

Happy New Year

Tip! Eerst lezen, dan pas programmeren! Gun jezelf 5 minuten om dit document even door te lezen, zodat je weet wat je te doen staat. Daar is meer dan genoeg tijd voor.

Regels en afspraken

Voor deze toets heb je totaal 100 minuten de tijd (2 lesuren). Het is hierbij toegestaan internet te gebruiken en zaken op te zoeken, maar gebruik van AI (bijvoorbeeld ChatGPT) is ten strengste verboden. Probeer gewoon zoveel mogelijk uit je hoofd te doen. **Belangrijk!** Maak notities van de zaken die je hebt opgezocht, want dit heb je voor je reflectie nodig.

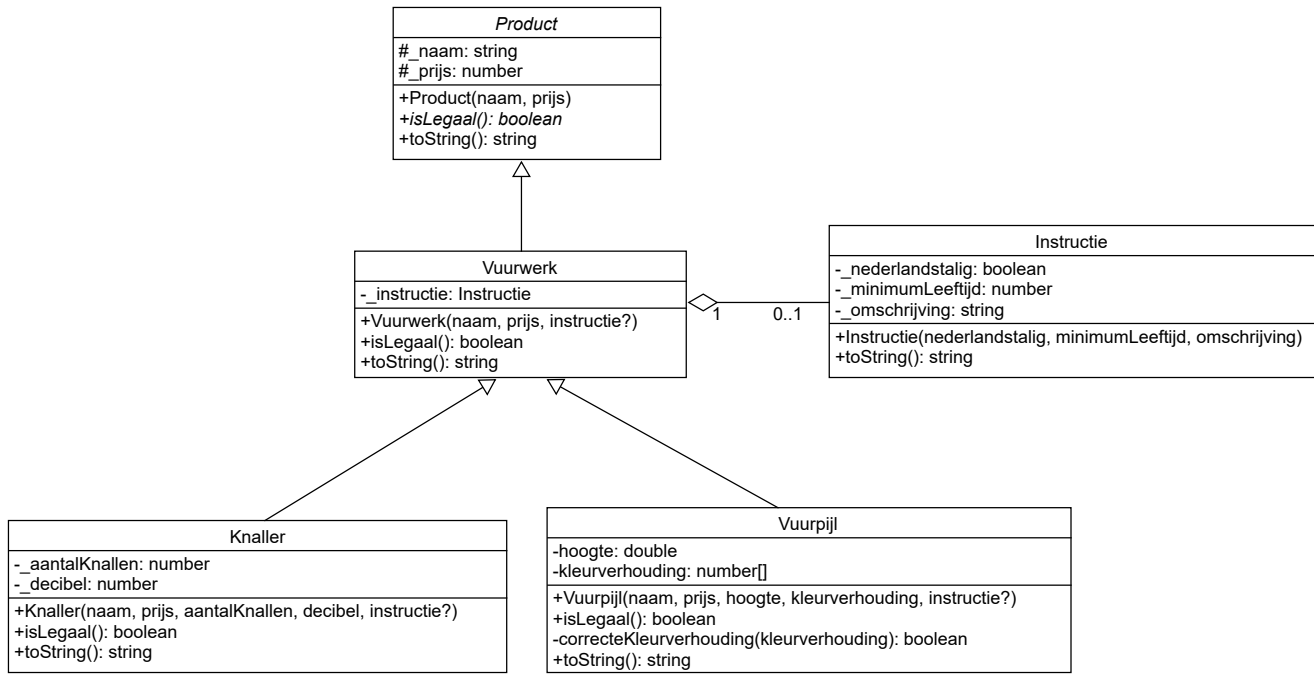
Als je klaar bent met de toets of de tijd is om, bepaal dan op basis van de normering wat je cijfer zou kunnen zijn. Je docent zal de toets na een korte pauze behandelen, zodat je kunt controleren of je implementatie ook klopt. Bepaal achteraf nogmaals je cijfer. **Belangrijk** Maak een kopie van de **src**-map voor je deze aan de hand van de uitleg eventueel gaat aanpassen, want dit heb je voor je reflectie nodig.

Over de toets maak je een reflectie die via Scorion naar je docent stuurt, dit is een *verplicht datapunt*. In deze reflectie vermeld je:

- Wat je ervaring was met de toets. Wat vond je makkelijk, wat vond je moeilijk. En waarom vond je dat?
- De onderdelen die je hebt opgezocht op internet en hoe dit je (eventueel) verder heeft geholpen.
- Het cijfer dat je dacht te hebben en wat het daadwerkelijk was. Waar komt (eventueel) dat verschil vandaan?
- Wat je volgende stappen zijn om een soortgelijke toets in de toekomst (nog) beter te doen. Welke kennis moet je nog bijschaven?
- Een kopie van je broncode voordat je deze ging aanpassen na de uitleg.

Inleiding

De opdracht bestaat uit een aantal stappen. Om deze stappen te kunnen testen is al veel testcode gemaakt in de `index.ts` (niet weergegeven in het class diagram). Maak hiervan gebruik! De voorbeeldoutput is ook gebaseerd op deze testcode.



Bovenstaand class diagram moet worden geïmplementeerd. Bekijk het goed, en lees alle opmerkingen hieronder goed door!

Let op:

- Er zijn geen getters en setters getoond. Voeg zelf getters en setters toe die je denkt nodig te hebben. Meer mag, maar hoeft niet.
- Zorg dat je `toString()` methodes **soortgelijke output** genereren als in de outputvoorbeelden.
- De classes **Instructie** en **Product** zijn aanwezig, maar nog incompleet. Deze zal je moeten aanpassen (zie stap 1a en 1b).
- De subclasses **Vuurwerk**, **Knaller** en **Vuurpijl** zijn niet aanwezig. Deze zal je in het geheel moeten maken (zie stap 1c, 2 en 3).
- Je hoeft alleen TSDoc te schrijven voor de class **Instructie**. Voor de andere classes hoef je geen TSDoc te schrijven.
- Als je zelf nog zaken wil toevoegen die niet in het class diagram staan (omdat je denkt dat ze nodig zijn), geef dat dan duidelijk met commentaar aan in de code.

Deel 1: Bouw klassenstructuur

De volgende stappen moet je in de gegeven volgorde doorlopen.

Stap 0: Verander de nieuwjaarswens

Voer `npm install` uit om alle benodigde pakketten binnen te halen. Start vervolgens de applicatie met `npm start`. De output moet er als volgt uit zien.

```
Happy new year from: <naam>, <studentnummer>, <klas>
```

Verander daarna de nieuwjaarsgroet zodat je eigen naam, je studentnummer en je klas verschijnt.

Stap 1a: Class `Instructie` (10 pt)

De class `Instructie` is vrijwel geheel af. Je moet alleen twee dingen toevoegen:

- Voeg visibility modifiers toe, zodat de regels voor data encapsulation correct zijn toegepast.
- Voeg TSDoc toe.

Stap 1b: Class `Product` (5 pt)

Maak de abstract class `Product` af volgens het class diagram. Let erop dat de methode `isLegaal()` abstract is.

Stap 1c: Class **Vuurwerk** (15 pt)

Maak de class **Vuurwerk**. Houd hierbij rekening met de volgende eisen:

- De **instructie** in de constructor is optioneel.
- De methode **isLegaal()** geeft aan of het vuurwerk legaal is. Voor al het vuurwerk in deze applicatie geldt: vuurwerk is legaal als het een instructie heeft, en als deze instructie Nederlandstalig is.
- Voor de methode **toString()** geldt:
 - Alle attributen moeten op een nieuwe regel staan.
 - Alle attributen van de superclass moeten ook verschijnen. Dit laat dus ook de naam en de prijs zien.
 - Voor de prijs moet het euro-teken staan. Gebruik hiervoor indien nodig **Product.EURO**.
 - Indien er geen Instructie is moet er "ontbreekt" staan.
 - Laat ook zien of het vuurwerk legaal of illegaal is.
- Test je code door in de **index.ts** alle code met TODO stap 1 te "uncommenten".

De output moet er als volgt uitzien.

```
--- STAP 1 ---
Naam: Veiligheidsbril
    Prijs: €2.50
    Instructie: Nederlandstalig=true, leeftijd=6, omschrijving=Draag bij
aansteken
    Legaal: true
Naam: Safety glass
    Prijs: €2.50
    Instructie: Nederlandstalig=false, leeftijd=6, omschrijving=Wear before
ignition
    Legaal: false
Naam: Aansteeklont zonder instructie
    Prijs: €0.25
    Instructie: ontbreekt
    Legaal: false
```

Stap 2: Subclass **Knaller** (15 pt)

Maak de subclass **Knaller**. Houd hierbij rekening met de volgende eisen:

- Override de methode `isLegaal()`. Een extra eis voor een Knaller is dat deze legaal is als het aantal decibel niet hoger is dan 120.
- Voor de methode `toString()` geldt:
 - Alle attributen moeten op een nieuwe regel staan.
 - Alle attributen van de superclass moeten ook verschijnen. Dit laat dus ook zien of het vuurwerk legaal of illegaal is.
- Test je code door in de `index.ts` alle code met TODO stap 2 te "uncommenten".

De output moet er als volgt uitzien.

```
--- STAP 2 ---
Naam: Celebration Crackers
    Prijs: €10.00
    Instructie: Nederlandstalig=false, leeftijd=21, omschrijving=Keep minimum
10 ft distance
    Legaal: false
    Aantal knallen: 777
    Decibel: 75
Naam: Peking Rol
    Prijs: €45.00
    Instructie: Nederlandstalig=true, leeftijd=21, omschrijving=Houd minimaal
5 meter afstand
    Legaal: true
    Aantal knallen: 500
    Decibel: 120
Naam: Shanghai Rol
    Prijs: €85.00
    Instructie: Nederlandstalig=true, leeftijd=21, omschrijving=Houd minimaal
5 meter afstand
    Legaal: false
    Aantal knallen: 1000
    Decibel: 125
Naam: Hongkong Rol
    Prijs: €82.50
    Instructie: ontbreekt
    Legaal: false
    Aantal knallen: 1000
    Decibel: 100
```

Stap 3: Subclass `Vuurpijl` (15 pt)

Codeer de subclass `Vuurpijl`. Houd hierbij rekening met de volgende eisen:

- Het attribuut `_hoogte` geeft in meters weer hoe hoog de pijl de lucht in schiet.
- Het attribuut `_kleurverhouding` is een array waarin van de 3 kleuren rood, groen en blauw de percentages staan. Zo is de kleur geel gecodeerd met het volgende array `[50, 50, 0]`, dat wil zeggen 50% rood, 50% groen en 0% blauw. Dit werkt natuurlijk alleen maar wanneer de som van de percentages 100 is. De methode `correcteKleurverhouding(kleurverhouding)` controleert of de percentages samen 100 zijn. Gebruik deze methode in je constructor. Als de aan de constructor meegegeven kleur niet correct is, moet een foutmelding worden afgedrukt, en wordt de kleur op rood gezet `[100, 0, 0]`. Gebruik voor deze foutmelding `console.error` in plaats van `console.log`.
- Override de methode `isLegaal()`. Een extra eis voor een `Vuurpijl` is dat deze legaal is wanneer de instructie aangeeft dat de leeftijd minimaal 16 jaar is.
- Voor de methode `toString()` geldt:
 - Alle attributen moeten op een nieuwe regel staan.
 - Alle attributen van de superclass moeten ook verschijnen. Dit laat dus ook zien of het vuurwerk legaal of illegaal is.
- Test je code door in de `index.ts` alle code met TODO stap 3 te "uncommenten".

De output moet er als volgt uitzien.

--- STAP 3 ---

Naam: Cruise Rocket

Prijs: €2.50

Instructie: Nederlandstalig=true, leeftijd=10, omschrijving=Niet in de hand houden

Legaal: false

Hoogte: 40.0 meter

Kleuren:

ROOD: 50%

GROEN: 25%

BLAUW: 25%

--> FOUT: Onjuiste kleurverhouding, kleur wordt rood

Naam: Killing Arrow

Prijs: €4.25

Instructie: Nederlandstalig=true, leeftijd=16, omschrijving=Niet in de hand houden

Legaal: true

Hoogte: 40.0 meter

Kleuren:

ROOD: 100%

GROEN: 0%

BLAUW: 0%

--> FOUT: Onjuiste kleurverhouding, kleur wordt rood

Naam: Magic Sky

Prijs: €2.75

Instructie: Nederlandstalig=false, leeftijd=20, omschrijving=Keep minimum 10 ft distance

Legaal: false

Hoogte: 40.0 meter

Kleuren:

ROOD: 100%

GROEN: 0%

BLAUW: 0%

Naam: Golden Sky

Prijs: €3.25

Instructie: ontbreekt

Legaal: false

Hoogte: 40.0 meter

Kleuren:

ROOD: 50%

GROEN: 50%

BLAUW: 0%

Deel 2: Gebruik klassenstructuur

De volgende stappen kun je in elke volgorde doorlopen, gegeven dat je stap 1 tot en met 3 hebt uitgevoerd.

Stap 4: Lijst met vuurwerk (15 pt)

- Zorg dat stappen 1 tot en met 3 zijn uitgevoerd, want dan is de pakket-variabele gevuld met allerlei vuurwerk.
- Maak in de `index.ts` een methode met de volgende signatuur:

```
function toonVuurwerk(pakket: Vuurwerk[]): void
```

Deze methode moet alle soorten vuurwerk uit het pakket afdrukken. Aan het eind moet de totaalprijs (de som van de afzonderlijke prijzen van het vuurwerk) worden afgedrukt.

- Test je code door in de `index.ts` alle code met TODO stap 4 te "uncommenten".

De output moet er als volgt uitzien.

```
--- STAP 4 ---
Naam: Veiligheidsbril
    Prijs: €2.50
    Instructie: Nederlandstalig=true, leeftijd=6, omschrijving=Draag bij
aansteken
    Legaal: true

Naam: Safety glass
    Prijs: €2.50
    Instructie: Nederlandstalig=false, leeftijd=6, omschrijving=Wear before
ignition
    Legaal: false

... (hier nog veel meer vuurwerk) ...

Naam: Golden Sky
    Prijs: €3.25
    Instructie: ontbreekt
    Legaal: false
    Hoogte: 40.0 meter
    Kleuren:
        ROOD: 50%
        GROEN: 50%
        BLAUW: 0%

Kosten vuurwerkpakket: €240.50
```


Stap 5: Toon instructies (15 pt)

- Zorg dat stappen 1 tot en met 3 zijn uitgevoerd, want dan is de pakket-variabele gevuld met allerlei vuurwerk.
- Maak in `index.ts` een methode met de volgende signatuur:

```
function printHardeKnallers(pakket: Vuurwerk[], maxDecibel: number): void
```

Deze methode moet de instructie details afdrukken van het gekozen vuurwerk. Het gekozen vuurwerk wordt bepaald door de `index`.

Indien het uitgekozen vuurwerk geen instructie heeft moet de tekst "Instructie ontbreekt." worden afgedrukt. Indien de index buiten de grenzen van de lijst ligt moet de tekst "Index valt buiten grenzen." worden afgedrukt. Gebruik voor deze foutmeldingen `console.error` in plaats van `console.log`.

- Test je code door in de `index.ts` alle code met TODO stap 5 te "uncommenten".

De output moet er als volgt uitzien.

```
--- STAP 5 ---  
Index valt buiten grenzen.  
Nederlandstalig=false, leeftijd=21, omschrijving=Keep minimum 10 ft distance  
Instructie ontbreekt.  
Index valt buiten grenzen.
```

Stap 6: Print harde knallers (10 pt)

- Zorg dat stappen 1 tot en met 3 zijn uitgevoerd, want dan is de pakket-variabele gevuld met allerlei vuurwerk.
- Maak in de `index.ts` een methode met de volgende signatuur:

```
function printHardeKnallers(pakket: Vuurwerk[], maxDecibel: number): void
```

Deze methode moet van de lijst alle knallers printen die harder zijn dan `maxDecibel`. Gebruik `instanceof` om te controleren welke instanties van het type `Knaller` zijn.

- Test je code door in de `index.ts` alle code met TODO stap 6 te "uncommenten".

De output moet er als volgt uitzien.

```
--- STAP 6 ---
Naam: Peking Rol
    Prijs: €45.00
    Instructie: Nederlandstalig=true, leeftijd=21, omschrijving=Houd minimaal
5 meter afstand
    Legaal: true
    Aantal knallen: 500
    Decibel: 120
Naam: Shanghai Rol
    Prijs: €85.00
    Instructie: Nederlandstalig=true, leeftijd=21, omschrijving=Houd minimaal
5 meter afstand
    Legaal: false
    Aantal knallen: 1000
    Decibel: 125
```

Richtlijnen bij coderen

- Zorg ervoor dat je naam en het doel van het programma bovenaan elke module staan.
- Gebruik correcte inspringing (indentation).
- Gebruik `let` en `const` correct.
- Voeg waar nodig commentaar toe dat inzicht geeft in je code.
- Zorg ervoor dat je geen fouten (gele/rode markeringen) in je uiteindelijke code hebt.

Normering

De toets wordt beoordeeld op basis van de onderstaande verdeling. Zorg ervoor dat alle stappen correct zijn uitgevoerd.

Normering per onderdeel

Stap 1a: Class **Instructie** (10 pt)

- Visibility modifiers (5 pt): Correct gebruik van private/public.
- TSDoc (5 pt): TSDoc voor class (doel en @author) en voor de methodes (doel, @param, @return).

Stap 1b: Class **Product** (5 pt)

- Abstract class (1 pt): Correct gebruik van abstract.
- Constructor (2 pt): Constructor volgens class diagram.
- Abstracte methode `isLegaal()` (2 pt): Correct gebruik van abstract.

Stap 1c: Class **Vuurwerk** (15 pt)

- Attriboot + visibility modifier (2 pt): Attriboot en visibility modifier volgens class diagram.
- Constructor met optionele `instructie`. (5 pt): Constructor volgens class diagram.
- Methode `isLegaal()` (3 pt): Alleen `true` wanneer instructie aanwezig en nederlandstalig is.
- Methode `toString()` (5 pt): Geeft exact de output zoals gevraagd.

Stap 2: Subclass **Knaller** (15 pt)

- Attributen + visibility modifiers (2 pt): Attributen en visibility modifiers volgens class diagram.
- Constructor (3 pt): Constructor volgens class diagram.
- Methode `isLegaal()` (5 pt): Alleen `true` wanneer de parent class een `true` geeft én decimal kleiner of gelijk is aan 120.
- Methode `toString()` (5 pt): Geeft exact de output zoals gevraagd.

Stap 3: Subclass **Vuurpijl** (15 punten)

- Attributen + visibility modifiers (2 pt): Attributen en visibility modifiers volgens class diagram.
- Constructor (3 pt): Constructor volgens class diagram, roept `correcteKleurverhouding` aan en toont foutmelding met `console.error`. Stelt kleur op 100% rood in bij foutmelding.
- Methode `isLegaal()` (3 pt): Alleen `true` wanneer de parent class een `true` geeft én de minimale leeftijd 16 is.
- Methode `correcteKleurverhouding()` (4 pt): Alleen `true` wanneer het totaal exact 100 is.
- Methode `toString()` (3 pt): Geeft exact de output zoals gevraagd.

Stap 4: Lijst met vuurwerk (15 punten)

- Lijst met vuurwerk tonen (10 pt): Correcte weergave van alle objecten zoals getoond in gegeven output.
- Totaalprijs (5 pt): Correcte berekening en weergave van totaalprijs.

Stap 5: Toon instructies (15 punten)

- Omgaan met incorrecte indexen (5 pt): Ongeldige index geeft foutmelding "Index valt buiten grenzen." via `console.error`.
- Omgaan met geen instructies (5 pt): Ontbreken van instructies geeft foutmelding "Instructie ontbreekt" via `console.error`.
- Instructies tonen (5 pt): Correcte weergave van alle instructies zoals getoond in gegeven output.

Stap 6: Print harde knallers (10 punten)

- Alleen knallers tonen (5 pt): Correcte filtering van objecten met `instanceof`.
- Alleen harder dan maxDecibel tonen (5 pt): Alleen knallers boven opgegeven decibelniveau.

Aftrek i.v.m. code conventions (10 punten)

Bijvoorbeeld i.v.m. magic numbers, afwijken van naamgeving conventies, verkeerd inspringen, ESLint foutmeldingen, enzovoorts. -1 pt per fout. Nooit meer dan -10 pt totaal.

Cijferberekening

Het cijfer wordt berekend als volgt:

```
Cijfer = Totaal behaalde punten / 10
```

Als het programma niet compileert of opstart, ontvang je een 1.