

Algemene instructies

Je hebt 90 minuten om volgende oefeningen te maken **en in te dienen. Dat wil zeggen dat je best een kwartier voor het einde van het evaluatiemoment stopt om alles te zippen en te uploaden. Niet op tijd ingediend betekent 0/20. Zet een timer die duidelijk afloopt na 75 minuten om zeker te zijn dat je nog kan indienen.**

Deze oefeningen maak je als onderdeel van je bestaand project **IndividueleOefeningen**. Maak daarin een klasse EvaluatieEenHerkansing met een methode ToonSubmenu om de voorbeeldinteracties uit te voeren.

Wanneer je klaar bent, maak je een zip van je project en dien je in via toets.ap.be. Om snel je bestanden terug te vinden, kan je in Visual Studio rechts klikken op een van je .cs files of op je .sln file. Als je dan klikt op "Open Containing Folder", kom je in de juiste map.

Je mag eigen nota's, bestaande oefeningen en de cursus gebruiken, of online documentatie. Je mag niet met anderen communiceren tijdens de test.

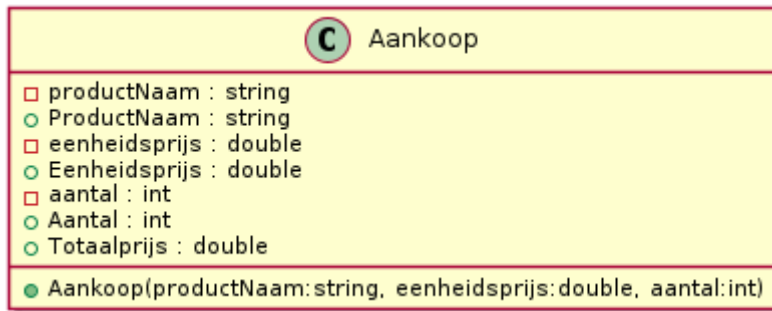
Opdracht Aankoop (6 punten)

Functionele analyse

Schrijf een eigen klasse Aankoop. Deze stelt de aankoop van een bepaald product voor. Je kan je dit inbeelden als één regel op een kasticket.

Technische analyse

Het klassendiagram ziet er als volgt uit:



Deze klasse bevat drie stukjes data. Deze worden allemaal ingesteld bij constructie en kunnen buiten de code van de klasse alleen gelezen worden. Er is geen manier om ze na aanmaak van het object van buiten de klasse aan te passen:

- Naam van het product (dit is een stuk tekst)
- Aantal aangekochte producten (hiervoor gebruik je het type int)
- Eenheidsprijs van het product (hiervoor gebruik je het type double)

Er wordt een foutmelding geprint op de console als:

- De naam van het product "" of null is
- Het aantal aangekochte producten kleiner dan 0 is
- De eenheidsprijs kleiner is dan 0.01 of groter is dan 5000

Anders wordt de gevraagde waarde gewoon opgeslagen.

Het programma hoeft niet te stoppen als er een foutmelding wordt getoond. Een ongeldige productnaam wordt "onbekend", een ongeldig aantal wordt "1" en een ongeldige eenheidsprijs wordt "1000".

Ten slotte kan voor elke aankoop de totale prijs bepaald worden via een berekende (read only) property Totaalprijs. Dit is de eenheidsprijs maal het aantal aangekochte producten.

Je moet ook een demonstratie geven van jouw klasse. Dit doe je met een methode DemonstreerAankoop, die je van uit het submenu van EvaluatieEenHerkansing kan oproepen. Deze methode maakt 4 aankopen aan:

- één aankoop met allemaal geldige waarden
- één aankoop met een ongeldige productnaam en verder geldige waarden
- één aankoop met een ongeldige eenheidsprijs en verder geldige waarden
- één aankoop met een ongeldig aantal en verder geldige waarden

Van de eerste en de vierde aankoop wordt ook de totaalprijs getoond.

Voorbeeldinteractie

Als je je demonstratiemethode oproept, krijg je iets zoals dit te zien (de waarden kunnen anders zijn voor jouw aankopen):

```
De totaalprijs van aankoop 1 is 11.98  
De productnaam moet ingevuld zijn.  
Ongeldige eenheidsprijs  
Ongeldig aantal  
De totaalprijs van aankoop 4 is 2.99
```

Opdracht Bankrekening (9.5 punten basis, 4.5 uitbreiding)

Deze oefening bestaat uit een basisgedeelte en een uitbreidingsgedeelte. De basis staat op het grootste aantal punten.

Functionele analyse

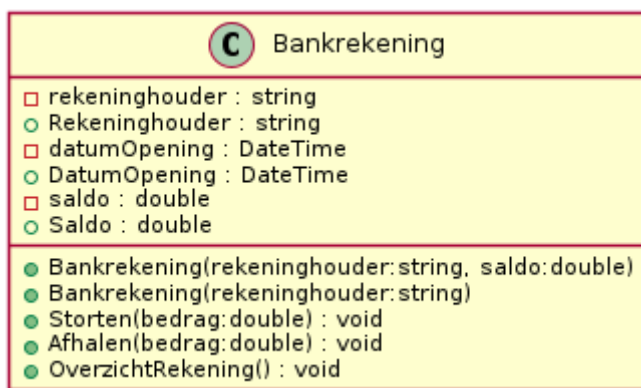
Om een bankrekening te kunnen gebruiken, maken we gebruik van een klasse Bankrekening. Daarmee kunnen we een nieuwe bankrekening maken, geld storten, geld afhalen en info over de rekening opvragen.

Technische analyse

Maak een klasse Bankrekening.

Basisaspecten

Voor de basis ziet de UML-voorstelling er zo uit:

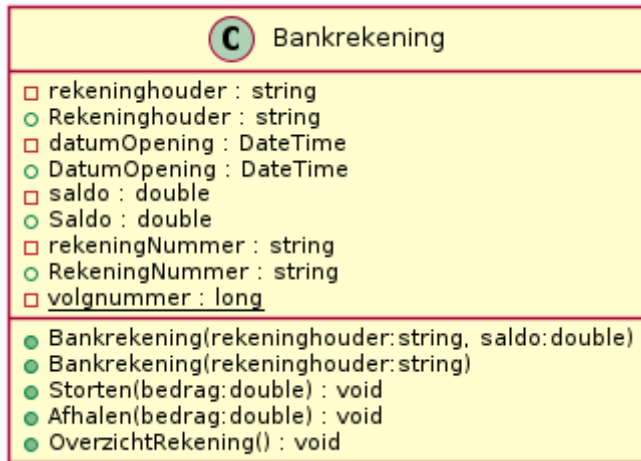


- Property `RekeningHouder`: Bij het opvragen wordt deze altijd in hoofdletters getoond. Indien bij gebruik van de setter een lege of ontbrekende string wordt ingegeven ("" of null) dan toon je "Je moet een rekeninghouder opgeven".
- Property's `DatumOpening` en `Saldo`: deze zijn read only
- Bij aanmaak van een nieuwe Bankrekening stel je waarden in voor de rekeninghouder, het saldo en de datum van opening. Deze laatste geef je de waarde van de huidige datum. Je kan ook een nieuwe bankrekening aanmaken door enkel de rekeninghouder mee te geven; stel dan een saldo in van 5 euro en maak hiervoor gebruik van chaining.
- `Storten`: verhoogt het saldo met een bepaald bedrag
- `Afhalen`: verlaagt het saldo met een bepaald bedrag. Je mag maximum 1000 euro afhalen. Indien het bedrag groter is dan 1000, dan geef je de melding: "Je mag maximum 1000 euro afhalen" en je haalt 1000 euro af.
- Met `OverzichtRekening` toon je de gegevens van de bankrekening zoals in het voorbeeld in de voorbeeldinteractie.

Uitbreiding

Controleer eerst dat de voorbeeldinteractie voor het basisgedeelte (zie beneden) in orde is voor je deze uitbreiding doet!

Breid de klasse Bankrekening uit en doe ook de nodige aanpassingen om een rekeningnummer toe te voegen. Dan ziet de UML er zo uit:



Voorzie hiervoor het volgende:

- Property `RekeningNummer`: read only. Dit geeft het volgnummer (zie hieronder) in de vorm van **exact** 12 cijfers.
- Attribueert `volgnummer`: dit heeft de beginwaarde 0 en wordt gebruikt om elke nieuwe rekening te voorzien van een rekeningnummer.
- Bij het aanmaken van een bankrekening verhoog je het volgnummer met 1
- Een update van `OverzichtRekening` zodat ook het rekeningnummer getoond wordt zoals in de tweede voorbeeldinteractie

Voorbeeldinteractie

Zet volgende code in een methode `DemonstreerBankrekening` van `EvaluatieEenHerkansing`:

```
Bankrekening mijnRekening = new Bankrekening("Bruno Peeters");
mijnRekening.Storten(100);
mijnRekening.Afhalen(10);
mijnRekening.OverzichtRekening();
Console.WriteLine();
```

```
Bankrekening rekeningCoucke = new Bankrekening("Marc Coecke", 200000.75);
rekeningCoucke.Storten(100000);
rekeningCoucke.Afhalen(2000);
rekeningCoucke.OverzichtRekening();
Console.WriteLine();
```

```
Bankrekening fouteRekening = new Bankrekening("");
```

Basis

Dit moet het volgende kunnen leveren:

Overzicht rekening van BRUNO PEETERS:
Opening rekening: donderdag 18 maart 2021
Saldo: 95 euro

Je mag maximum 1000 euro afhalen
Overzicht rekening van MARC COECKE:
Opening rekening: donderdag 18 maart 2021
Saldo: 299000,75 euro

Je moet een rekeninghouder opgeven

Uitbreiding

Overzicht rekening van BRUNO PEETERS:
Opening rekening: donderdag 18 maart 2021
Rekeningnummer: 000000000001
Saldo: 95 euro

Je mag maximum 1000 euro afhalen
Overzicht rekening van MARC COECKE:
Opening rekening: donderdag 18 maart 2021
Rekeningnummer: 000000000002
Saldo: 299000,75 euro

Je moet een rekeninghouder opgeven