



Java

Basissyllabus

Egon Pas

1 Programmeren

- 1.1 Hoe werkt een computer?
- 1.2 Bits en bytes
 - 1.2.1 Het binair concept
 - 1.2.2 Binair naar Decimaal
 - 1.2.3 Decimaal naar Binair
 - 1.2.4 Binaire bewerkingen
 - 1.2.5 Negatieve binaire getallen
 - 1.2.6 Van bits naar bytes
 - 1.2.7 Binair naar Octaal en Hexadecimaal
- 1.3 Wat is een programmeertaal?
 - 1.3.1 Wat is programmeren?
 - 1.3.2 Geschiedenis
 - 1.3.3 Gecompileerde programmeertalen
 - 1.3.4 Geïnterpreteerde programmeertalen

2 Java

- 2.1 Geschiedenis
- 2.2 Releases
- 2.3 Gebruik
- 2.4 Over de Java-technologie
 - 2.4.1 De programmeertaal Java
 - 2.4.1.1 Wat gebeurt er nu concreet bij een Java programma?
 - 2.4.1.2 Snelheid van Java
 - 2.4.1.3 Kenmerken van Java
 - 2.4.2 Het Java Platform
- 2.5 Installatie JDK 6
- 2.6 Ons eerste Java-programma
 - 2.6.1 Klasse schrijven
 - 2.6.2 Klasse compileren
 - 2.6.3 Programma uitvoeren



3 Object-oriëntatie: inleiding

- 3.1 Wat is goede object-georiënteerde software?
- 3.2 Gestructureerd programmeren versus object-georiënteerd programmeren
 - 3.2.1 Gestructureerd programmeren
 - 3.2.2 Object georiënteerd programmeren
- 3.3 Objecten: Eigenschappen en gedragingen
 - 3.3.1 Eigenschappen
 - 3.3.2 Gedragingen
- 3.4 Basis OO-principes
 - 3.4.1 Inkapseling (encapsulation)
 - 3.4.2 Hoge specificiteit (high cohesion)
 - 3.4.3 Beperkte koppeling (loose coupling)

4 Klassen en objecten

- 4.1 Wat zijn klassen en objecten?
- 4.2 Een Java klasse
 - 4.2.1 Naamgeving
 - 4.2.1.1 Volgens de syntax
 - 4.2.1.2 Volgens de conventies
 - 4.2.2 Uitleg over de keywords
 - 4.2.3 Signatuur en body
- 4.3 Regels voor een Java-bronbestand
- 4.4 Een klasse schrijven
- 4.5 Instantiëren van een klasse
 - 4.5.1 Schrijven van een opstartklasse
 - 4.5.2 Een andere klasse gebruiken
- 4.6 Oefeningen

5 Variabelen en operatoren

- 5.1 Variabelen
 - 5.1.1 Wat is een variabele?
 - 5.1.2 Declaratie van een variabele
 - 5.1.3 Naamgeving



- 5.1.4 Type
- 5.2 Primitieve types
 - 5.2.1 Logische type (boolean)
 - 5.2.2 Karaktertype (char)
 - 5.2.3 Gehele getallen (byte, short, int, long)
 - 5.2.4 Reële getallen (float, double)
 - 5.2.5 Keuze en overzicht primitieve gegevenstypes
 - 5.2.6 String
- 5.3 Initialisatie
 - 5.3.1 Standaardwaarden
 - 5.3.2 Initialisatie via een literal
 - 5.3.3 Initialisatie via een variabele
 - 5.3.4 Constanten
- 5.4 Type casting
 - 5.4.1 Primitieve typen
 - 5.4.2 Objecten
- 5.5 Scope
 - 5.5.1 Instantievariabelen
 - 5.5.2 Lokale variabelen
 - 5.5.3 Globale variabelen
 - 5.5.4 Klassevariabelen
 - 5.5.5 Parameters
- 5.6 Variabelen achter de schermen
- 5.7 Operatoren
 - 5.7.1 Rekenkundige operatoren
 - 5.7.1.1 Overzicht
 - 5.7.1.2 Primitieve types en rekenkundige bewerkingen
 - 5.7.1.3 Verkorte notatie
 - 5.7.1.4 Increment & decrement
 - 5.7.1.5 String en operatoren
 - 5.7.2 Relationale operatoren
 - 5.7.3 Conditionele operatoren
 - 5.7.4 Logische operatoren



5.7.5 Bit-operatoren

5.7.6 Shift-operatoren

5.7.6.1 (Signed) Shift left

5.7.6.2 Signed shift right

5.7.6.3 Unsigned shift right

5.7.7 Operatoren buiten categorie

5.7.7.1 De dot-operator

5.7.7.2 De instanceof-operator

5.7.7.3 De new-operator

5.7.7.4 De equals-methode

5.7.7.5 De ternaire operator

5.7.8 Prioriteitsregels voor operatoren

5.8 Oefeningen

6 Controlestructuren

6.1 Sequenties

6.1.1 Uitdrukkingen en opdrachten

6.1.2 Codeblokken

6.2 Keuzestructuren

6.2.1 Beslissingen met 2 alternatieven: if

6.2.2 Beslissingen met 2 alternatieven: if... else

6.2.3 Ternaire operator

6.2.4 Beslissingen met meerdere alternatieven: switch (en break)

6.3 Lussen en iteraties

6.3.1 De gewone for-lus

6.3.2 Enhanced for-lus (for each)

6.3.3 De while lus

6.3.4 De do-while lus

6.4 Aftakkingen

6.4.1 Labels

6.4.2 break

6.4.3 continue

6.4.4 return



6.5 Oefeningen

6.6 Extra oefeningen

7 Reeksen

7.1 Wat is een reeks?

7.2 Declareren, aanmaken en opvullen van een reeks

7.2.1 Declareren van een reeks

7.2.2 Aankmaken van een reeks

7.2.3 Opvullen van een reeks

7.3 Multidimensionale reeks

7.3.1 Wat is een multidimensionale reeks

7.4 Manipuleren van een reeks

7.4.1 Length-eigenschap

7.4.2 Itereren over een reeks

7.5 Oefeningen

7.6 Extra oefeningen

8 Methoden

8.1 Wat is een methode

8.2 Declaratie

8.3 Keywords en naamgeving

8.3.1 Keywords

8.3.2 Naamgeving en conventies

8.3.3 Modifiers (keywords)

8.3.4 Access modifiers

8.4 Gebruik van methoden

8.4.1 Een methode gebruiken

8.4.2 Lokale- en instantievariabelen

8.4.3 Volgorde van uitvoeren

8.5 Soorten methoden

8.5.1 Constructors

8.5.1.1 Wat is een constructor

8.5.1.2 Aanroepen super()

8.5.1.3 Aanroepen this()



- 8.5.2 Statische methoden
- 8.5.3 Getters en setters
 - 8.5.3.1 Opvragen (getter)
 - 8.5.3.2 Instellen (setter)
- 8.6 Vervangen en herdefiniëren
 - 8.6.1 Vervangen (override)
 - 8.6.2 Herdefiniëren (overload)
- 8.7 Methoden koppelen
- 8.8 Oefeningen
- 8.9 Extra oefeningen

9 Documentatie

- 9.1 Waarom is documentatie belangrijk?
- 9.2 Soorten commentaar in Java
 - 9.2.1 Een regel
 - 9.2.2 Meerdere regels
 - 9.2.3 Speciale Java-commentaar
- 9.3 Javadoc
 - 9.3.1 De Javadoc tool
 - 9.3.2 Documentatie van een klasse
 - 9.3.3 Documentatie van variabelen
 - 9.3.4 Documentatie van methoden
 - 9.3.5 Documentatie van een package
 - 9.3.5.1 package-info.java
 - 9.3.5.2 package.html
 - 9.3.6 Javadoc genereren
- 9.4 Oefeningen
- 9.5 Extra oefeningen

10 De JDK verkennen

- 10.1 De klasse Object
 - 10.1.1 Inleiding
 - 10.1.2 De methode toString()
 - 10.1.3 De methode equals()



- 10.1.4 De methode hashCode()
- 10.1.5 De methode clone()
- 10.2 De Klasse String
 - 10.2.1 Constructie van een String
 - 10.2.2 De String Constant Pool
 - 10.2.3 Concatenatie van Strings
 - 10.2.4 De onveranderlijke String
 - 10.2.5 Oefeningen met Strings
- 10.3 StringBuilder en StringBuffer
 - 10.3.1 Vergelijking StringBuilder met StringBuffer
 - 10.3.2 Vergelijking met String
 - 10.3.3 Enkele veelgebruikte methoden
- 10.4 De klasse Number
 - 10.4.1 Introductie
 - 10.4.2 Boxing en unboxing
 - 10.4.3 Oefeningen
- 10.5 De klasse Scanner
- 10.6 De klasse Math
- 10.7 De klasse Arrays
- 10.8 Oefeningen

11 Object-Oriëntatie: Intermediair

- 11.1 Overerving
- 11.2 Interfaces
- 11.3 Polymorfisme
- 11.4 IS-EEN versus HEEFT-EEN

12 Overerving

- 12.1 Realisatie in Java
 - 12.1.1 Final
 - 12.1.2 Deadly Diamond of Death
- 12.2 Overerving van methoden
 - 12.2.1 Override
 - 12.2.2 Het keyword super



12.3 Overerving van eigenschappen

12.3.1 Field hiding

12.3.1.1 Polymorfisme van instantievariabelen

12.4 Oefeningen

13 Abstracte klassen en methoden

13.1 Abstracte klasse

13.2 Abstracte methode

13.3 Oefeningen

14 Interfaces

14.1 Interfaces

14.1.1 Wat is een interface?

14.1.2 Hoe gebruiken we een interface?

14.2 Interface of abstracte klasse?

14.3 Veel gebruikte interfaces

14.4 Oefeningen

15 Enumerations

15.1 Inleiding

15.2 Declaratie

15.3 Vergelijking en verschillen met een Java-klasse

15.4 Eigenschappen meegeven

15.5 Methoden overschrijven

15.6 Oefeningen

16 Inwendige klassen

16.1 Inleiding

16.2 Gewone inner classes

16.3 Static nested classes

16.4 Method-local inner classes

16.5 Anonymous inner classes

16.6 Oefeningen



17 Robuustheid

17.1 Annotations

17.1.1 Inleiding

17.1.2 Syntax

17.1.3 Aanbevolen gebruik

17.1.4 In- en uitschakelen

17.1.5 Zelf annotaties schrijven

17.1.5.1 Declaratie en methoden

17.1.5.2 Standaardwaarden

17.1.5.3 De @Target-annotatie

17.1.5.4 Retention Policy

17.1.5.5 Annotaties en javadoc

17.1.5.6 De CopyrightReader

17.2 Exception handling

17.2.1 Inleiding

17.2.2 Soorten fouten

17.2.3 Exception handling syntax

17.2.3.1 Throws declaratie

17.2.3.2 Try, catch en finally

17.2.4 Zelf exceptions gooien

17.3 Assertions

17.3.1 Wanneer gebruik ik een assertion?

17.3.2 Compiler switches

17.4 Oefeningen

