 - ☐ int
- ☐ De Detail-class heeft een attribuut verwijzend naar de Master-class.
- ☐ Er is een *Main.java* beschikbaar in het project die een knop toont bij het opstarten.
- ☐ Er is een *models*-package beschikbaar in het project met daarin de twee classes.
- ☐ Het project bevat geen compile-errors.
- ☐ Het project is gecommited en gepushed naar GitLab.