

	<p>Datum: 12/09/2017</p> <p>Opleiding: Java Developer</p> <p>Lesmodule: Java Fundamentals</p> <p>Oefening: Merkproducten</p>
	<p>Naam:</p>

Inhoudstafel

Inhoudstafel	1
Deel 1: Project Merkproducten.....	1
Git integratie	1
Swing Integratie	5
Deel 2: Maven Integratie.....	6
Deel 3: junit Integratie	Fout! Bladwijzer niet gedefinieerd.
Deel 4: Database Integratie.....	Fout! Bladwijzer niet gedefinieerd.
JDBC Integratie	Fout! Bladwijzer niet gedefinieerd.

Deel 1: Project Merkproducten

Projectnaam: Merkproducten
Java versie: 1.8

GIT INTEGRATIE

- Maak voor deze opdracht op je persoonlijk GitHub account een git repository aan genaamd Merkproducten.
- Synchroniseer deze met je aangemaakt project in je IDE. Geef het bericht "initial commit" mee hierbij.
- Je maakt alle klassen van een package aan op een aparte branch. Telkens als er een speciale feature is afgewerkt doe je een commit met duidelijke message.
BV: Nieuwe branch **model**
→ Product is afgewerkt: **commit**
→ Aftershave, deodorant en Parfum zijn afgewerkt: **commit**
→ Interface toegevoegd: **commit**
→ Volledig model is afgewerkt: **push** naar branch model en **merge** met **master**
- Voor grote wijzigingen werk je telkens op een andere branch. Voor zeer kleine wijzigingen (typfout, refactoring van code ...) mag je rechtstreeks commit en pushen naar master.

Maak voor dit project gebruik van de meegeleverde klasse **Data** en de klasse **TestApp**. Plaats deze twee klassen in het package **be.oak3.persistence**.

package: be.oak3.model

Klasse Product

Maak een abstracte klasse Product die de volgende variabelen bevat:

- productNummer: int
- merk: string
- naam: string
- volume: int
- prijs: double

Voorzie de volgende constructor en methodes:

- Constructor die de nodige variabelen initialiseert.
- Voorzie de nodige accessor methods (getters en setters), niet meer dan nodig is!
- Zorg dat de **natuurlijke volgorde** van Product sorteert op productnummer.
- Schrijf een methode **"getProductCode()"** die een string teruggeeft

De code bestaat uit:

- o De eerste 3 letters van het merk;
- o De eerste 3 letters van de naam;
- o De volume waarde.

Bovendien wordt alles in hoofdletters gezet en worden eventuele spaties door een underscore vervangen.

Voorbeelden:

Merk: **Chanel** Naam: **Coco Mademoiselle** Volume: **100ml** => **CHACOC100**

Merk: **DKNY** Naam: **Be Delicious Woman** Volume: **100ml** => **DKNBE_100**

- Schrijf een statische methode **"sorteerOpMerkNaam"** die een **comparator** object teruggeeft van het type Product.
 - o Implementeer de compare() methode die de inkomende Product objecten met elkaar vergelijkt op basis van de variabele "merk". Maak hiervoor gebruik van lamda expressions.
- Override de ToString() methode: Kijk voor een goede implementatie van deze methode even naar de printscreen op pagina 3. Maak gebruik van de FORMAT methode!

De klassen AfterShave, Deodorant, en Parfum

Implementeer de volgende klassen

- Laat de klassen AfterShave, Deodorant, en Parfum **overerven** van de abstracte klasse Product.
- Voorzie in de klasse AfterShave een lokale **enum Soort** met de constanten VAPO en GEL voorzie een variabele soort van het type Soort.
- Voorzie in de klasse Deodorant een lokale **enum DeoType** met de constanten VAPO en STICK voorzie een variabele soort van het type DeoType.
- In de klasse Parfum is geen specifieke variabele nodig.
- Schrijf de nodige constructors in de klasse AfterShave, Deodorant, en Parfum. Alles moet aanwezig zijn, ook de 'nieuwe' variabelen.
- Voorzie, indien nodig een overriding van de toString() methode van de super klasse.

De Interface Berekenbaar

Maak een interface Berekenbaar met daarin de methode **"totalePrijs()"**;

package: be.oak3.persistence

De interface Bestelling: laat deze interface extenden van Berekenbaar.

Voorzie volgende abstracte methodes:

- **voegProductToe(Product product);**
- **sorteer();**
- **sorteerOpMerk();**
- **sorteerOpVolume();**
- **toonPerMerk();**
- **toonGoedkopeProducten();**
- **zoekDuursteProduct();**

De klasse BestellingImpl

Implementeer de interface Bestelling in de klasse BestellingImpl. Test elke implementatie stap voor stap aan de hand van de meegeleverde testklasse.

voorzie de volgende members:

- bestelling: ArrayList<>
- Voorzie een static variabele productNummer, die start bij 1000.
- Voorzie de nodige constructors
- Implementeer de methode **"voegProductToe()"** Zorg er ook voor dat het productnummer met 1000 start en met één verhoogd wordt voor elke volgende product dat wordt toegevoegd.
- Implementeer de methode **"sorteer()"**, waar je de lijst sorteert op natuurlijke volgorde!
- Implementeer de methode **"sorteerOpMerk()"** om de bestelling op merk te sorteren, implementeer aan de hand van **"sorteerOpMerkNaam()"** methode van Product.
- Implementeer de methode **"sorteerOpVolume()"** om de bestelling te sorteren op volume
- Sc Implementeer hrijf een methode **"toonPerMerk()"** die een merk als string verwacht waarmee je alleen het product van een bepaalde merk uit de bestelling afdruckt.
- Implementeer de methode **"toonGoedkopeProducten()"**. Deze toont alleen de producten die goedkoper zijn dan 50€.
- Implementeer de methode **"zoekDuursteProduct()"** die als return waarde het product met de hoogste prijs uit de bestelling geeft.
- Implementeer de methode **"toonParfums()"** om alle parfums uit de lijst bestelling te tonen.
- Implementeer de methode **"totalePrijs()"** geef de totale prijs terug van alle producten in de bestelling.

➔ Maak waar je kan gebruik van Lamda Expressions / Streams ...

Klasse TestApp

- Deze maakt gebruik van de klasse data en al je aangemaakte klassen.

→ Volgende uitvoer zou je moeten bekomen:

```

Problems Javadoc Declaration Console Progress
<terminated> TestApp (3) [Java Application] C:\Program Files\Java\jdk1.8.0\bin\javaw.exe (8-sep-2015 16:56:39)
Oplossing van Kenneth Van Gijssel Java Instructeur

Lijst gesorteerd op natuurlijke volgorde:
1000 Merk: Dolce & Gabbana      Naam: Light Blue      Volume: 100ml Prijs: 66,72 Code: DOLLIG100
1001 Merk: BVLGARI              Naam: BLV             Volume: 75ml  Prijs: 61,52 Code: BVLBLV75
1002 Merk: DKNY                 Naam: Be Delicious Women Volume: 100ml Prijs: 33,65 Code: DKNBE_100 STICK
1003 Merk: Giorgio Armani       Naam: Code Donna      Volume: 50ml  Prijs: 59,32 Code: GEOCOD50
1004 Merk: Yves Saint Laurent  Naam: Jazz            Volume: 50ml  Prijs: 39,84 Code: YVEJAZ50 VAPO
1005 Merk: Givency              Naam: Absolutely Irresistible Volume: 75ml  Prijs: 75,42 Code: GIVABS75
1006 Merk: Yves Saint Laurent  Naam: Jazz            Volume: 100ml Prijs: 57,76 Code: YVEJAZ100 VAPO
1007 Merk: Ted Lapidus         Naam: Pour Elle       Volume: 50ml  Prijs: 44,48 Code: TEDPOU50
1008 Merk: Giorgio Armani       Naam: Code Donna      Volume: 30ml  Prijs: 39,84 Code: GEOCOD30
1009 Merk: Giorgio Armani       Naam: Code Donna      Volume: 75ml  Prijs: 76,00 Code: GEOCOD75
1010 Merk: Cacharel            Naam: Anais           Volume: 50ml  Prijs: 24,50 Code: CACANA50 VAPO

Lijst gesorteerd op merk:
1001 Merk: BVLGARI              Naam: BLV             Volume: 75ml  Prijs: 61,52 Code: BVLBLV75
1010 Merk: Cacharel            Naam: Anais           Volume: 50ml  Prijs: 24,50 Code: CACANA50 VAPO
1002 Merk: DKNY                 Naam: Be Delicious Women Volume: 100ml Prijs: 33,65 Code: DKNBE_100 STICK
1000 Merk: Dolce & Gabbana      Naam: Light Blue      Volume: 100ml Prijs: 66,72 Code: DOLLIG100
1003 Merk: Giorgio Armani       Naam: Code Donna      Volume: 50ml  Prijs: 59,32 Code: GEOCOD50
1008 Merk: Giorgio Armani       Naam: Code Donna      Volume: 30ml  Prijs: 39,84 Code: GEOCOD30
1009 Merk: Giorgio Armani       Naam: Code Donna      Volume: 75ml  Prijs: 76,00 Code: GEOCOD75
1005 Merk: Givency              Naam: Absolutely Irresistible Volume: 75ml  Prijs: 75,42 Code: GIVABS75
1007 Merk: Ted Lapidus         Naam: Pour Elle       Volume: 50ml  Prijs: 44,48 Code: TEDPOU50
1004 Merk: Yves Saint Laurent  Naam: Jazz            Volume: 50ml  Prijs: 39,84 Code: YVEJAZ50 VAPO
1006 Merk: Yves Saint Laurent  Naam: Jazz            Volume: 100ml Prijs: 57,76 Code: YVEJAZ100 VAPO

Lijst gesorteerd op volume:
1008 Merk: Giorgio Armani       Naam: Code Donna      Volume: 30ml  Prijs: 39,84 Code: GEOCOD30
1010 Merk: Cacharel            Naam: Anais           Volume: 50ml  Prijs: 24,50 Code: CACANA50 VAPO
1003 Merk: Giorgio Armani       Naam: Code Donna      Volume: 50ml  Prijs: 59,32 Code: GEOCOD50
1007 Merk: Ted Lapidus         Naam: Pour Elle       Volume: 50ml  Prijs: 44,48 Code: TEDPOU50
1004 Merk: Yves Saint Laurent  Naam: Jazz            Volume: 50ml  Prijs: 39,84 Code: YVEJAZ50 VAPO
1001 Merk: BVLGARI              Naam: BLV             Volume: 75ml  Prijs: 61,52 Code: BVLBLV75
1009 Merk: Giorgio Armani       Naam: Code Donna      Volume: 75ml  Prijs: 76,00 Code: GEOCOD75
1005 Merk: Givency              Naam: Absolutely Irresistible Volume: 75ml  Prijs: 75,42 Code: GIVABS75
1002 Merk: DKNY                 Naam: Be Delicious Women Volume: 100ml Prijs: 33,65 Code: DKNBE_100 STICK
1000 Merk: Dolce & Gabbana      Naam: Light Blue      Volume: 100ml Prijs: 66,72 Code: DOLLIG100
1006 Merk: Yves Saint Laurent  Naam: Jazz            Volume: 100ml Prijs: 57,76 Code: YVEJAZ100 VAPO

Van het merk Giorgio Armani:
1008 Merk: Giorgio Armani       Naam: Code Donna      Volume: 30ml  Prijs: 39,84 Code: GEOCOD30
1003 Merk: Giorgio Armani       Naam: Code Donna      Volume: 50ml  Prijs: 59,32 Code: GEOCOD50
1009 Merk: Giorgio Armani       Naam: Code Donna      Volume: 75ml  Prijs: 76,00 Code: GEOCOD75

Alle Parfums:
1008 Merk: Giorgio Armani       Naam: Code Donna      Volume: 30ml  Prijs: 39,84 Code: GEOCOD30
1003 Merk: Giorgio Armani       Naam: Code Donna      Volume: 50ml  Prijs: 59,32 Code: GEOCOD50
1007 Merk: Ted Lapidus         Naam: Pour Elle       Volume: 50ml  Prijs: 44,48 Code: TEDPOU50
1001 Merk: BVLGARI              Naam: BLV             Volume: 75ml  Prijs: 61,52 Code: BVLBLV75
1009 Merk: Giorgio Armani       Naam: Code Donna      Volume: 75ml  Prijs: 76,00 Code: GEOCOD75
1005 Merk: Givency              Naam: Absolutely Irresistible Volume: 75ml  Prijs: 75,42 Code: GIVABS75
1000 Merk: Dolce & Gabbana      Naam: Light Blue      Volume: 100ml Prijs: 66,72 Code: DOLLIG100

Alle goedkope producten:
1008 Merk: Giorgio Armani       Naam: Code Donna      Volume: 30ml  Prijs: 39,84 Code: GEOCOD30
1010 Merk: Cacharel            Naam: Anais           Volume: 50ml  Prijs: 24,50 Code: CACANA50 VAPO
1007 Merk: Ted Lapidus         Naam: Pour Elle       Volume: 50ml  Prijs: 44,48 Code: TEDPOU50
1004 Merk: Yves Saint Laurent  Naam: Jazz            Volume: 50ml  Prijs: 39,84 Code: YVEJAZ50 VAPO
1002 Merk: DKNY                 Naam: Be Delicious Women Volume: 100ml Prijs: 33,65 Code: DKNBE_100 STICK

Duurste product:
1009 Merk: Giorgio Armani       Naam: Code Donna      Volume: 75ml  Prijs: 76,00 Code: GEOCOD75

Totale prijs: €579.05

```

SWING INTEGRATIE

Momenteel is deze applicatie een volledige console applicatie. Tijdens dit deel is het de bedoeling om de volledige applicatie te voorzien van een User Interface. Ook hierbij ga je weer te werk zoals in deel 1 van deze opdracht. Een nieuwe feature maak je aan op een nieuwe branch, kleine wijzigingen mag je rechtstreeks committen op de master.

- Toon de lijst op natuurlijke volgorde wanneer de applicatie opstart;
- De gebruiker krijgt verschillende keuzemogelijkheden om de lijst te tonen;
Bv: Gesorteerd op volume, enkel parfums tonen ...
- Toon telkens de totale prijs van de getoonde lijst en het aantal items die getoond worden.

Je bent volledig vrij in hoe je de User Interface opbouwt, zolang alles maar duidelijk is en werkt.

PS: Je gaat hiervoor wat extra code moeten schrijven om alle opdrachten tot een goed einde te brengen.

Deel 2: Maven Integratie

Wijzig je huidig project naar een Maven Project. Voeg een property toe voor de JVM version toe te voegen en zorg ervoor dat je kan werken met JAVA 8 in je project.

Voeg onderstaande dependencies toe:

1. **jUnit:**
Zie volgend deel.
2. **AssertJ:**
Zie volgend deel.
3. **Apache Commons:**
Gebruik methodes van klasse StringUtils om de getProductCode() op de correcte manier te implementeren.
4. **Log4j:**
Verwijder alle System.out.println() uit je backend code en verander deze in een Logger. Toon de info met datum in je console. Maak hiervoor een aangepast log4j.xml bestand. De output van je programma wijzigt dus een heel klein beetje.
5. **Guava:**
Maak de ArrayList aan met behulp van een methode van de Guava Library.

Doe een mvn clean install en zorg ervoor dat je BUILD successfull is. Run nadien je applicatie en zorg ervoor dat je nog steeds dezelfde output krijgt.