

Les 1 - Klassen en objecten

Oefeningen

1. Kauwgom automaat

KauwgomAutomaat
<<Property>> -AantalBallen : int
<<Property>> -Kleur : String
<<Property>> -Vergrendeld : bool = true
+KauwgomAutomaat(aantalBallen : int)
+KauwgomAutomaat()
+IsLeeg() : bool
+VulBij(aantalBallen : int) : void
-SetAantalBallen(aantalBallen : int) : void
-SetKleur(kleur : String) : void

1. Maak een klasse KauwgomAutomaat aan volgens het DCD schema.
2. Maak 2 kauwgomautomaat objecten aan: de ene automaat is een rode kauwgom automaat die momenteel nog leeg is. De andere kauwgomautomaat is ook rood maar is al onmiddellijk gevuld met 50 balletjes.
3. Druk van beide kauwgomautomaten de kleur én het aantal balletjes af.
4. Ontgrendel de eerste kauwgom automaat.
5. Vul die bij met 20 balletjes.
6. Vergrendel de kauwgomautomaat opnieuw.
7. Druk van beide kauwgomautomaten de kleur én het aantal balletjes af.
8. Vul diezelfde kauwgomautomaat nog eens bij met 75 balletjes (in vergrendelde toestand).
9. Druk van beide kauwgomautomaten de kleur én het aantal balletjes af.

2. Fiets

Ontwerp de klasse Fiets

Zorg dat een fiets volgende eigenschappen heeft: kleur, merk, aantal versnellingen, herenfiets (ja/nee), snelheid.

Volgende functionaliteiten heeft:

- een manier om de kleur van een fiets op te vragen.
- een manier om het aantal versnellingen van een fiets te wijzigen.
- een manier om een fiets te laten versnellen, de toe te voegen snelheid wordt als extra informatie via de methode doorgegeven.

Volgende constructor heeft

- een constructor die toelaat dat de je de kleur kan bepalen van een nieuw fiets object, deze kleur wordt als extra informatie doorgegeven aan de constructor.

3. Rekening

Ontwerp de klasse rekening

Eigenschappen van een object Rekening

- een rekeningnummer voorgesteld door een geheel getal (tot 15 cijfers lang)
- het saldo van de rekening
- naam van de houder van de rekening

Constructoren

- Constructor met 2 parameters, één voor het rekeningnummer en één voor de naam van de houder.
- Constructor met 1 parameter: enkel het rekeningnummer wordt doorgegeven.
- Constructor zonder parameters

Bij creatie van een Rekening-object is het saldo altijd 0.0. Als het rekeningnummer niet meegegeven wordt, dan is de waarde 123456789, houder is “onbekend” als het niet meegegeven wordt.

Properties

- De 3 eigenschappen hebben een getter.
- De naam van de houder van de rekening kan een nieuwe waarde krijgen via een setter, rekeningnummer ook (maar deze wijziging kan alleen maar van binnen in de klasse gebeuren, andere objecten kunnen dat niet doen). Voor saldo bestaat geen setter.

Methodes

- Er is een methode om een bedrag te storten. Dit kan alleen als het te storten bedrag groter is dan 0. De methode laat weten of het gelukt is of niet.
- Er is een methode om een bedrag af te halen. Dit kan alleen als het af te halen bedrag kleiner of gelijk aan het huidige saldo van de rekening is. De methode laat weten of het gelukt is of niet.

4. Veelhoeken

4.1. Driehoek

Maak een klasse driehoek, deze heeft drie properties: ZijdeA, ZijdeB en ZijdeC. In elk van deze properties zit de lengte van hun respectievelijke zijde.

De constructor vult elk van deze properties in via een waarde die langs de parameter wordt meegegeven.

De waarde in deze properties mag niet meer aangepast worden na het aanmaken van de driehoek. De lengte van elke zijde moet groter zijn dan 0, indien de meegegeven waarde niet groter is dan 0 is de default waarde 1.

De klasse Driehoek heeft nog een methode: IsRechthoekigeDriehoek deze geeft true/false terug naargelang de zijden een rechthoekige driehoek voorstellen.

Tip Stelling van Pythagoras: In een rechthoekige driehoek is de som van de kwadraten van de lengtes van de rechthoekszijden gelijk aan het kwadraat van de lengte van de schuine zijde.

4.2. Rechthoek

De klasse heeft als properties lengte en breedte.

De constructor is zo gedefinieerd dat bij creatie van een nieuw Rechthoek-object de waarden voor de properties worden meegegeven. Er is ook een default constructor gedefinieerd. Die maakt een rechthoek aan waarvan de lengte op 10.0 wordt ingesteld en de breedte op 7.0.

Lengte en breedte moeten strikt positief zijn. Anders worden ze ingesteld op de defaultwaarde 1.0.

Verder beschikt de klasse over 2 extra methodes BerekenOmtrek en BerekenOppervlakte, die respectievelijk de omtrek en de oppervlakte van de rechthoek berekenen.

4.3. VeelhoekenApp

Maak nu gebruik van de 2 klassen uit vorige oefening om volgende applicatie uit te werken.

Allereerst vragen we aan de gebruiker of hij een vorm wil ingeven.

- Bij keuze 0: applicatie stopt met het uitschrijven van de nodig info.
- Bij keuze 1: applicatie vraagt om lengte en breedte van een rechthoek in te geven.
- Bij keuze 2: applicatie vraagt om 3 zijden van een driehoek in te geven.

Na het ingeven van een rechthoek of driehoek krijg je terug de kans om te stoppen of om een nieuwe vorm in te geven. Wat je ook koos, op het eind wordt volgende informatie uitgeschreven:

Overzicht vormen:

Totaal aantal vormen:

Aantal rechthoeken met opp > 50:

Aantal rechthoekige driehoeken:

Hieronder zie je 3 keer een verschillende uitvoer. Zorg ervoor dat je uitvoer ook volgens deze manier geformatteerd is.

Om te weten of de oppervlakte van de rechthoek groter is dan 50 maak je gebruik van de voorziene methode uit de klasse. Het bepalen of een driehoek rechthoekig is, hebben we ook al uitgewerkt in de klasse. Communiceer dus met het object dat je aanmaakt om deze informatie te krijgen.

Probeer niet alles in 1 keer uit te werken. Doe het stap per stap. Vraag bijvoorbeeld eerst gewoon de gegevens van een rechthoek op en schrijf de uitvoer uit. Als dat werkt kan je de gegevens van 1 rechthoek én 1 driehoek opvragen, daarna de keuze geven en op het eind de herhaling. Probeer telkens stukje per stukje te laten werken. Zo kom je snelst tot een werkende, goede oplossing.

Voorbeeld uitvoer 1

```
Rechthoeken en driehoeken
-----
Wil je graag nog een vorm ingeven (1=een rechthoek, 2=een driehoek, 0=nee)? 0
Overzicht vormen:
Totaal aantal vormen: 0
Aantal rechthoeken met opp > 50: 0
Aantal rechthoekige driehoeken: 0
```

Voorbeeld uitvoer 2

Rechthoeken en driehoeken

```
-----  
Wil je graag nog een vorm ingeven (1=een rechthoek, 2=een driehoek, 0=nee)? 1  
Geef de lengte van de rechthoek: 8  
Geef de breedte van de rechthoek: 6,5  
Wil je graag nog een vorm ingeven (1=een rechthoek, 2=een driehoek, 0=nee)? 1  
Geef de lengte van de rechthoek: 2  
Geef de breedte van de rechthoek: 0,5  
Wil je graag nog een vorm ingeven (1=een rechthoek, 2=een driehoek, 0=nee)? 0  
Overzicht vormen:  
Totaal aantal vormen: 2  
Aantal rechthoeken met opp > 50: 1  
Aantal rechthoekige driehoeken: 0
```

Voorbeeld uitvoer 3

Rechthoeken en driehoeken

```
-----  
Wil je graag nog een vorm ingeven (1=een rechthoek, 2=een driehoek, 0=nee)? 1  
Geef de lengte van de rechthoek: 2,5  
Geef de breedte van de rechthoek: 2  
Wil je graag nog een vorm ingeven (1=een rechthoek, 2=een driehoek, 0=nee)? 2  
Geef de lengte van zijde A: 3  
Geef de lengte van zijde B: 4  
Geef de lengte van zijde C: 5  
Wil je graag nog een vorm ingeven (1=een rechthoek, 2=een driehoek, 0=nee)? 0  
Overzicht vormen:  
Totaal aantal vormen: 2  
Aantal rechthoeken met opp > 50: 0  
Aantal rechthoekige driehoeken: 1
```

5. Fitness

We ontvangen gegevens van een aantal loopbanden van een fitness-centra die bedoeld waren om te importeren in een SQL databank.

Het zijn geavanceerde toestellen waarbij de snelheid wordt bijgehouden telkens de gebruiker deze aanpast. Naast het klantnummer wordt ook de datum en starttijd bijgehouden, de gemiddelde snelheid en de totale looptijd. Daarnaast wordt voor elk interval, de tijdsduur en de snelheid gedurende dit tijdsinterval opgeslagen. In het voorbeeld is dit duidelijk te zien, de loper heeft zijn loopsessie opgedeeld in 4 intervallen, een eerste interval van iets meer dan 20 min

aan een snelheid van 20 km/h, daarna 20 min rustig aan (10 km/h), weer een snel interval aan 19 km/h en een laatste 20 minuten aan ongeveer 15km/h.



De gegevens die we binnenkrijgen zien er als volgt uit. Het zijn insert-statements met daarin alle waarden voor de verschillende intervallen van een loopessie. De statements mag je zien als gewone tekst die we niet nodig hebben, enkel de informatie die er in staat hebben we nodig. De samenwerking tussen C# en databases zien we pas later in het semester.

```
insert into runningession values(1, '2021-11-14 11:22:00', 6, 63, 14.110714285714286, 1, 945, 5.7);
insert into runningession values(1, '2021-11-14 11:22:00', 6, 63, 14.110714285714286, 2, 945, 20.9);
insert into runningession values(1, '2021-11-14 11:22:00', 6, 63, 14.110714285714286, 3, 945, 7.9);
insert into runningession values(1, '2021-11-14 11:22:00', 6, 63, 14.110714285714286, 4, 960, 21.6);
insert into runningession values(2, '2021-07-06 09:24:00', 6, 20, 13.624999999999998, 1, 300, 12.7);
insert into runningession values(2, '2021-07-06 09:24:00', 6, 20, 13.624999999999998, 2, 300, 20.9);
insert into runningession values(2, '2021-07-06 09:24:00', 6, 20, 13.624999999999998, 3, 300, 9.7);
insert into runningession values(2, '2021-07-06 09:24:00', 6, 20, 13.624999999999998, 4, 300, 11.2);
insert into runningession values(3, '2021-06-17 15:32:00', 6, 117, 13.977777777777778, 1, 780, 38.4);
```

Op elke lijn vinden we de volgende gegevens en validatie regels:

- SessieNr : numeriek, moet steeds groter dan 0 zijn.
- Datum (en tijdstip).
- KlantNr : numeriek, moet altijd groter dan 0 zijn.
- De totale trainingsduur van de sessie (minuten). Een sessie duurt minstens 5 minuten en mag niet langer duren dan 3 uur.
- De gemiddelde snelheid van de loopessie (km/h). De snelheid mag niet lager zijn dan 5 en niet hoger dan 22 km/h.
- Het sequentienummer van het loopinterval is steeds een positief getal.

- De tijdsduur van het loopinterval (in seconden). Een interval duurt minstens 5 seconden en maximaal 3 uur.
- De loopsnelheid van het tijdsinterval. Net zoals bij de gemiddelde snelheid tussen 5 en 22 km/h.

De eerste sessie (sessieNr=1) vond plaats op 14/11/2021 om 11:22 en behoort aan klantnummer 6. De gemiddelde snelheid voor de loopsessie was 14.11 km/h en duurde in totaal 63 minuten. Deze sessie bestond uit 4 intervallen (seqNr van 1 tot 4). De eerste 3 intervallen duurden 945 seconden en het laatste 960 seconden.

Wat wordt er nu van ons verwacht ?

De gegevens moeten worden ingelezen en gecontroleerd (zie de voorwaarden bij elk van de parameters). Als er niet wordt voldaan aan deze voorwaarden dan moet de rij worden genegeerd en schrijven we in een logbestand wat er fout is.

Bijvoorbeeld :

```
|LoopInterval-ZetSnelheid|Snelheid:38,4
|LoopTraining-ZetSessieNummer|sessieNummer:-3
```

We moeten ook een applicatie maken die op een efficiënte manier de volgende twee vragen snel kan beantwoorden :

1. Geef voor een klant het overzicht van de trainingen weer.
2. Geef voor een bepaalde dag alle trainingen weer.

Voorbeeld van de output kan zijn :

De drie trainingssessies van klantnummer 154.

Begin met het inlezen van het kleinste bestand (*insertRunningTest.txt*), hier zit minder data in en heeft ook minder rijen met fouten. Eenmaal de applicatie volledig werkt met dit bestand kan je het grotere tekstbestand (*insertRunning.sql*) gebruiken en kan je je toespitsen op het opvangen van fouten en optimalisatie van de snelheid van de applicatie.

```
Microsoft Visual Studio Debug Console

Sessie:7531,19/01/2022 16:09:00,Klant:154,Duur:31,Snelheid:13,012903225806449,Intervals:3
  SeqNr:1,Snelheid:13,Duur:620
  SeqNr:2,Snelheid:5,6,Duur:620
  SeqNr:3,Snelheid:19,2,Duur:660
Sessie:7532,12/10/2021 13:05:00,Klant:154,Duur:60,Snelheid:11,7,Intervals:3
  SeqNr:1,Snelheid:8,2,Duur:1200
  SeqNr:2,Snelheid:10,8,Duur:1200
  SeqNr:3,Snelheid:16,1,Duur:1200
Sessie:7533,2/12/2021 12:53:00,Klant:154,Duur:113,Snelheid:14,377581120943953,Intervals:3
  SeqNr:1,Snelheid:13,7,Duur:2260
  SeqNr:2,Snelheid:14,3,Duur:2260
  SeqNr:3,Snelheid:15,Duur:2280

C:\VisualStudioProjects\SolutionFitnessRunning\FitnessApplicatie\bin\Debug\net5.0\FitnessApplicatie.exe (process 22604)
exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console
when debugging stops.
Press any key to close this window . . .
```

Alle trainingssessies op 1/12/2021.

```
Microsoft Visual Studio Debug Console

Sessie:1151,1/12/2021 9:28:00,Klant:31,Duur:24,Snelheid:13,2,Intervals:8
  SeqNr:1,Snelheid:19,6,Duur:180
  SeqNr:2,Snelheid:13,3,Duur:180
  SeqNr:3,Snelheid:7,6,Duur:180
  SeqNr:4,Snelheid:8,6,Duur:180
  SeqNr:5,Snelheid:21,Duur:180
  SeqNr:6,Snelheid:14,Duur:180
  SeqNr:7,Snelheid:9,7,Duur:180
  SeqNr:8,Snelheid:11,8,Duur:180
Sessie:1328,1/12/2021 12:53:00,Klant:33,Duur:84,Snelheid:14,824999999999998,Intervals:4
  SeqNr:1,Snelheid:12,8,Duur:1260
  SeqNr:2,Snelheid:15,2,Duur:1260
  SeqNr:3,Snelheid:15,4,Duur:1260
  SeqNr:4,Snelheid:15,9,Duur:1260
Sessie:1423,1/12/2021 20:05:00,Klant:36,Duur:41,Snelheid:12,404878048780487,Intervals:3
  SeqNr:1,Snelheid:14,3,Duur:820
  SeqNr:2,Snelheid:5,5,Duur:820
  SeqNr:3,Snelheid:17,Duur:840
Sessie:1614,1/12/2021 10:09:00,Klant:38,Duur:82,Snelheid:16,83089430894309,Intervals:3
  SeqNr:1,Snelheid:13,1,Duur:1640
  SeqNr:2,Snelheid:16,7,Duur:1640
  SeqNr:3,Snelheid:20,2,Duur:1680
Sessie:1648,1/12/2021 18:58:00,Klant:39,Duur:36,Snelheid:15,325000000000001,Intervals:4
  SeqNr:1,Snelheid:16,5,Duur:540
  SeqNr:2,Snelheid:17,5,Duur:540
  SeqNr:3,Snelheid:16,Duur:540
  SeqNr:4,Snelheid:11,3,Duur:540
Sessie:2317,1/12/2021 17:26:00,Klant:47,Duur:55,Snelheid:14,936818181818182,Intervals:4
  SeqNr:1,Snelheid:7,1,Duur:825
  SeqNr:2,Snelheid:21,5,Duur:825
```

De data wordt ter beschikking gesteld op Chamilo.