

Introductie tot de cursus

- 1 Plaats en functie van de cursus 7
- 2 Inhoud van de cursus 8
 - 2.1 Voorkennis 8
 - 2.2 Leerdoelen 8
 - 2.3 Opbouw van de cursus 9
 - 2.4 Leermiddelen 10
- 3 Aanwijzingen voor het bestuderen van de cursus 10
- 4 Software 12
- 5 Tentaminering 13
- 6 Overige informatie 13

Introductie tot de cursus

Om u wegwijs te maken in de cursus *Objectgeoriënteerd programmeren in Java 2*, informeren wij u eerst over de bedoeling van de cursus, de opzet van het cursusmateriaal en de manier waarop u de cursus kunt bestuderen. U vindt in deze introductie praktische en studietechnische informatie die u inzicht geeft in de aard en opzet van de cursus en u helpt bij het studeren.

1 Plaats en functie van de cursus

De cursus Objectgeoriënteerd programmeren in Java 2 is een cursus van het tweede niveau met een studielast van 100 uur (4.3 EC). Het is de tweede cursus in de informaticaopleidingen van de Open Universiteit Nederland die geheel aan programmeren is gewijd en is een direct vervolg op de basiscursus *Objectgeoriënteerd programmeren in Java 1* (T.25.1.5.1).

De twee cursussen samen bieden een brede inleiding in objectgeoriënteerd programmeren in het algemeen en in het gebruik van de taal Java in het bijzonder.

Aan het eind van de basiscursus kon u, gebruikmakend van een ontwikkelomgeving, een eenvoudige applicatie construeren die is opgebouwd uit een domeinlaag van enkele klassen en een interface-laag van één klasse, waarbij de taak van de interfacelaag beperkt blijft tot de communicatie met de gebruiker. Voor de constructie van de gebruikersinterface leerde u een visuele editor te gebruiken. U leerde ook waar nodig gebruik te maken van klassen uit de Java API, met de nadruk op veel gebruikte klassen uit `java.util` en `java.lang`. De cursus kende daarmee drie belangrijke lijnen, namelijk objectgeoriënteerd ontwerpen, de taal Java en het gebruik van de Java API.

In deze tweede cursus zetten we de drie genoemde lijnen voort. Het maken van een goed ontwerp is iets wat door de hele cursus in alle voorbeelden aandacht krijgt. In de eerste twee blokken komen alle taalelementen aan bod die in de basiscursus nog niet zijn behandeld. In het derde en vierde blok besteden we aandacht aan twee belangrijke onderdelen van de API. Het onderwerp van het derde blok is gegevensopslag, waarbij aandacht wordt besteed aan de packages `java.io` (voor het werken met tekstbestanden) en `java.sql` (voor de koppeling van een Java-programma aan een relationele database). Het onderwerp van het vierde blok is de constructie van gebruikersinterfaces, met aandacht voor de packages `javax.swing`, `java.awt` en `java.awt.geom`.

Om voeling te houden met de actualiteit worden bovendien op de cursus relevante artikelen beschikbaar gesteld over onderwerpen die geen plaats in het cursusmateriaal hebben gekregen.

De cursus is bij de Open Universiteit een verplicht onderdeel van de propedeuse en daarmee van de bacheloropleiding Informatica. De cursus maakt eveneens deel uit van het Certified Professional Program Javaprogrammeur. Samen met de basiscursus vormt de cursus ook een goede inleiding programmeren in Java voor iedereen die daar behoefte aan heeft.

2 Inhoud van de cursus

2.1 VOORKENNIS

De cursus is een direct vervolg op Objectgeoriënteerd programmeren in Java 1 en kan dus pas bestudeerd worden als deze cursus is afgerond of als op andere wijze voldoende basiskennis van Java is verworven.

Concreet betekent dit dat u

- de basisbegrippen van objectgeoriënteerd programmeren (klasse, instantie, attribuut, methode, de eerste beginselen van overerving) moet begrijpen en moet kunnen toepassen
- een eenvoudige klasse moet kunnen definiëren
- zonder al te veel moeite methoden moet kunnen schrijven, gebruikmakend van de volgende elementaire taalconstructies van Java: toekenningen, expressies, keuzeopdrachten, herhalingsopdrachten, arrays en ArrayLists
- inzicht moet hebben in de manier waarop objecten in het geheugen zijn gerepresenteerd en in de consequenties daarvan (aliasing), ook bij parameteroverdracht
- overweg moet kunnen met de bij deze cursus gebruikte ontwikkelomgeving.

2.2 LEERDOELEN

Na het volgen van deze cursus wordt verwacht dat u

- de syntaxis en semantiek van Java kent
- met name begrip heeft van overerving (inclusief het gebruik van abstracte klassen en interfaces), van exception handling en van threads
- eenvoudige generieke klassen kunt definiëren
- begrip heeft van de wijze waarop in objectgeoriënteerde programma's gebruik gemaakt kan worden van programmeren per contract
- de verschillende manieren kent waarop persistentie van gegevens gerealiseerd kan worden en twee daarvan ook kunt toepassen, namelijk streams (package java.io) en de koppeling met databases (package java.sql).
- in staat is om met behulp van de package javax.swing zelf een grafische gebruikersinterface te programmeren.

Na het volgen van de cursus bent u in staat om, vanuit een gegeven specificatie, zelfstandig een overeenkomstig objectgeoriënteerd programma te ontwerpen en te implementeren, gebruikmakend van alle elementen van de taal Java. Dat ontwerp kan verscheidene (tot circa tien) eigen

klassen bevatten en maakt een zinvol gebruik van behandelde delen van de Java API (het gebruik van streams, de koppeling met een database met behulp van JDBC, het programmeren van een grafische gebruikersinterface gebaseerd op Swing). Het programma is goed ontworpen (opbouw in lagen, klassen met duidelijke, beperkte verantwoordelijkheden, gebruik van packages), de code is helder en begrijpelijk, het programma is systematisch getest en het is gedocumenteerd met behulp van Javadoc.

2.3 OPBOUW VAN DE CURSUS

Blok 1 Typen en hiërarchieën

5 leereenheden
studielast: 30 uur

We beginnen de cursus met een *inleidende leereenheid*, waarin de vraag wordt gesteld – en deels beantwoord – wat een goed programma is. In deze leereenheid behandelen we ook delen van UML, een notatie speciaal gericht op het ontwerpen van objectgeoriënteerde programma's. De rest van het eerste blok is gewijd aan het typesysteem van Java. Ten eerste wordt *overerving* behandeld. Dynamische binding van methoden is daarbij een sleutelbegrip. Vervolgens komen abstracte klassen en interfaces aan de orde, begrippen die een belangrijke rol spelen bij het definiëren van zogeheten *ontwerppatronen*: schematische oplossingen voor veel voorkomende problemen. Het blok eindigt met een korte behandeling van generics: de uitbreiding van Java die het gebruik van typeparameters mogelijk maakt. Dit is een lastig en vrij specialistisch onderwerp; een deel behoort daarom niet tot de tentamenstof.

Blok 2 Exceptions en threads

3 leereenheden
studielast: 22 uur

Het tweede blok voltooit de behandeling van de taalconcepten van Java. De eerste twee leereenheden gaan over fouten die kunnen optreden in programma's en hoe daarmee om te gaan. Leereenheid 6 kijkt welke soorten fouten er zijn, wat er gedaan moet worden om ze op te merken (testen) en hoe hun oorzaak opgespoord kan worden (debuggen). Java biedt een mechanisme om bepaalde soorten fouten af te handelen (exception handling). Dit is het onderwerp van leereenheid 7. Leereenheid 8 tenslotte biedt een korte inleiding in het programmeren met Threads, dat een vorm van parallelisme mogelijk maakt. We geven aan hoe een programma met meerdere threads gemaakt kan worden, maar laten ook zien dat u zich daarmee op glad ijs begeeft.

Blok 3 Gegevensopslag

2 leereenheden
studielast: 15 uur

In de blokken 3 en 4 ligt de nadruk op delen van de API. Het onderwerp van blok 3 is gegevensopslag. In twee opeenvolgende leereenheden worden eerst het werken met streams behandeld (package java.io) en vervolgens de koppeling van een Java-programma met een relationele database via JDBC (Java database connectivity; package java.sql).

Blok 4 Gebruikers- interfaces

4 leereenheden
studielast: 28 uur

Blok 4 behandelt de constructie van gebruikersinterfaces zonder gebruik te maken van een visual editor. De eerste twee leereenheden worden besteed aan het werken met Swing-componenten; ook het event handling mechanisme komt daarbij aan de orde. De derde leereenheid van het blok gaat over het zelf tekenen van componenten. De laatste leereenheid beschrijft het Observerpatroon, een veel gebruikt en belangrijk ontwerp-patroon dat het mogelijk maakt om de domeinlaag geheel onafhankelijk te houden van de gebruikersinterfaces, ook wanneer het initiatief voor wijzigingen in die interfaces bij de domeinlaag ligt.

Artikelen

Op de cursussite op studienet verzamelen we artikelen die een verdieping geven aan de in de cursus behandelde stof en/of ingaan op nieuwe ontwikkelingen. Enkele daarvan kunnen geselecteerd worden als tentamenstof. Zie ook paragraaf 5.

2.4 LEERMIDDELEN

Het cursusmateriaal bestaat uit de volgende onderdelen:

- cursusdeel 1, met blok 1
- cursusdeel 2, met blokken 2 en 3
- cursusdeel 3, met blok 4
- een cd-rom met de gebruikte ontwikkelomgeving.

Cursussite

Daarnaast behoort ook de *cursussite* tot het cursusmateriaal. U vindt deze via <http://studienet.ou.nl>. Als dit de eerste OUNL-cursus is die u bestudeert, dan moet u zich vooraf registreren (dat kan via de genoemde pagina). Nadat u zich geregistreerd hebt en bent ingelogd, komt u op uw werkplek. Als u voor deze cursus bent ingeschreven, bevat uw werkplek een link naar de bijbehorende cursussite.

Informatie over begeleiding en tentaminering

De cursussite biedt de meest actuele informatie over de cursus. Op de cursussite vindt u onder meer informatie over begeleiding en tentaminering, handleidingen bij de gebruikte software (zie ook paragraaf 4), bouwstenen voor te ontwikkelen Java-programma's, een eindtoets en errata bij de cursus. Ook kunt u via deze site in contact komen met medestudenten en docenten.

3 **Aanwijzingen voor het bestuderen van de cursus***Leereenheid*

Een blok is verdeeld in *leereenheden*. Een leereenheid is een afgerond deel van de stof dat u in een of twee dagdelen kunt bestuderen. Elke leereenheid wordt voorafgegaan door een inhoudsopgave en bestaat verder uit een introductie, een leerkern, een zelftoets en een terugkoppeling. Deze onderdelen van een leereenheid omschrijven we kort.

*Introductie**Leerdoelen*

De *introductie* van een leereenheid geeft kort aan wat de inhoud van de leereenheid is en schept een kader waarin deze inhoud geplaatst moet worden. In de introductie vindt u de *leerdoelen*. Die stellen u in staat na te gaan welke kennis, inzichten en vaardigheden u zich eigen moet maken door bestudering van de leereenheid. De introductie wordt afgesloten met studeeraanwijzingen voor de betreffende leereenheid. Daarin wordt in elk geval de studielast vermeld.

*Leerkern
Opgave
Opdracht*

De *leerkern* van een leereenheid bevat de feitelijke leerstof. Op bepaalde plaatsen in de leerkern treft u *opgaven* of *opdrachten* aan. Het verschil tussen een opgave en een opdracht is dat u een opdracht achter uw pc moet uitvoeren, terwijl een opgave gewoon met pen en papier kan.

OPGAVE 0.0

Dit is een voorbeeld van een opgave.

- a Werk elke opgave *tijdens* het bestuderen van de leereenheid uit.
- b Vergelijk uw antwoord direct met het antwoord in de terugkoppeling aan het eind van de leereenheid.

Studeeropdracht

U treft in de leerkern ook ongenummerde vragen aan: de *studeer-opdrachten*.

Dit is een voorbeeld van een ongenummerde vraag ofwel studeeropdracht. Wat denkt u dat de functie daarvan is?

Samenvatting

Een studeeropdracht wordt altijd in de direct daarop volgende tekst beantwoord. De functie van studeeropdrachten is dat u zelf uw eigen antwoord op de vraag probeert te formuleren, voordat u de betreffende tekst leest. Dat helpt u om aandachtig de denkstappen in de tekst te volgen.

Zelftoets

Meteen na de leerkern staat een *samenvatting* waarin de belangrijkste zaken uit deze leerkern nog eens worden opgesomd.

Terugkoppeling

De *zelftoets* van een leereenheid bestaat uit opgaven die bedoeld zijn om u te helpen beoordelen of u, na bestudering van de leerkern, de leerdoelen inderdaad bereikt hebt. Het is van belang dat u na bestudering van de leerstof de gehele zelftoets nauwgezet uitwerkt.

Belangrijk

In de *terugkoppeling* van de leereenheid zijn de uitwerkingen opgenomen van de opgaven en opdrachten en van de zelftoets. Dit onderdeel van de leereenheid heet niet voor niets terugkoppeling: u wordt in staat gesteld uw antwoord op een opdracht te vergelijken met het gewenste antwoord.

Wij raden u ten zeerste af om te snel het antwoord in de terugkoppeling op te zoeken. U leert meer en beter als u eerst zelf een oplossing probeert te vinden en de opgave of opdracht volledig en aandachtig uitwerkt. Meer nog dan bij andere cursussen is dat bij een programmeercursus als deze van belang. *Programmeren leert u alleen maar door het zelf te doen!* Al mag u op het tentamen het cursusmateriaal gebruiken (zie paragraaf 5), dat is geen reden de opgaven en opdrachten niet te maken.

Margeteksten

Het zal u opvallen dat iedere pagina aan de linkerkant twee kolommen heeft: de *marge*. In de meest linkse kolom treft u kernbegrippen aan.

Kernbegrippen

Kernbegrippen zijn, zoals de naam al zegt, centrale begrippen. Deze kolom maakt het gemakkelijk om bepaalde tekstonderdelen snel terug te vinden en schetst tegelijkertijd de hoofdlijn van de leerinhoud.

Studeeraanwijzingen

Dit is een voorbeeld van een studeeraanwijzing.

In de tweede kolom van de marge staan korte *studeeraanwijzingen*. Dat kunnen toelichtingen op vreemde woorden zijn, aanwijzingen hoe de tekst gelezen moet worden, herhalingen of verwijzingen naar eerdere leereenheden.

Leestekst

Soms is in de leerstof een passage ingevoegd die nader op de stof ingaat of een kanttekening plaatst. Een dergelijke passage behoort niet tot de tentamenstof en is te herkennen aan het kleinere lettertype.

Een leestekst wordt afgedrukt in kleine letters. Zo kunt u meteen zien dat het hier een aanvulling betreft die u alleen maar door hoeft te lezen en die niet tot de tentamenstof behoort.

4 Software

In de voorgaande versies van deze cursus zijn drie verschillende ontwikkelomgevingen gebruikt: eerst Visual Café van Symantec en toen twee versies van JBuilder van Borland. De opbouw van de cursus veranderde in die tijd veel minder; dit is pas de eerste grote revisie.

We verwachten dat dit in de toekomst niet anders zal worden: ontwikkelomgevingen verouderen veel sneller dan de taal. We hebben daarom besloten om het cursusmateriaal onafhankelijk te maken van de gebruikte software.

Benodigde software	<p>Om de opdrachten in deze cursus te kunnen doen, moet u eerst de benodigde software installeren. Deze software bestaat uit de volgende onderdelen:</p> <ul style="list-style-type: none"> – de taal Java – een ontwikkelomgeving voor Java – bouwstenen bij de cursus – een database server (voor leereenheid 10) – een JDBC-driver (voor leereenheid 10).
Installatie- en gebruikershandleidingen op studienet	<p>De eerste twee onderdelen vindt u op de cd-rom die als onderdeel van het cursusmateriaal is meegeleverd. De andere drie onderdelen zijn beschikbaar via de cursussite op studienet. Daar vindt u ook de benodigde installatie- en gebruikershandleidingen.</p> <p>U moet in elk geval de eerste drie onderdelen installeren voor u begint met het bestuderen van de cursus. De laatste twee onderdelen zijn alleen nodig voor leereenheid 10; u kunt die naar keuze nu installeren of wanneer u aan die leereenheid toe bent.</p>
Apparatuur	<p>In de installatiehandleiding van de ontwikkelomgeving zijn ook apparatuurspecificaties opgenomen. In het algemeen gaat de Open Universiteit uit van het gebruik van computers met Windows en zal een computer die niet ouder is dan ongeveer vijf jaar voldoende zijn om de software te kunnen draaien. We streven naar het gebruik van een ontwikkelomgeving die ook voor andere besturingssystemen beschikbaar is (bijvoorbeeld Linux en MacOS), zelfs al geven we daarbij geen ondersteuning. Raadpleeg voor details de installatiehandleiding; als u daar nog geen toegang toe heeft kunt u contact opnemen met de examinator van de cursus.</p> <p>U bent uiteraard vrij om een andere ontwikkelomgeving te gebruiken dan degene die we meeleveren. De Java-broncode uit de bouwstenen blijft bruikbaar, maar u moet deze dan zelf in een project invoegen. U kunt bij het gebruik van een andere ontwikkelomgeving geen aanspraak maken op ondersteuning bij het gebruik daarvan.</p>

5 Tentaminering

Schriftelijk tentamen

Voor studenten uit de wo-opleiding Informatica wordt de cursus afgesloten met een *schriftelijk tentamen* van drie uur. Het tentamen bestaat geheel uit open vragen. Het is toegestaan om tijdens het tentamen gebruik te maken van het cursusmateriaal. De tentamendata vindt u op de cursussite. Algemene informatie over de gang van zaken bij het tentamen kunt u vinden via de website van de Open Universiteit Nederland: www.ou.nl.

Verplichte tentamenstof

Om aandacht te kunnen geven aan nieuwe, belangrijke ontwikkelingen en enige ruimte te creëren om aan wensen van studenten tegemoet te komen, houden we de mogelijkheid open om één leereenheid uit dit materiaal te vervangen door een of meer artikelen. Dat betekent dat de feitelijke tentamenstof licht kan variëren.

Uiterlijk zes weken voor elk tentamen wordt op de cursussite, als onderdeel van de informatie over de tentaminering, bekend gemaakt of er artikelen tot de tentamenstof behoren en zo ja, welke dat dan zijn en welke leereenheid niet bestudeerd hoeft te worden. Wij raden u met klem aan om tijdig deze informatie te raadplegen.

Eindtoets

Bij de cursus hoort een eindtoets die representatief is voor het tentamen. Wij adviseren u nadrukkelijk deze pas te maken als u klaar bent met de tentamenvoorbereiding. U vindt de eindtoets op de cursussite.

Voor cursisten uit het CPP Javaprogrammeur bestaat de toetsing uit het maken van een viertal programmeeropdrachten, aangevuld met een schriftelijke toets van anderhalf uur. Voorafgaand aan de echte toets wordt u in de gelegenheid gesteld om een voorbeeldtoets te maken. Precieze informatie wordt door de begeleider verstrekt.

6 Overige informatie

Java-web

www.ou.nl/java

Algemene informatie over deze en andere Java-cursussen van de Open Universiteit en over het CPP Javaprogrammeur is te vinden op het *Java-web*: www.ou.nl/java. Deze website is vooral bedoeld voor voorlichting en bevat informatie over de inhoud en opbouw van de verschillende cursussen.