

Oefeningen Hoofdstuk 14

Maak een Java Project aan met naam "H14".

Per oefening maak je een aparte package. Voor oefening 1 geef je deze als naam "be.pxl.h14.oef1".

Oefening 1

Gegeven de code van de klasse FormatterApp.java: zie p205 van Hoofdstuk 12. Welke exception kan zich voordoen door het ingeven van een verkeerde datum? Vang deze exception op.

Oefening 2

Bekijk de code van Oefening2.java en run het programma enkele keren met verkeerde invoer. Welke 3 exceptions kunnen zich voordoen? Vang ze alle 3 op en zorg telkens voor een gepaste foutmelding. Indien er meerdere oplossingsmogelijkheden zijn, test alle mogelijke oplossingen uit.

Oefening 3

Geef een datum (datatype LocalDate) in via het toetsenbord van het formaat dd-MM-yyyy. Blijf een datum vragen totdat de gebruiker een juiste datum in de juiste vorm heeft ingegeven. Maak hierbij gebruik van Exceptions.

Oefening 4

Vertrek van de code van oefening1 hoofdstuk 9. Maak de klasse DatumException afgeleid van de klasse Exception. Zorg ervoor dat er een DatumException opgeworpen wordt wanneer maand een waarde kleiner dan 1 of groter dan 12 krijgt. De DatumExceptions kunnen opgeworpen worden vanuit Datum, maar ook vanuit Huwelijk. Alle Exceptionhandling gebeurt in de klasse HuwelijkApp.

Oefening 5

Maak de klasse **Persoon**. Deze klasse heeft als velden

naam (datatype String)

geboorteDatum (datatype LocalDate)

Voorzie een constructor die een waarde voor naam, geboortedag, geboortemaand en geboortjaar heeft.

```
Persoon p = new Persoon("Gert",3,4,2001);
```

Zorg ervoor dat twee personen als gelijk beschouwd worden als ze dezelfde naam hebben.

Maak de methode `toString` die de gegevens afdruckt in onderstaand formaat:

```
"Gert 3 4 2001"
```

Via de methode `aantalDagenTotVerjaardag` wordt het aantal dagen tot de volgende verjaardag van de persoon teruggegeven.

Voorzie de getter voor naam.

Maak de klasse **Vriendengroep**. Deze klasse heeft als velden

vrienden (een array met daarin waarden van datatype `Persoon`)

`MAXAANTALVRIENDEN` een constante met waarde 10

In het initialisatie-block wordt de rij vrienden aangemaakt. De inhoud van vrienden is nu een rij met daarin 10 keer de null-waarde.

Via de methode `voegVriendToe` wordt een `Persoon`-object toegevoegd aan de array vrienden. Let wel, er wordt gecontroleerd of de naam van de persoon nog niet voorkomt in de array, indien dit wel zo is dan wordt een exception opgeworpen met de boodschap "Deze naam bestaat al" gevolgd door de naam van de vriend. Verder wordt ook gekeken of er nog

een vrije positie in de rij is, indien dit niet zo is dan wordt een exception opgeworpen met de boodschap "De rij vrienden is al vol".

Maak de methode `verwijderVriend`. Deze methode neemt een `String` als argument, deze `String` bevat de naam van de te verwijderen vriend. De rij vrienden wordt doorlopen en op de index waar een `Persoon`-object met de opgegeven naam zich bevindt, wordt de waarde null geplaatst. Indien de naam niet voorkomt in de rij vrienden, wordt er een exception opgegooid met de boodschap "Deze persoon is geen vriend en kan niet verwijderd worden uit de vriendenlijst".

Voorzie de methode `toString`. Via deze methode worden alle `Persoon`-objecten in de array teruggegeven, ook wordt de naam van de persoon wiens verjaardag het dichtst bij de huidige datum ligt afgedrukt. De uitvoer van de methode `toString` is van de vorm

Gert 03 apr 2001

Koen 01 jan 1995

An 02 feb 1999

De volgende jarige is An.

In de main-method van de klasse `VriendenApp` wordt telkens een waarde voor naam, geboortedag, geboortemaand en geboortjaar gevraagd tot voor naam de waarde stop ingegeven wordt. Er wordt telkens een `Persoon`-object aangemaakt en dit object wordt toegevoegd aan een vriendengroep. Indien er 2 keer dezelfde naam wordt ingegeven, mag het programma niet stoppen. De invoer van de vrienden stopt alleen als er voor de naam de waarde stop ingegeven wordt.

Vervolgens wordt heel de vriendengroep afgedrukt adhv de `toString` methode.

Vraag vervolgens aan de gebruiker of hij vrienden wenst te verwijderen. Je blijft dit doen tot de gebruiker aangeeft dat hij wil stoppen of totdat er geen vrienden meer over zijn.

Druk vervolgens de overgebleven vriendengroep af adhv de `toString` methode of in het geval er geen vrienden meer overblijven druk een boodschap af.

Zie voorbeelden op volgende pagina's.

Voorbeeld1

naam van je vriend in

Annelies

geef geboortedatum in dag maand jaar

1 5 1999

naam

Peter

geef geboortedatum in dag maand jaar

5 7 1996

naam

Annelies

geef geboortedatum in dag maand jaar

15 8 1996

deze naam bestaat al Annelies

naam

Mo

geef geboortedatum in dag maand jaar

3 02 1994

naam

stop

Annelies 01 mei 1999

Peter 05 jul 1996

Mo 03 feb 1994

De volgende jarige is Mo

Wil je vrienden verwijderen j/n

j

Geef de naam van de vriend die je wil verwijderen

naam

Mo

Wil je vrienden verwijderen j/n

j

Geef de naam van de vriend die je wil verwijderen

naam

Miek

deze persoon is geen vriend en kan niet verwijderd worden uit de vriendenlijst

Wil je vrienden verwijderen j/n

j

Geef de naam van de vriend die je wil verwijderen

naam

Peter

Wil je vrienden verwijderen j/n

j

Geef de naam van de vriend die je wil verwijderen
naam

Annelies

Wil je vrienden verwijderen j/n

j

Er zijn geen vrienden meer over

Voorbeeld 2

naam van je vriend in

Mo

geef geboortedatum in dag maand jaar

1 8 1996

naam

Kerim

geef geboortedatum in dag maand jaar

8 12 1999

naam

Conrad

geef geboortedatum in dag maand jaar

1 2 2000

naam

stop

Mo 01 aug 1996

Kerim 08 dec 1999

Conrad 01 feb 2000

De volgende jarige is Conrad

Wil je vrienden verwijderen j/n

j

Geef de naam van de vriend die je wil verwijderen

naam

Kerim

Wil je vrienden verwijderen j/n

n

Mo 01 aug 1996

Conrad 01 feb 2000

De volgende jarige is Conrad

Extra oefeningen Hoofdstuk 14

Voor extra oefening 1 maak je een package met als naam "be.pxl.h14.exoef1".

Extraoefening 1

Gegeven de code van Opdracht5.java van de voorbeelden van Hoofdstuk 12.

Wanneer treden er exceptions op bij het runnen van dit programma?

Vang deze exceptions op met een gepaste melding.

Indien er meerdere oplossingsmogelijkheden zijn, test alle mogelijke oplossingen uit.

Extraoefening 2

Vertrek van oefening 3 uit hoofdstuk 8.

Maak de klasse NegatiefGetalException afgeleid van de klasse RuntimeException. Zorg ervoor dat er een NegatiefGetalException wordt opgegooid wanneer er een Bankrekening wordt aangemaakt met een negatief saldo of met een negatief rentepercentage.

Maak een BankrekeningApp waar je 3 bankrekeningen aanmaakt waarbij alle gegevens via het toetsenbord worden ingegeven. Wanneer een negatief saldo, negatief rentepercentage wordt ingegeven, moet de gebruiker de mogelijkheid krijgen om een nieuw saldo/ nieuw rentepercentage in te geven. Maak hierbij gebruik van de hierboven gedefinieerde exception.

Zorg ervoor dat de klasse NegatiefGetalException afgeleid wordt van de klasse Exception. Geeft dit een verschil? Waarom? Pas eventueel je code aan zodat alles blijft werken.

Voorbeeld

Geef rekeningnummer in

745-23456-789

Geef naam in

Koen

geef saldo in

15

geef rente in

-5

rente negatief bij rekening 1, opnieuw ingeven

10

Geef rekeningnummer in

895-78965-897

Geef naam in

Jan

geef saldo in

-500

geef rente in

5

saldo negatief bij rekening 2, opnieuw ingeven

500

Geef rekeningnummer in

456-987456-12

Geef naam in

Ayise

geef saldo in

895

geef rente in

2

Saldo op spaarrekening 745-23456-789 op naam van Koen bedraagt 15,00 euro

Saldo op spaarrekening 895-78965-897 op naam van Jan bedraagt 500,00 euro

Saldo op spaarrekening 456-987456-12 op naam van Ayise bedraagt 895,00 euro