# Vragen OOO

State : p415 no sense to put it in class variables

Button aanpassen

Zoek

if(ks.getOmschrijving().equals("Groepskorting")){

ks.setCategorie(categorie);

}

if(ks.getOmschrijving().equals("Drempelkorting")){

ks.setDrempel(drempel);

}

alternatief : zet setCategorie in constructur, en niet in interface

observablelist: waarom opnieuw setten?

Afrekenen: moet klant ook afrekening zien

Factory geeft

In volgorde die je ze hebt

# Jasper

maak klasse kortingHandler en zet die in verkoop. Haal in onze kassapain dan settingscontroller weg

g moet moogt de producten niet aanpassen in prijs, g moet een nieuwe lijst nemen as g de kortingen gaat toepassen. Reden: as g een product verwijdert na afrekenen te drukken moet g opnieuw korting berekenen want hoogstePrijs kan bv anders zijn. Dan moet g dus ne lijst hebben met originele prijzen. Kopieer in uw « topklasse » kortingHandler eerst ne lijst as g kortingen toepast en die lijst kan dan altij bewerkt worden door de kortingen. G moet ook ni de producten in de observable list van verkoop veranderen, g moet enkel het totaalbedrag met korting er af berekenen en da terug geven

Om lijst te kopieren te maken:

Loop lijst : voeg aan nieuwe lijst toe: huidgArtikelContainer.clone();

addKorting : verplaats naar verkoop

calculateTotaal implementeren, zet best in

Niet prioritized:

Categorie invoer: maak combobox

G doet convertkorting telkens opnieuw terwijl g da ook 1x kunt setten

Gebruik Template

G hebt ne setter gezet in de artikelcontainer : hoort daar ni thuis

public void setPrijs(double prijs){

artikel.setPrijs(prijs);

}

Die gebruikt g in kortingen