

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 8
 - 2.4 Leermiddelen 10
- 3 Aanwijzingen voor het bestuderen van de cursus 10
 - 3.1 Opbouw van een leereenheid 10
 - 3.2 Java en Eclipse 12
- 4 Installeren van de software 12
 - 4.1 Apparatuurspecificaties 12
 - 4.2 Installeren van Java 13
 - 4.3 Instellen omgevingsvariabelen 14
 - 4.4 Kopiëren van jar-bestanden 17
 - 4.5 Installeren van Eclipse 18
 - 4.5.1 Licentie bekijken en accepteren 18
 - 4.5.2 Eclipse installeren 18
 - 4.6 Installeren van de bouwstenen 20
- 5 Cursusteam 20
- 6 Tentaminering 21
- 7 Overige informatie 21

Introductie tot de cursus

Om u wegwijs te maken in de cursus *Objectgeoriënteerd programmeren in Java 1*, 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.

Ook vindt u aanwijzingen voor het *installeren van de software*.

1 Plaats en functie van de cursus

De cursus Objectgeoriënteerd programmeren in Java 1 is een cursus van het eerste niveau met een studielast van 100 uur (4.3 EC). Het is de eerste cursus in de informaticaopleiding van de Open Universiteit Nederland die geheel aan programmeren is gewijd.

Java

Objectgeoriënteerd

Als programmeertaal wordt *Java* gebruikt, een taal die sinds de introductie midden jaren '90 onverminderd populair is. Deze taal is *objectgeoriënteerd* (vaak afgekort tot OO), wat in eerste benadering betekent dat delen van de totale programmeertaak worden uitgevoerd door verschillende objecten. Er is in deze cursus voor gekozen om vanaf het begin daadwerkelijk gebruik te maken van objecten. De cursus vormt samen met de vervolgcursus *Objectgeoriënteerd programmeren in Java 2* een volledige inleiding in het programmeren in Java.

*Grafische
gebruikersinterface*

Visuele editor

Programma's worden ontwikkeld binnen een ontwikkelomgeving die veel steun biedt bij het opstellen van een programma dat voldoet aan de taaldefinitie, en dat helder en leesbaar is vormgegeven. Vanaf leereenheid 4 worden de programma's bovendien voorzien van een *grafische gebruikersinterface*. De gebruikte ontwikkelomgeving beschikt daartoe over een *visuele editor*, die het de programmeur toestaat om de interfacecomponenten met behulp van de muis op een ontwerpscherm te plaatsen. De benodigde Java-code wordt vervolgens automatisch gegenereerd.

Verder is binnen deze cursus gekozen voor een cyclische benadering. In het eerste blok leert u eenvoudige programma's te schrijven en deze te voorzien van een grafische gebruikersinterface. Nog niet alle details van de gebruikte programmeerconstructies worden in dat blok uitgediept. Vanaf het tweede blok volgt een volledige en systematische behandeling van de concepten die wij u in deze cursus aan willen leren.

Doelgroepen	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 Kort Hoger Onderwijsprogramma Gecertificeerd Java-programmeur. De cursus is ook geschikt als losse cursus voor iedereen die behoefte heeft aan een cursus programmeren in Java.
	<p>2 Inhoud van de cursus</p>
	<p>2.1 VOORKENNIS</p>
<i>Windows</i>	De software die in de cursus wordt gebruikt draait onder <i>Windows</i> ; het is daarom nodig dat u beschikt over een behoorlijke vaardigheid met dit besturingssysteem. We gaan er verder van uit dat u kunt omgaan met het programma <i>WinZip</i> .
<i>WinZip</i>	De cursus veronderstelt geen voorkennis op het gebied van programmeren of objectoriëntatie, maar kennis van deze onderwerpen op het niveau aangeboden in de cursus Inleiding informatica vergemakkelijkt de bestudering wel. Dit geldt ook voor ervaring met andere niet-objectgeoriënteerde programmeertalen zoals JavaScript, C, Pascal of Cobol.
	<p>2.2 LEERDOELEN</p>
	<p>Na het volgen van deze cursus bent u in staat om, vanuit een gegeven specificatie, zelfstandig een objectgeoriënteerd programma te ontwerpen en te implementeren, en dit programma te voorzien van een eenvoudige grafische gebruikersinterface. Hierbij maakt u gebruik van het deel van de taal Java dat is aangeleerd in de cursus. Het programma is goed ontworpen, wat inhoudt dat het bestaat uit klassen (in een objectgeoriënteerde taal zijn klassen de programma-eenheden waar de objecten van worden afgeleid) met een duidelijke, beperkte verantwoordelijkheid. De programmacode is helder en begrijpelijk en het programma is gedocumenteerd met behulp van de tool Javadoc. U kunt bovendien de belangrijkste functies gebruiken van de ontwikkelomgeving Eclipse.</p> <p>Hierbij moet wel worden gezegd dat noch de taal Java, noch alle mogelijkheden van Eclipse uitputtend worden behandeld. In een tijdsbestek van 100 uur is dat niet mogelijk. Om een volledig overzicht te krijgen van de taal Java, dient u ook de vervolgcursus Objectgeoriënteerd programmeren in Java 2 te bestuderen.</p>
	<p>2.3 OPBOUW VAN DE CURSUS</p>
Blok 1 Inleiding	Blok 1 is een <i>inleiding</i> van vier leereenheden, die afwisselend de nadruk leggen op de taal Java en op de ontwikkelomgeving Eclipse. In dit blok maakt u kennis met objectoriëntatie door gebruik te maken van bestaande klassen. In leereenheid 1 wordt een eenvoudig Java-programma ontwikkeld en met minimale hulpmiddelen uitgevoerd, dus nog zonder ontwikkelomgeving. Leereenheid 2 gaat over het schrijven en verwerken van programma's met behulp van de ontwikkelomgeving Eclipse. In leereenheid 3 leert u programma's te schrijven die gebruikmaken van eenvoudige vormen van keuze en herhaling. Leereenheid 4 is gewijd aan de visuele editor. In het hele blok wordt als doorlopend voorbeeld het stemmen met behulp van een stemmachine gebruikt.

- Blok 2
Objecten en klassen
- In blok 2 leert u alles wat u voor deze cursus moet weten over *objecten en klassen*. Leereenheid 5 behandelt objectoriëntatie op conceptueel niveau, dus (grotendeels) los van de taal Java. In deze leereenheid worden onder meer de concepten klasse en object uitgediept en worden de relaties tussen klassen verduidelijkt. Ook het in OO belangrijke begrip *overerving* komt hier voor het eerst aan de orde. Vervolgens maken we, in leereenheid 6, de overstap van het *gebruik* van klassen in blok 1, naar het *zelf definiëren* van klassen. In leereenheid 7 leert u werken met de uitgebreide Java-bibliotheek en de documentatie daarvan op het internet. In leereenheid 8 besteden we uitgebreid aandacht aan de manier waarop objecten in het geheugen zijn gerepresenteerd en de consequenties daarvan bij toekenning en parameteroverdracht.
- Blok 3
Algoritmiek
- Blok 3 behandelt de implementatie van *methoden* (dit zijn in zekere zin de belangrijkste programma-eenheden, omdat de programmeur hier het gewenste gedrag van de objecten codeert). Door de gevolgde cyclische benadering hadden we daar in blok 1 al een voorschot op genomen; in dit blok worden alle elementaire opdrachten gedetailleerd behandeld. In leereenheid 9-11 worden achtereenvolgens primitieve typen en expressies, keuzeopdrachten en herhalingsopdrachten behandeld. Leereenheid 12 gaat over het werken met Strings (tekst) en ArrayLists en arrays (lijsten van waarden). Leereenheid 13 gaat over documenteren en testen.
- Leereenheid 14 bevat een wat grotere programmeeropdracht, waarmee u uw programmeervaardigheid kunt testen en ook verder ontwikkelen. Deze opdracht betreft het uitprogrammeren van het spel yahtzee. U schrijft dit programma bovendien in een iets andere vorm dan de andere programma's in deze cursus, namelijk als *applet*. Een applet is een programma dat opgenomen kan worden in een webpagina. Java dankte zijn enorme populariteit aanvankelijk aan deze applets, maar in de loop van de eerste tien jaar van het bestaan van Java is het belang van deze toepassing sterk afgenomen.
- Op de cursussite vindt u ten slotte de eindtoets, die u kunt maken als voorbereiding op het tentamen.
- Bijlagen
- Het laatste cursusdeel bevat een aantal bijlagen. Deze bijlagen maken geen onderdeel uit van de leerstof, maar zijn vooral bedoeld als naslagwerk. De eerste bijlage bevat aanwijzingen voor het gebruik van Eclipse (zie ook paragraaf 3.1). De tweede bijlage bevat een overzicht van de syntaxis van Java. De derde bijlage geeft een overzicht van de betekenis en het gebruik van klassen- en toestandsdiagrammen die door de gehele cursus gebruikt worden.
- Belangrijk
- Sommige Java-concepten worden in blok 1 slechts oppervlakkig behandeld. Daarbij verdwijnen sommige details onder het tapijt. Later in de cursus volgt echter een veel grondiger behandeling, waarbij die details wel duidelijk worden gemaakt. Dit geldt met name voor toekenning, parameteroverdracht en voor het verschil tussen primitieve waarden en objecten. Het is mogelijk dat u door deze aanpak bij de bestudering van blok 1 met vragen blijven zitten. We raden u met klem aan niet te lang bij dit soort vragen stil te blijven staan; de meeste vragen worden in blok 2 beantwoord.

2.4 LEERMIDDELEN

Het cursusmateriaal bestaat uit de volgende onderdelen:

- Cursusdeel 1 met blok 1
- Cursusdeel 2 met blok 2
- Cursusdeel 3 met blok 3
- Cursusdeel 4 met de bijlagen
- Een cd-rom met de taal Java en de Java-ontwikkelomgeving Eclipse.

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

Op de cursussite vindt u onder meer informatie over begeleiding en tentaminering, handleidingen bij de ontwikkelomgeving Eclipse, bouwstenen voor te ontwikkelen Java-programma's 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**

3.1 OPBOUW VAN EEN LEEREENHEID

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*. Zij 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 dient uit te voeren, 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 leerkernel ook ongenummerde vragen aan: de *studeeropdrachten*.

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 leerkernel staat een *samenvatting* waarin de belangrijkste zaken uit deze leerkernel nog eens worden opgesomd. De *zelftoets* van een leereenheid bestaat uit opgaven die bedoeld zijn om u te helpen beoordelen of u, na bestudering van de leerkernel, de leerdoelen inderdaad bereikt hebt. Het is van belang dat u na bestudering van de leerstof de gehele zelftoets nauwgezet uitwerkt.

Terugkoppeling

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, dat wil zeggen: nog voor de opgave of opdracht volledig en aandachtig is uitgewerkt, het antwoord in de terugkoppeling op te zoeken. U leert meer en beter als u eerst zelf een oplossing probeert te vinden. 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 6), dat is geen reden de opgaven en opdrachten niet te maken.

Margeteksten

Kernbegrippen

Register

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* 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. De kernbegrippen zijn veelal opgenomen in het *register* achterin elk cursusdeel.

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 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.

3.2 JAVA EN ECLIPSE

Sinds de eerste versie van deze cursus, die in 1997 verscheen, zijn vijf verschillende ontwikkelomgevingen gebruikt: twee versies van Visual Café van Symantec, twee versies van JBuilder van Borland en nu versie 3.3 van de open-sourceomgeving Eclipse. De opbouw van de cursus veranderde in die tien jaar 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 zoveel mogelijk onafhankelijk te maken van de gebruikte ontwikkelomgeving.

In deze cursus verwijst alleen dit eerste cursusdeel expliciet naar Eclipse. In leereenheden 2 en 4 wordt stap voor stap uitgelegd hoe u met Eclipse en met de Visual Editor kunt werken.

Alle aanwijzingen voor het gebruik van Eclipse in latere leereenheden zijn bij elkaar gebracht in bijlage 1 bij de cursus en in filmpjes die u kunt bekijken op de cursussite. In de tekst van de leereenheid wordt als volgt naar deze bijlage verwezen (dit voorbeeld komt uit leereenheid 13):

OPDRACHT 13.6

Raadpleeg voor het uitvoeren van deze opdracht bijlage 1 over de ontwikkelomgeving of de filmpjes op de cursussite.

a Open uit de bouwstenen het project Le13Schrikkeltest. Voeg zonodig de JUnit-library aan het project toe.

Hoe in Eclipse de beschreven actie (in dit geval het toevoegen van de JUnit-library) moet worden uitgevoerd, staat in woorden beschreven in bijlage 1. Hebt u aan deze beschrijving niet genoeg, dan wordt de actie ook nog eens getoond in een filmpje dat u kunt bekijken op de cursussite.

4 Installeren van de software

4.1 APPARATUURSPECIFICATIES

In vrijwel elke leereenheid zijn opdrachten opgenomen die u op de computer dient uit te voeren. Om deze opdrachten uit te voeren, moet u beschikken over een niet al te oude pc met Windows 2000, XP of Vista (waarschijnlijk voldoen andere Windowsversies ook, maar dat durven we niet te garanderen). U dient bovendien toegang te hebben tot het internet.

Precieze apparatuurspecificaties zijn moeilijk te geven. Eclipse is een soort bouwdoos, waarvan meer of minder blokjes geïnstalleerd kunnen worden. Als u meer blokjes zou toevoegen of intensief gebruikmaakt van blokken die zijn meegeleverd maar die in deze cursus niet gebruikt worden, dan gaan de systeemeisen ook omhoog.

Minimum-
configuratie

Een acceptabele minimumconfiguratie lijkt de volgende te zijn:
– processorsnelheid van 1Ghz
– 512 Mb RAM
– 1 Gb vrije ruimte op de harde schijf.

We hebben echter ook een melding gevonden dat Eclipse bleek te werken op een laptop met Windows '98, een 233Mhz processor en 92Mb geheugen, al ging dat wel erg langzaam.

U moet bovendien beschikken over het programma WinZip.

Andere
besturingssystemen

Java en Eclipse zijn ook beschikbaar voor Linux, Solaris en MacOS. Het is dus mogelijk om deze cursus te volgen op een niet-Windowscomputer.

U moet dan wel zelf de juiste versie van Java en van Eclipse ophalen van de sites van Sun respectievelijk Eclipse (de cursussite op studienet bevat links naar de sites waar u deze software kunt downloaden) en deze installeren. Voor Java is dit eenvoudig. De installatie van de juiste bouwblokken van Eclipse is iets complexer. U moet kiezen voor de JEE download en daar vervolgens zelf de Visual Editor (plug-in VE) aan toevoegen. Verder moet u voor alle stappen in deze paragraaf passende equivalenten vinden.

De Open Universiteit geeft in principe geen ondersteuning bij het gebruik van andere besturingssystemen. We verzoeken wel cursisten die hiermee ervaring hebben, deze met anderen te delen via de cursussite.

4.2 INSTALLEREN VAN JAVA

Als eerste stap dient u de taal Java te installeren, vanaf de cd-rom die behoort bij de cursus. Figuur 1 toont de structuur van de cd-rom.

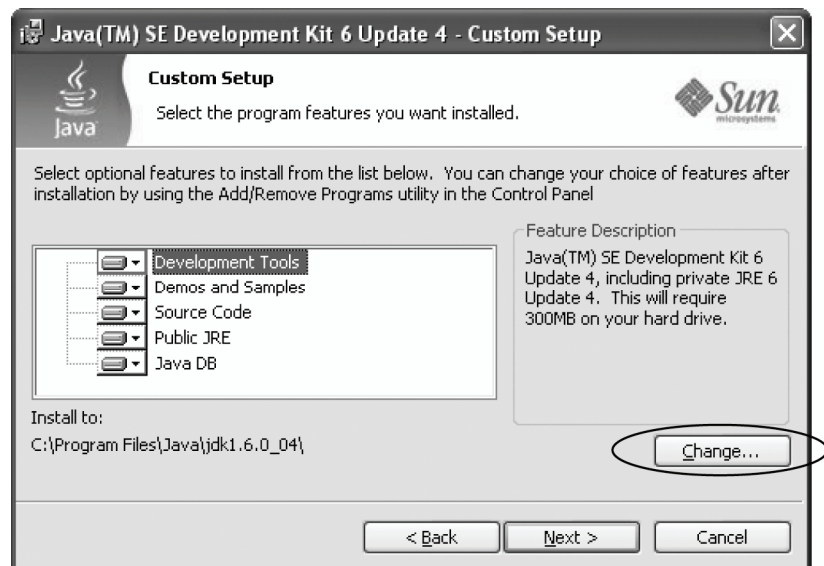


FIGUUR 1 Software op de meegeleverde cd-rom

Dubbelklik op het bestand *jdk-6u4-windows-i586-p.exe*; Java wordt nu geïnstalleerd.

U krijgt eerst het welkomstscherf van de InstallShield Wizard te zien en dan een licentieovereenkomst; hier dient u op Accept te klikken.

Het volgende scherm heeft de titel Custom Setup. Standaard wordt Java geïnstalleerd in de map C:\Program Files\Java\jdk1.6.0_04. Als u dat wilt kunt u hier, via de knop Change, een andere map kiezen (zie figuur 2). Klik vervolgens op Next.



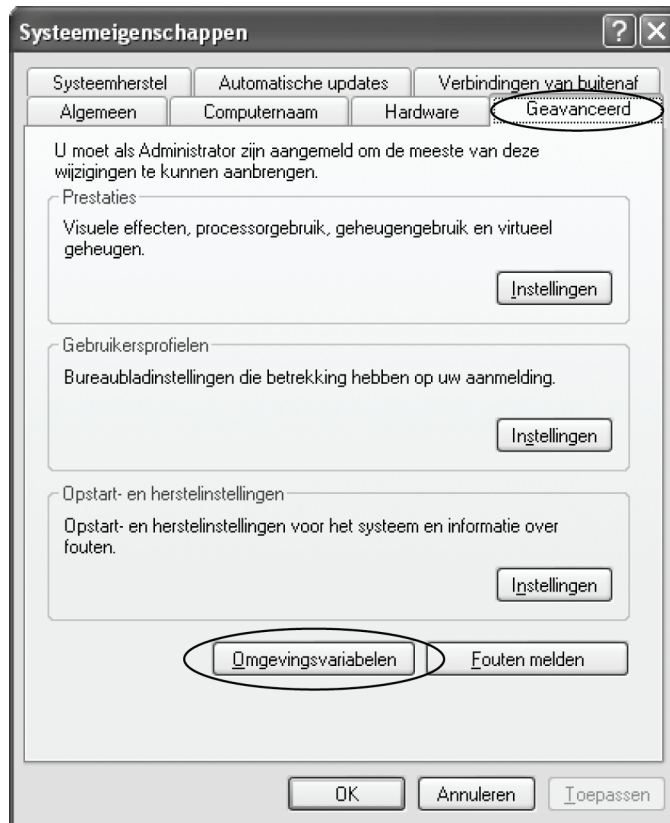
FIGUUR 2 Wijzig eventueel de installatiemap

De installatie begint. Na enige tijd krijgt u nogmaals een scherm met Custom Setup; klik hier meteen op Next. Na voltooiing van de installatie krijgt u het eindscherf van de InstallShield Wizard, dat aangeeft dat de installatie voltooid is. U hoeft de ReadMe-file niet per se te lezen, dus u kunt het betreffende hokje uitvinken. Klik tot slot op Finish.

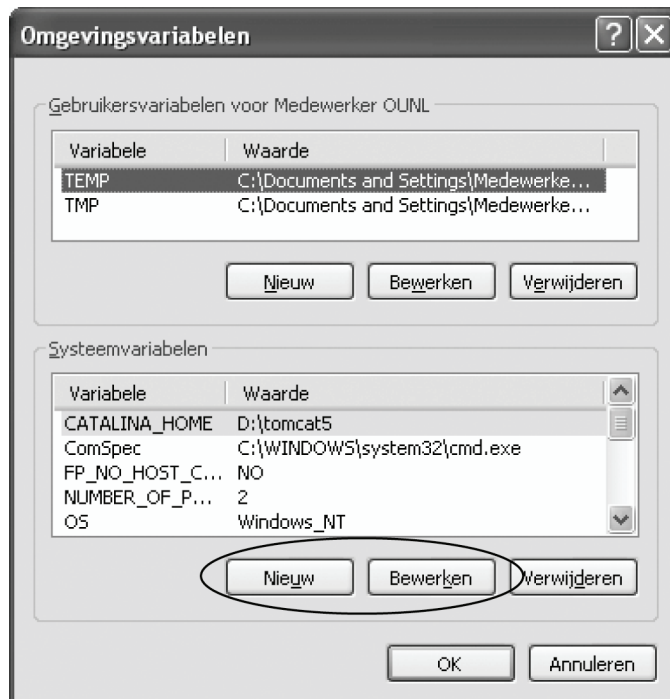
4.3 INSTELLEN OMGEVINGSVARIABLEN

Opdat Eclipse straks goed werkt, moet u nu twee omgevingsvariabelen instellen.

- Ga via de het Startmenu naar het Configuratiescherf.
- Kies daarin voor de optie System en vervolgens voor het tabblad Geavanceerd (zie figuur 3).
- Klik op de knop Omgevingsvariabelen; u krijgt het scherm van figuur 4.



FIGUUR 3 Het tabblad Geavanceerd van de systeemeigenschappen



FIGUUR 4 Het scherm Omgevingsvariabelen

De omgevings-
variabele
JAVA_HOME

- Klik op de onderste knop Nieuw. Vul als naam JAVA_HOME in, en als waarde de installatiemap van Java (indien u de standaard niet hebt gewijzigd is dit dus C:\Program Files\Java\jdk1.6.0_04; zie figuur 5).
- Mocht er al een variabele JAVA_HOME zijn, wijzig dan de waarde via de knop Bewerken in figuur 4; anders gebruikt Eclipse straks de verkeerde versie van Java.
- Klik op OK in het scherm van figuur 5.



FIGUUR 5 Het scherm Nieuwe systeemvariabele

De omgevings-
variabele Path

- Kies vervolgens de variabele Path en klik op de onderste knop bewerken. Voeg *vooraan* de waarde van deze variabele het volgende toe (laat de rest van de waarde vooral staan):

<waarde van JAVA_HOME>\bin;

Hierbij staat *<waarde van JAVA_HOME>* voor de installatiemap van Java (indien u de standaard niet hebt gewijzigd, voegt u dus C:\Program Files\Java\jdk1.6.0_04\bin; toe, als getoond in figuur 6. Let op de punt-komma!).



FIGUUR 6 Voeg *<waarde van JAVA_HOME>\bin;* toe aan variabele Path (wat er al stond, blijft staan)

Klik op OK in achtereenvolgens het scherm Systeemvariabele bewerken van figuur 6, het scherm Omgevingsvariabelen van figuur 4 en tot slot het scherm Systeemeigenschappen van figuur 3.

Let op!

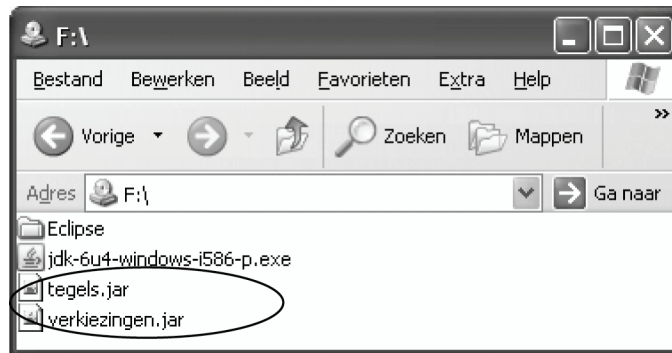
Wijzigingen in omgevingsvariabelen worden niet meteen zichtbaar in geopende DOS-boxen (Command-prompts), zoals u die in leereenheid 1 gebruikt. U moet die dus sluiten en na het aanbrengen van de wijzigingen een nieuwe openen.

4.4 KOPIËREN VAN JAR-BESTANDEN

Java Archives
jar

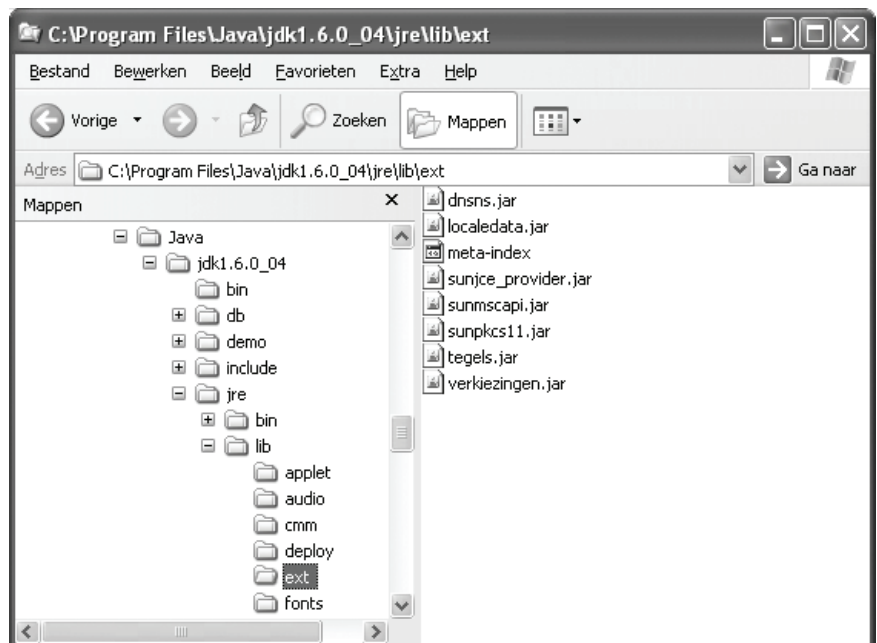
In het eerste blok van de cursus maken we gebruik van enkele klassen die we zelf hebben geschreven. Deze klassen zijn gebundeld in twee zogeheten *Java Archives* ofwel *jar*-bestanden (een bestandsformaat voor de uitlevering van Java-programma's). Opdat u hier zo eenvoudig mogelijk mee kunt werken, voegen we deze klassen toe aan de Java-bibliotheek.

Het gaat hier om de bestanden *verkiezingen.jar* en *tegels.jar* die op de cd-rom staan (zie figuur 7).



FIGUUR 7 De bestanden *tegels.jar* en *verkiezingen.jar* op de cd-rom

– Plaats deze bestanden in de map *<waarde van JAVA_HOME>\jre\lib\ext*, waarbij *<waarde van JAVA_HOME>* staat voor de installatiemap van Java (indien u de standaard niet hebt gewijzigd, gaat het dus om de map *C:\Program Files\Java\jdk1.6.0_04\jre\lib\ext*, als getoond in figuur 8).

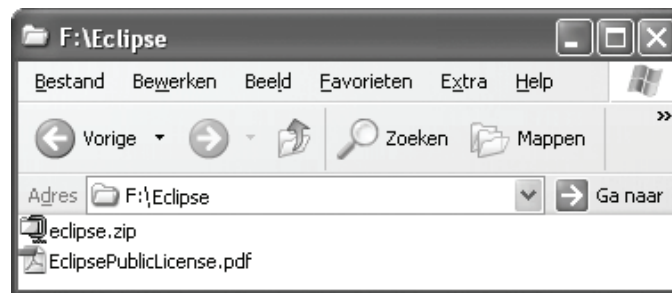


FIGUUR 8 Voeg *tegels.jar* en *verkiezingen.jar* toe aan *<waarde van JAVA_HOME>\jre\lib\ext*

4.5 INSTALLEREN VAN ECLIPSE

4.5.1 Licentie bekijken en accepteren

In de map Eclipse op de cd-rom vindt u twee bestanden (zie figuur 9). Het bestand met de naam EclipsePublicLicense.pdf bevat de licentievoorwaarden voor het gebruik van Eclipse. U dient nu allereerst dit bestand te bekijken. Door verder te gaan met de installatie en later Eclipse te gebruiken, geeft u aan dat u deze licentievoorwaarden accepteert.



FIGUUR 9 De inhoud van de map Eclipse op de cd-rom

4.5.2 Eclipse installeren

Het bestand met de naam eclipse.zip bevat alle bestanden die voor het gebruik van Eclipse (inclusief de door ons geselecteerde plug-ins) nodig zijn. Om Eclipse te kunnen gebruiken, is het voldoende om dit bestand uit te pakken.

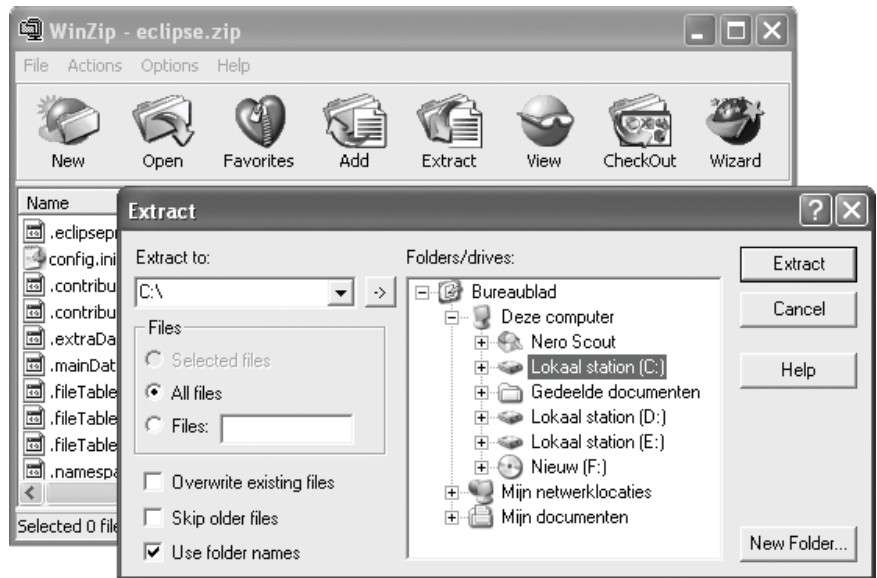
- Open het bestand eclipse.zip met behulp van WinZip en pak het uit naar een geschikte locatie, bijvoorbeeld C:\ of C:\Program Files. Alle bestanden worden binnen de door u gekozen locatie in een map eclipse geplaatst. Zorg er wel voor dat bij het uitpakken de optie Use folder names staat aangevinkt (zie figuur 10).

Om Eclipse makkelijk te kunnen starten, voegen we Eclipse toe aan het bureaublad.

- Open de zojuist aangemaakte map eclipse. Rechtsklik op het bestand eclipse.exe, kies Kopiëren naar en kies vervolgens Bureaublad (zie figuur 11). Op het bureaublad staat nu een snelkoppeling met de naam ‘Snelkoppeling naar eclipse.exe’. Wijzig deze naam desgewenst in Eclipse.
- Voeg indien u dit wilt Eclipse ook nog toe aan het Startmenu (bijvoorbeeld door het Eclipse pictogram vanaf het bureaublad te slepen naar de knop Start).

De installatie van Eclipse is hiermee voltooid.

Start Eclipse nog niet; installeer eerst de bouwstenen (zie paragraaf 4.6).



FIGUUR 10 Uitpakken van eclipse.zip



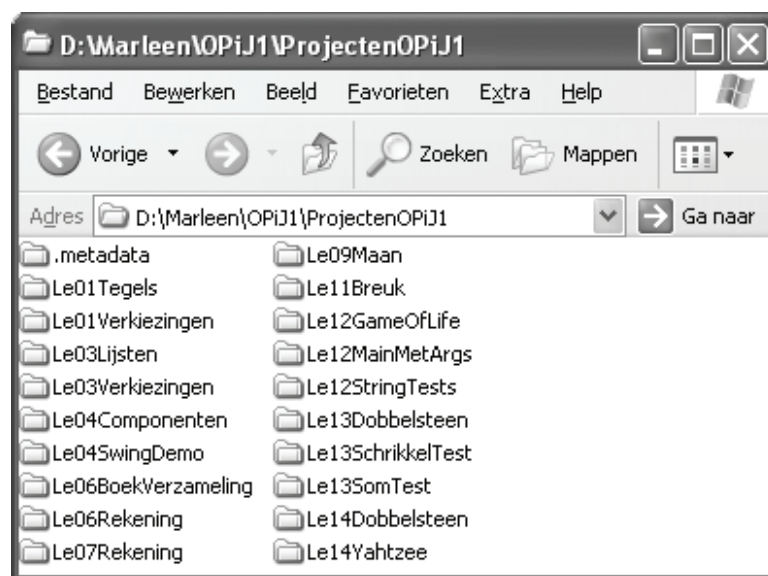
FIGUUR 11 Snelkoppeling maken op Bureaublad

4.6 INSTALLEREN VAN DE BOUWSTENEN

Tot slot dient u de bouwstenen te installeren. Dit zijn delen van programmeerprojecten die u in de loop van de cursus zelf zult gaan voltooien. Deze bouwstenen zijn verzameld in een projectmap voor Eclipse.

– Haal het bestand met bouwstenen op van de cursussite; u vindt dit bestand in de categorie Cursusdocumenten onder de naam ProjectenOPiJ1.zip.

– Open het bestand ProjectenOPiJ1.zip en pak het uit naar een geschikte locatie. Denk er weer aan dat Use folder names moet zijn aangevinkt. Er wordt bij het uitpakken één map ProjectenOPiJ1 aangemaakt met verschillende submappen (zie figuur 12).



FIGUUR 12 De map ProjectenOPiJ1

5 Cursusteam

De eerste versie van deze cursus verscheen in 1997. Dit is de vijfde, ingrijpend herziene versie. We hebben ervoor gekozen om in de colofon dit keer alleen de auteurs te vermelden die hebben meegewerkt aan deze versie, plus de auteurs waarvan nog aanzienlijke stukken tekst uit oude versies zijn gebruikt.

Naast de auteurs genoemd in de colofon, hebben ook de volgende personen in de loop der jaren een bijdrage geleverd.

dhr. dr.ir. J.P.H.W. van den Eijnde
 mw. drs. J.S. Lodder
 dhr. drs. H.J. Pootjes
 mw. M. Witsiers-Voglet

Bij de totstandkoming van deze cursus is dankbaar gebruikgemaakt van het commentaar op eerdere versies van de leereenheden van dhr. prof. dr. S.D. Swierstra (de extern referent) en dhr. dr. ir. A.J.F. Kok.

6 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.

Eindtoets

Bij de cursus hoort een eindtoets die representatief is voor het tentamen. Wij raden u sterk aan deze eindtoets pas te maken als u klaar bent met de tentamenvoorbereiding.

Voor cursisten uit het KHO Gecertificeerd Java-programmeur bestaat de toetsing uit het maken van een viertal programmeeropdrachten, aangevuld met een schriftelijke toets van anderhalf uur. Voorafgaand aan de echte toets zult u in de gelegenheid worden gesteld om een voorbeeldtoets te maken.

7 Overige informatie

Java-web

www.ou.nl/java

Algemene informatie over deze en andere Java-cursussen van de Open Universiteit en over het Kort Hoger Onderwijsprogramma Gecertificeerd Java-programmeur 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.