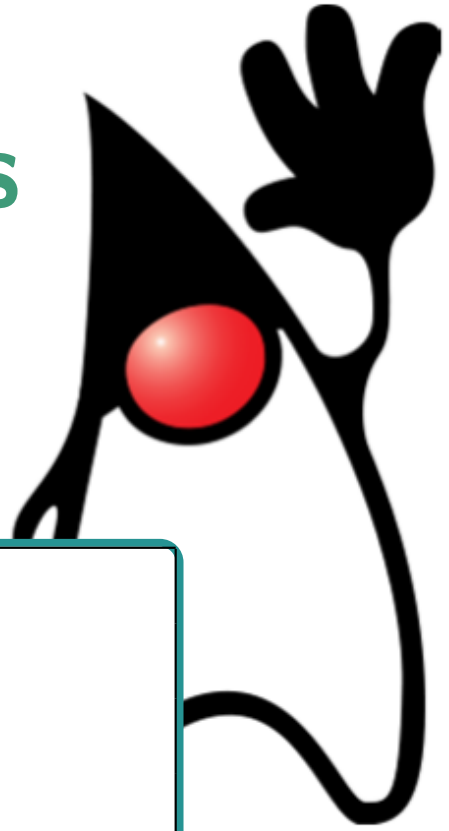
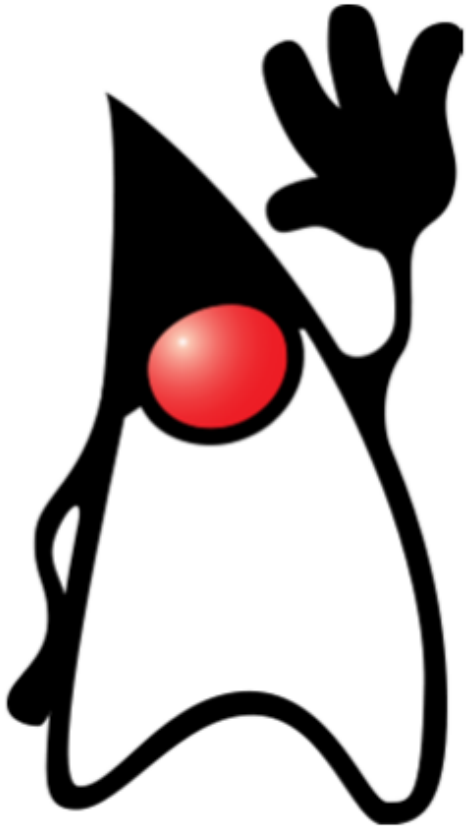
Programming Fundamentals
Inleiding



Klasgroep	1EO-ICT
Opleiding	Bachelor Elektronica-ICT
Theorie	DI: 8:45 - 9:45
	WOE: 11:45 - 12:45
(Werk)Labo	DI: 9:45 - 12:45 + 13:30 - 14:30
	WOE: 13:30 - 15:30 + 15:30 - 17:30
Docent	Katja Verbeeck
Contact	katja.verbeeck@odisee.be

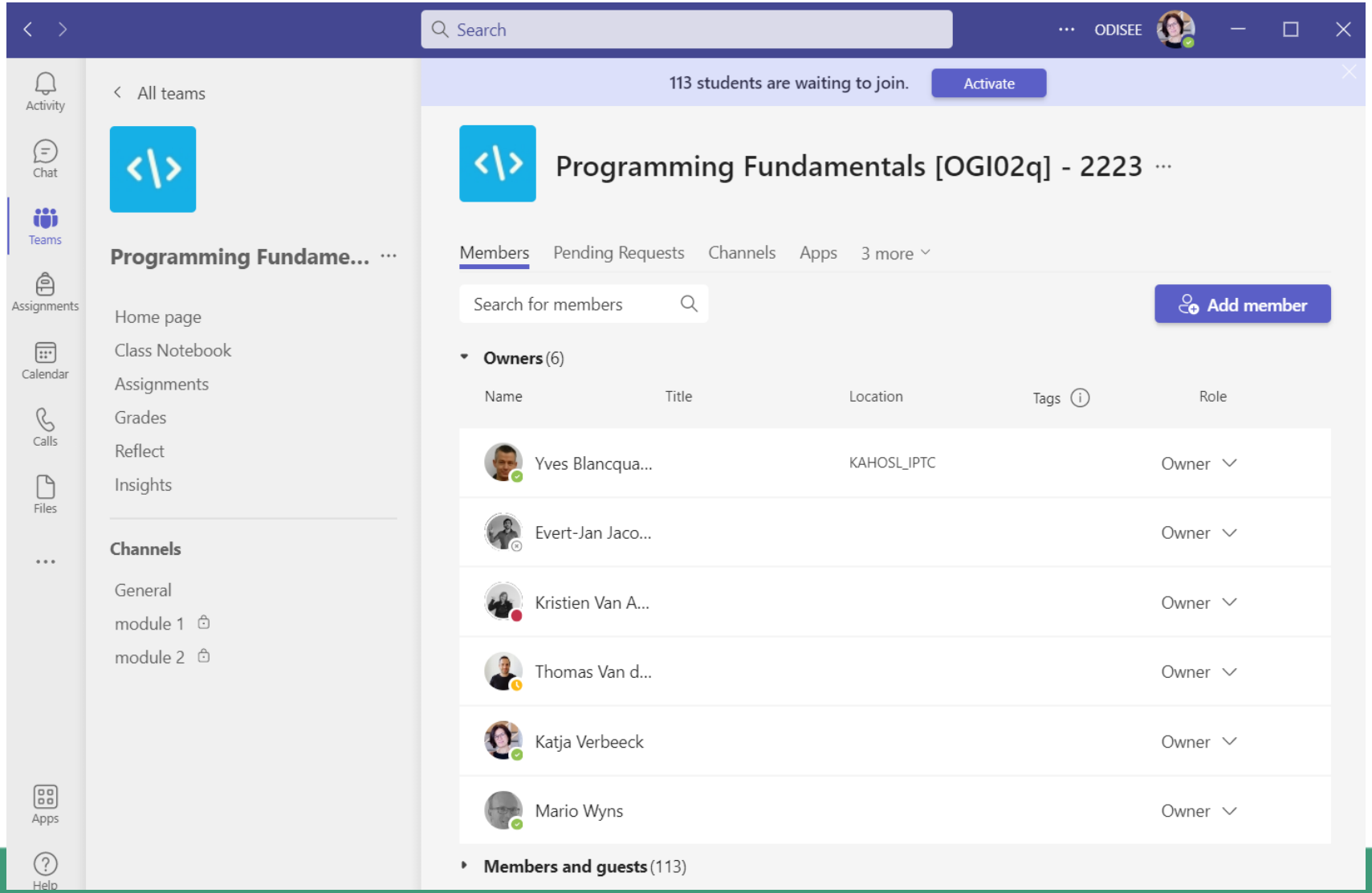
Inhoud



Toledo


Programming Fundamentals [OGI02q]
Uitleg bij het cursusmenu
Mededelingen
E-BOOK
Online theorie les
Inhoud
Opdrachten
Mijn cijfers
Feedback
Evaluatie
Discussieruimte
ECTS
Overzicht studiemateriaal
Tools
Mededelingen
Mededeling maken
Welkom @ Programming fundamentals
Gepost door: Katja Verbeeck
Gepost: donderdag 15 september 2022 8:00:00 uur CEST
Beste studenten,
Welkom in het vak Programming Fundamentals ! Vanaf volgende week op dinsdag 20 september verwachten we de groepen 1 ICT1 tot 1ICT5 in de les - Voor de groepen 1ELO1 tot 1ELO3 en 1ICT6 tot 1ICT7/8 is het wachten op de volgende module die zal starten op dinsdag 22 november.
We gaan het hebben over de basisprincipes van programmeren en zullen dit doen aan de hand van de programmeertaal Java.
We organiseren dit als volgt:
theorie : dinsdag : 8u45 - 9u45 - lokaal EO32 + woensdag : 9u45-10u45 - lokaal JOO9
labo+ werklabo : dinsdag 9u45 - 11u45 - D-gang + 13u30-15u30 Cafetaria Zone3
woensdag: 13u30 - 15u30 D-gang + 15u30-17u30 Cafetaria Zone 3
Jullie kunnen je alvast voorbereiden op het eerste labo door vooraf de ontwikkelomgeving Intellij te installeren op jullie laptop, zo kunnen we het eerste labo meteen van start gaan.
Tip: How to Install Intellij IDEA on Windows 10 + Creating First Hello World Java Application
(https://www.jetbrains.com/help/idea/installation-guide.html en https://www.youtube.com/watch?v=EMLTOMdIz4w).
Installeer de gratis versie (Community Edition).
alvast veel programmeerplezier,

Teams


Search
ODISEE
Activity
Chat
Teams
Assignments
Calendar
Calls
Files
Apps
Help
All teams
Programming Fundame...
Home page
Class Notebook
Assignments
Grades
Reflect
Insights
Channels
General
module 1
module 2
113 students are waiting to join.
Activate
Programming Fundamentals [OGI02q] - 2223
Members
Pending Requests
Channels
Apps
3 more
Search for members
Add member
Owners (6)
Name
Title
Location
Tags
Role
Yves Blancqua...
KAHOSL_IPTC
Owner
Evert-Jan Jaco...
Owner
Kristien Van A...
Owner
Thomas Van d...
Owner
Katja Verbeeck
Owner
Mario Wyns
Owner
Members and guests (113)

Werkwijze

Periode van 7 weken

Onderdelen : Java theorie - Java oefeningen - Java project

Java theorieles 2 x 1u di + woe

Java labo 2 x 2u

Java (zelfstandig)werklabo 2 x 2u De werklabo's zijn **niet vrijblijvend**. Je begeleidende labodocent beslist op basis van je werk of je vrijgesteld bent.

Het Teams kanaal is aanvullend en is de plaats om vragen te posten.

Begeleidende docenten

Module 2 :

- **1ICT6(LINFO3)** Evert-Jan Jacobs (evertjan.jacobs@odisee.be)
- **1ICT7(LINFO4)** Katja Verbeeck (katja.verbeeck@odisee.be)
- **1ICT8-1ELO1 (LINFO5)** Kristien Van Assche (kristien.vanassche@odisee.be)
- **1ELO2 (LINFO6)** Yves Blanquaert (yves.blancquaert@odisee.be)
- **1ELO (LINFO3-LINFO6)** Thomas Van den Bossche (thomas.vandenbossche@odisee.be)

1ste examenkans : permanente evaluatie

Theorie (20%)	**scantesten** : korte theoretische denkvragen
Week 3	t.e.m. Java API en prim operaties (4%)
Week 5	t.e.m. iteraties en selecties (6%)
Week 7	alles :-) (10%)
Praktijk (80%)	**Individuele Java oefening** op PC + project
week 3	t.e.m. Java API en prim operaties (10%)
week 5	t.e.m. iteraties en selecties (20%)
week 7	alle geziene topics (30%)
week 7	**Project** : Java oefening (20%)

Evaluatie : 2de examenkans

Je beslist zelf welk onderdeel (theorie - individuele oef - project) je hermaakt. Bij voorkeur alle onderdelen waarvoor je niet geslaagd was, en minstens 1 uiteraard.

Theorie (20%)	Toledotest tijdens examenperiode (20%)
Praktijk (80%)	Individuele Java laboproef (60%) + project Java (20%)

Lesmateriaal

- Alle materiaal : slides, opgaven, testen, ... beschikbaar op Toledo
- Java API online
- Goed boek : **Java A Beginner's Guide** , 6th Edition
  by **Herbert Schildt**
  Publisher: McGraw-Hill Osborne Media
  ISBN: 9780071809252
- Trail: Learning the Java Language, online @
  `https://docs.oracle.com/javase/tutorial/java/index.html`

Inhoud

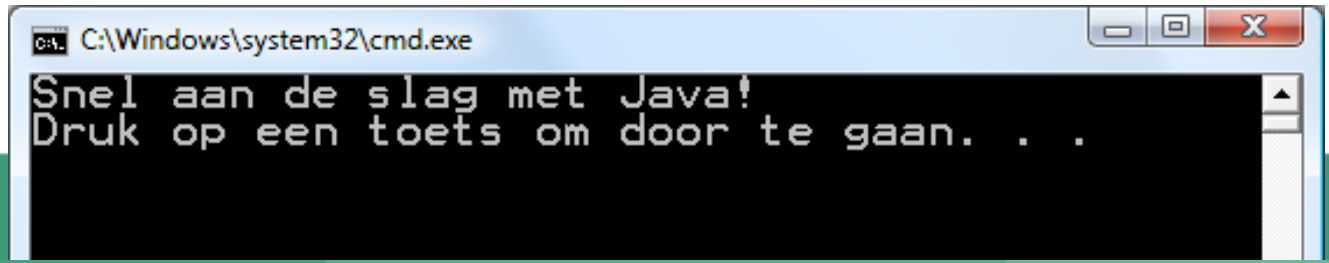
Java

- First cup of Java
- Primitieve types en operaties
- Java API (oa String, Math, Scanner)
- Selecties
- Iteraties
- Methoden
- Intro OO
- Arrays

MijnEersteVoorbeeld.java

MijnEersteVoorbeeld.java

```
public class MijnEersteVoorbeeld{
    public static void main(String[] args){

        System.out.println("Snel aan de slag
            met Java!");
    }

}
```



Stappen

1. editeren
2. opslaan
3. compileren

```
javac MijnEersteVoorbeeld.java

-> MijnEersteVoorbeeld.class
```

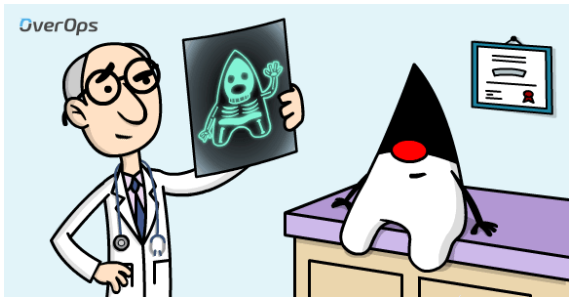
4. uitvoeren

```
java MijnEersteVoorbeeld
```

De Java Virtuele Machine

Java wordt gecompileerd naar bytecode voor een virtuele machine. De Java Virtuele Machine (Java VM of kortweg JVM). Er is een JVM beschikbaar voor allerlei verschillende soorten computers. Hierdoor is de gecompileerde bytecode platformonafhankelijk, d.w.z. de code kan op elke willekeurige computer draaien, onafhankelijk van het besturingssysteem (zoals Windows, Unix of Mac OS X).

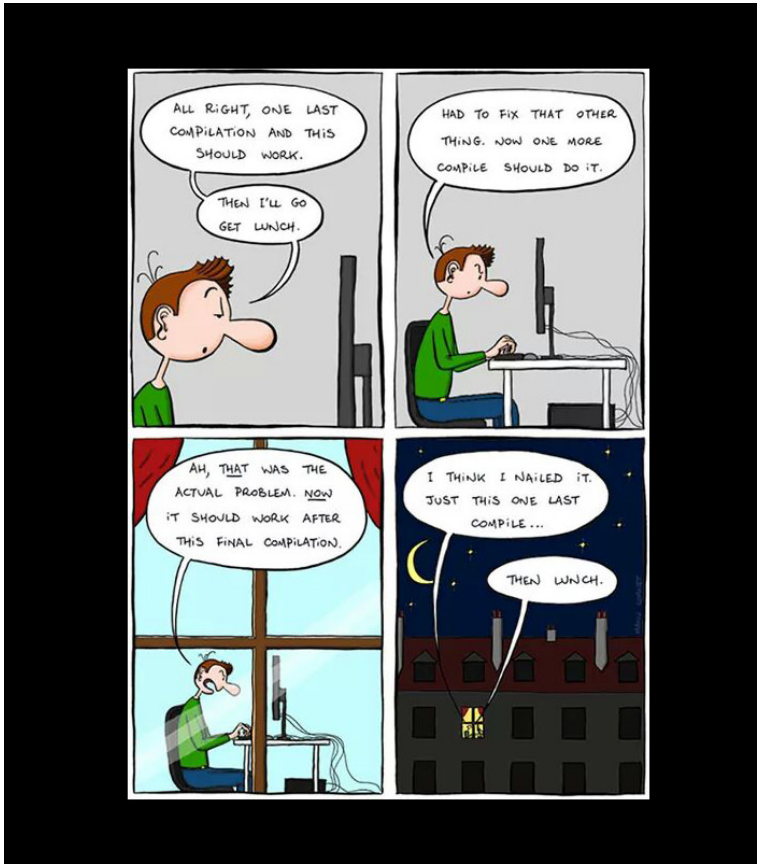
"Write once, run anywhere"



Om een gegeven .class bestand te kunnen gebruiken moet wel een compatibele JVM geïnstalleerd zijn.

Compileren ....

99 little bugs in the code,
99 bugs in the code,
1 bug fixed...compile again,
100 little bugs in the code.



Garbage Collection ....

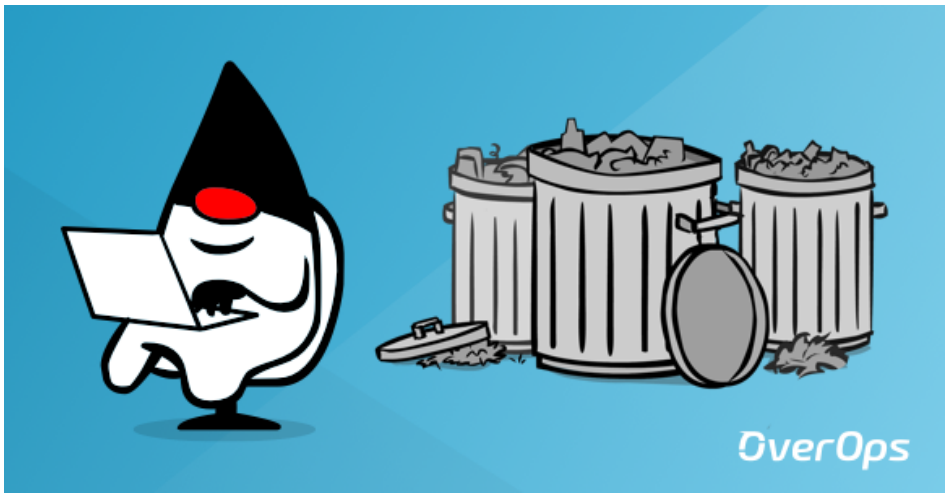
This is Java

Java does not leave it's mess for others to clean.

It collects it's own garbage.

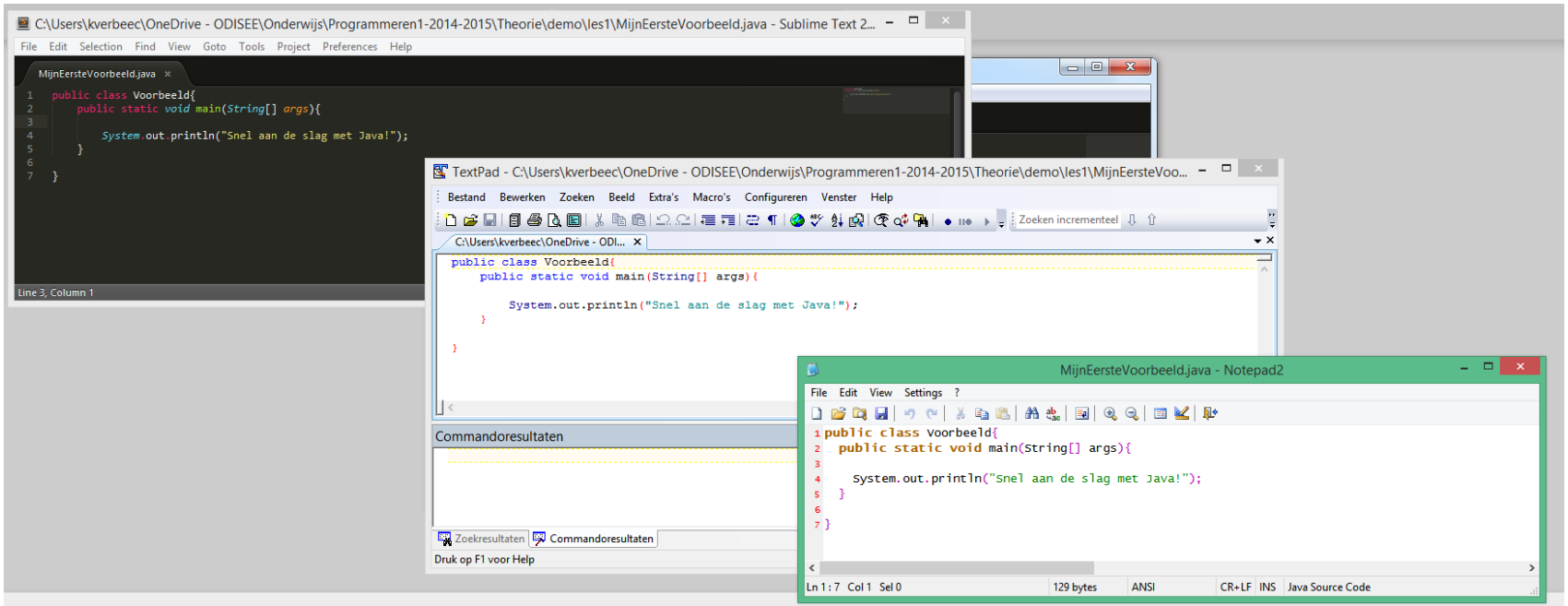
Be like Java.
Keep your country clean.

@anshmeme's



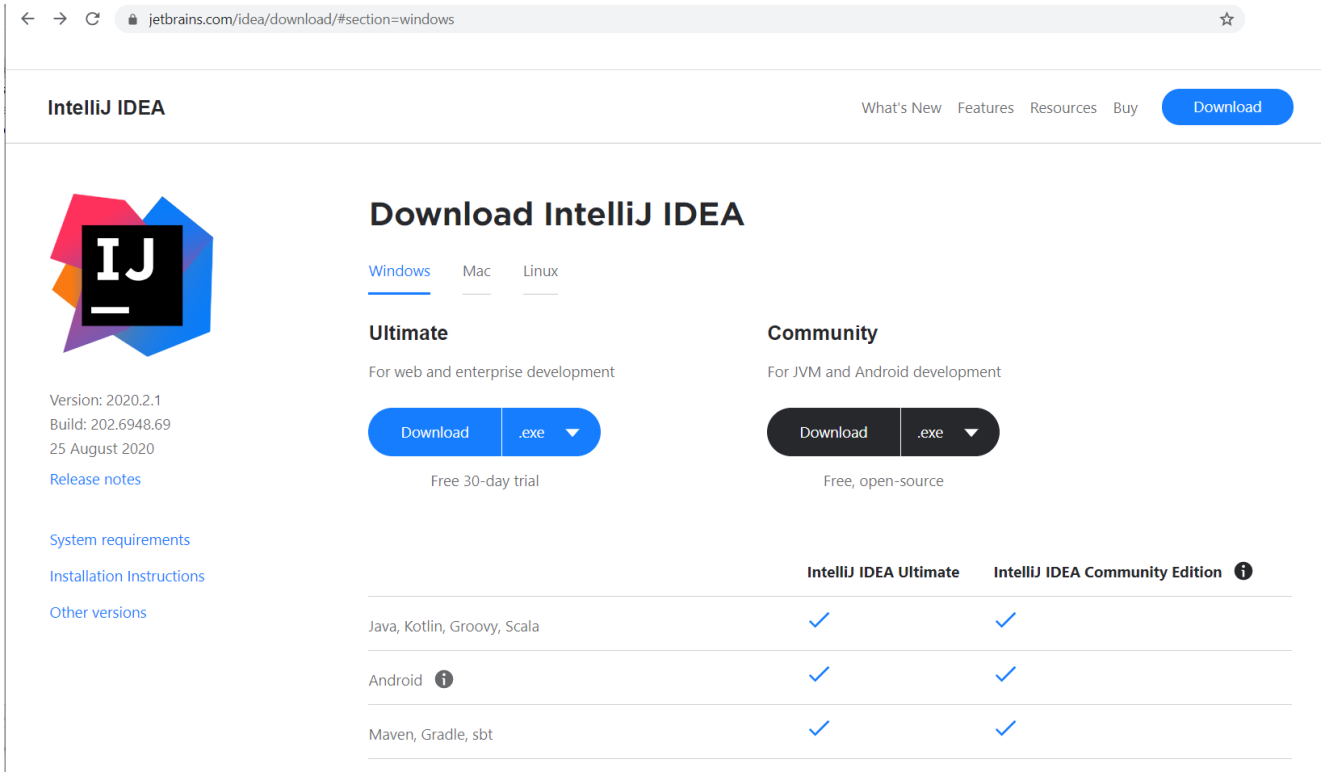
Ontwikkelomgeving of IDE

De ontwikkelomgeving of IDE (integrated development environment) kan een simpele tekstverwerker zijn :



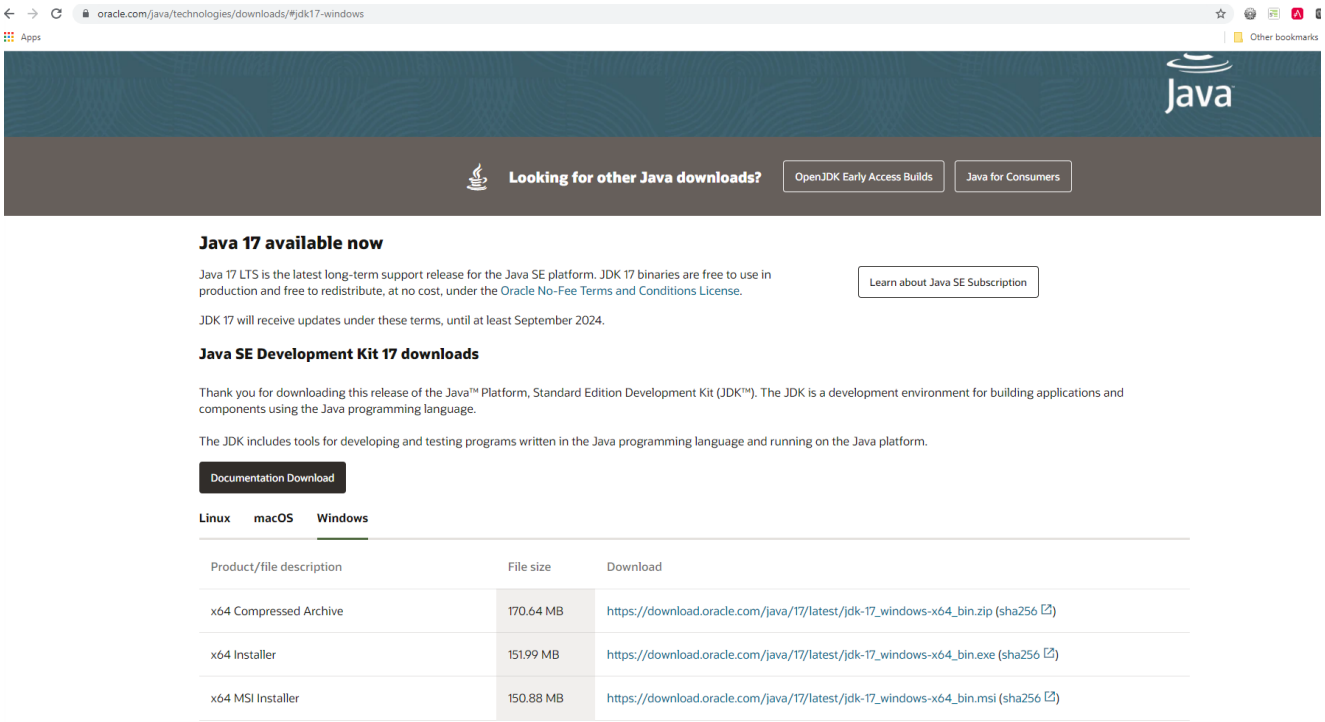
IntelliJ

Wij werken in het labo met de community edition van IntelliJ IDEA

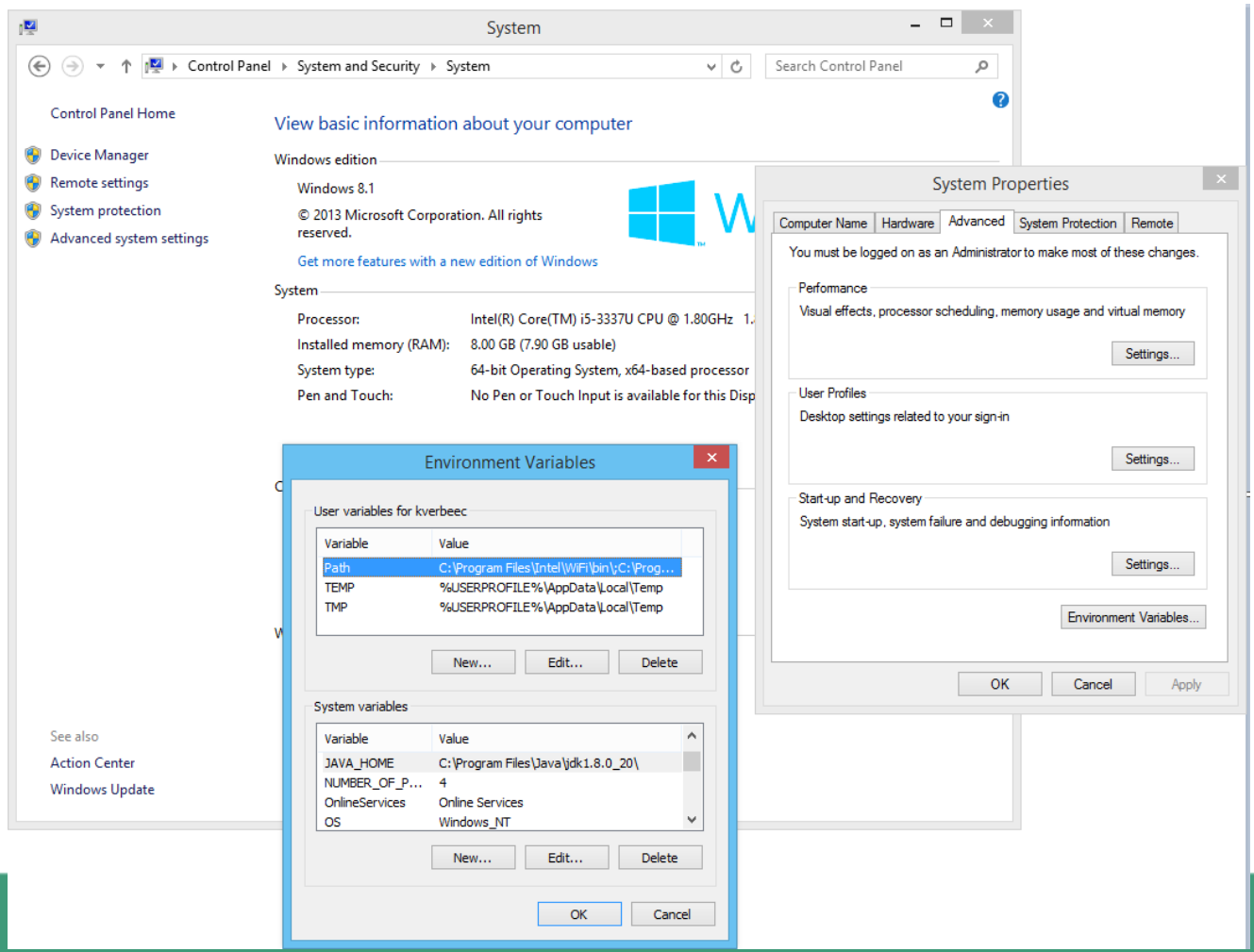


Java installeren

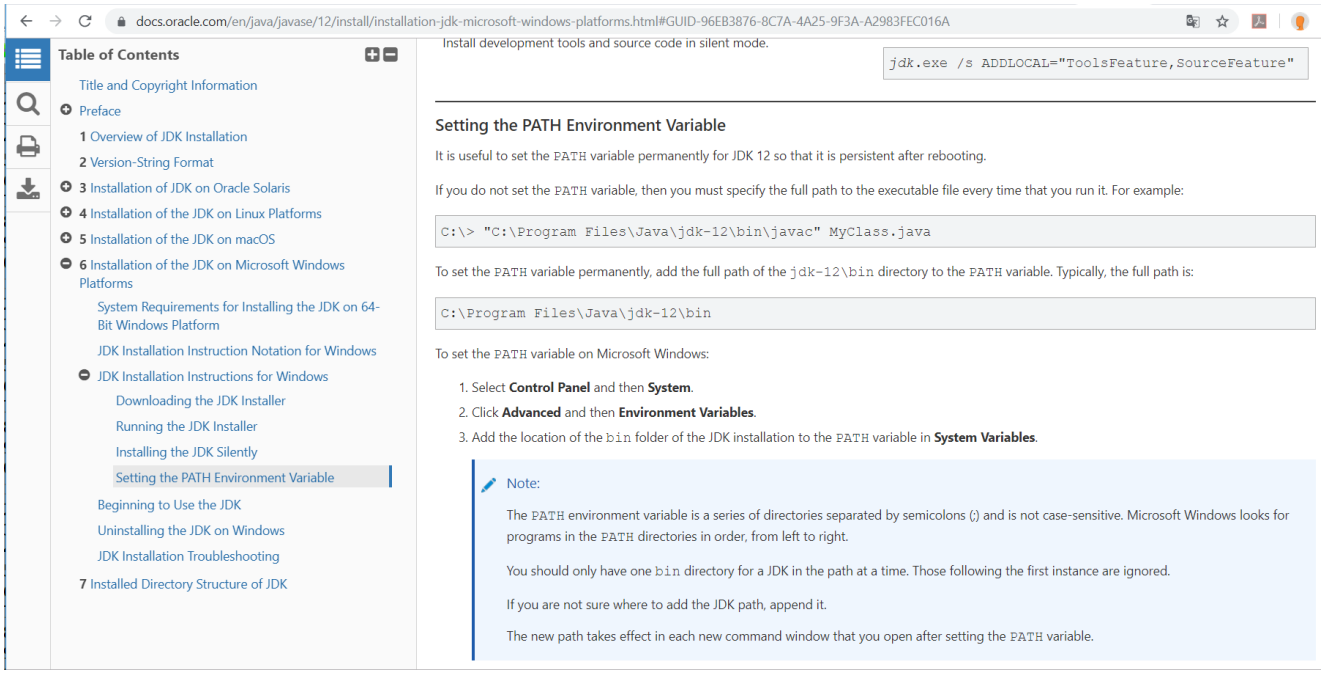
Je kan rechtstreeks de jdk installeren, zie :
`https://www.oracle.com/java/technologies/downloads/`



Instellen omgevingsvariabelen (path)



Oracle Installation Guide



Installatie IntelliJ

- How to Install Intellij IDEA on Windows 10 + Creating First Hello World Java Application (`https://www.youtube.com/watch?v=EMLTOMdIz4w`)
- Creating your first Java application with IntelliJ IDEA (2020) (`https://www.youtube.com/watch?v=H_XxH66lm3U`)

Fouten at compile-time = syntaxfouten

Naam van een klasse

Hallo.java

```
public class hallo {
    public static void main(String[] args) {
        System.out.println("Dag 1 EO-ICT !!!");
    }
}
```

Fouten at compile-time = syntaxfouten

Naam van een klasse

```
C:\java\>javac Hallo.java
Hallo.java:1: class hallo is public, should be declared in a file
named hallo.java

public class hallo {
       ^
1 error
```

Fouten at compile-time = syntaxfouten

Naam van een klasse

Hallo.java

```
public class Hallo {
    public static void main(String[] args) {
        System.out.println("Dag 1 EO-ICT !!!");
    }
}
```

Fouten at compile-time = syntaxfouten

Naam van een klasse

- Java is case-sensitive
- bestandsnaam = naam van de klasse

Fouten at compile-time = syntaxfouten

Hallo.java

```
public class Hallo {
    public static void main(String[] args) {
        System.out.println("Dag 1 EO-ICT !!!")
    }
}
```

Fouten at compile-time = syntaxfouten

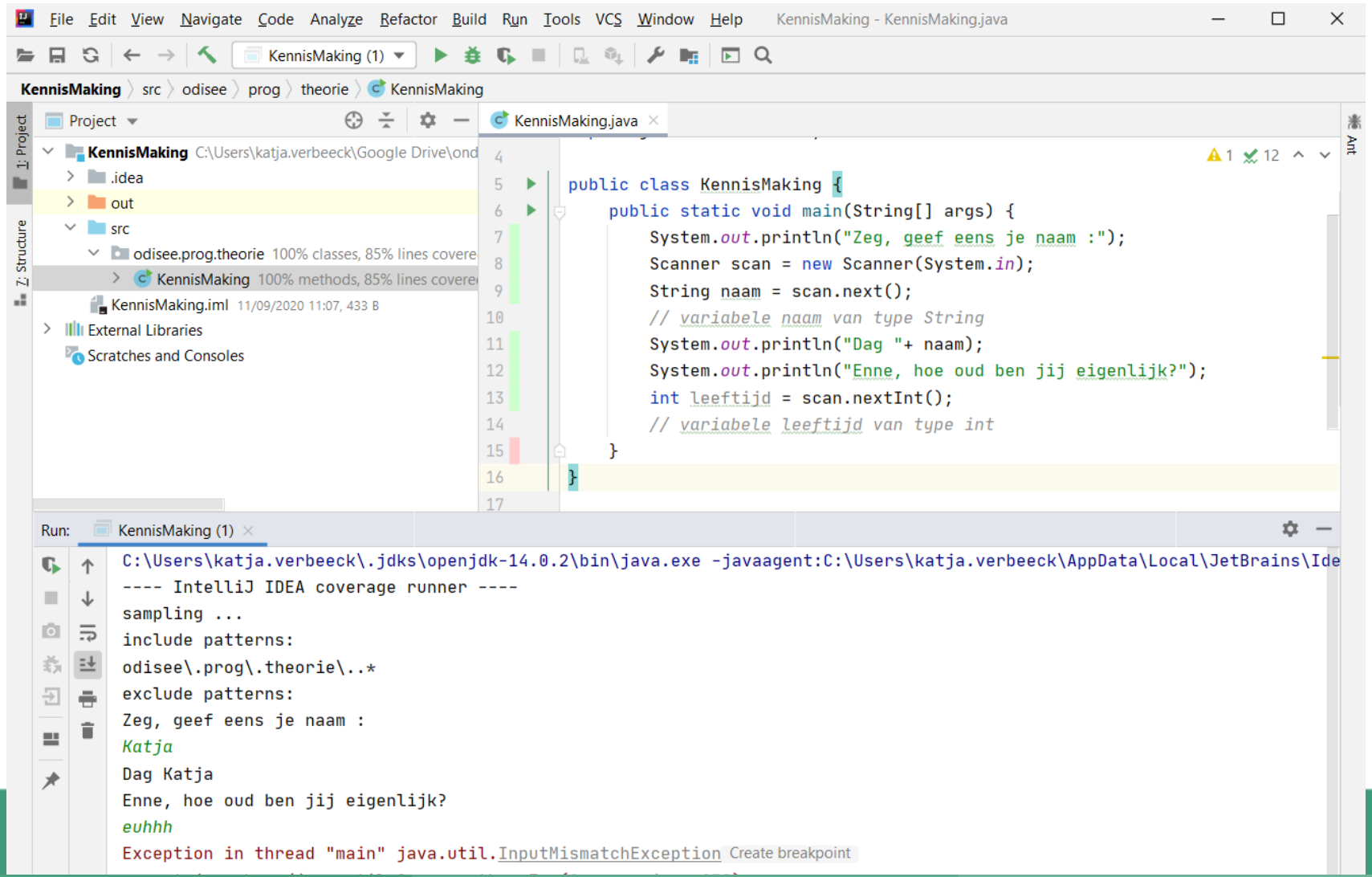
```
C:\java\>javac Hallo.java
Hallo.java:3: ';' expected
        System.out.printIn("Hallo 1ICT!")
                                          ^
1 error
```

Fouten at runtime = semantische fouten

KennisMaking.java

```
import java.util.Scanner;
public class KennisMaking {

    public static void main(String [] args) {
        System.out.println("Zeg, geef eens je naam :");
        Scanner scan = new Scanner(System.in);
        String naam = scan.next();
        // variabele naam van type String
        System.out.println("Dag "+ naam);
        System.out.println("Enne, hoe oud ben jij
            eigenlijk?");
        int leeftijd = scan.nextInt();
        // variabele leeftijd van type int
    }
}
```

Fouten at runtime = semantische fouten

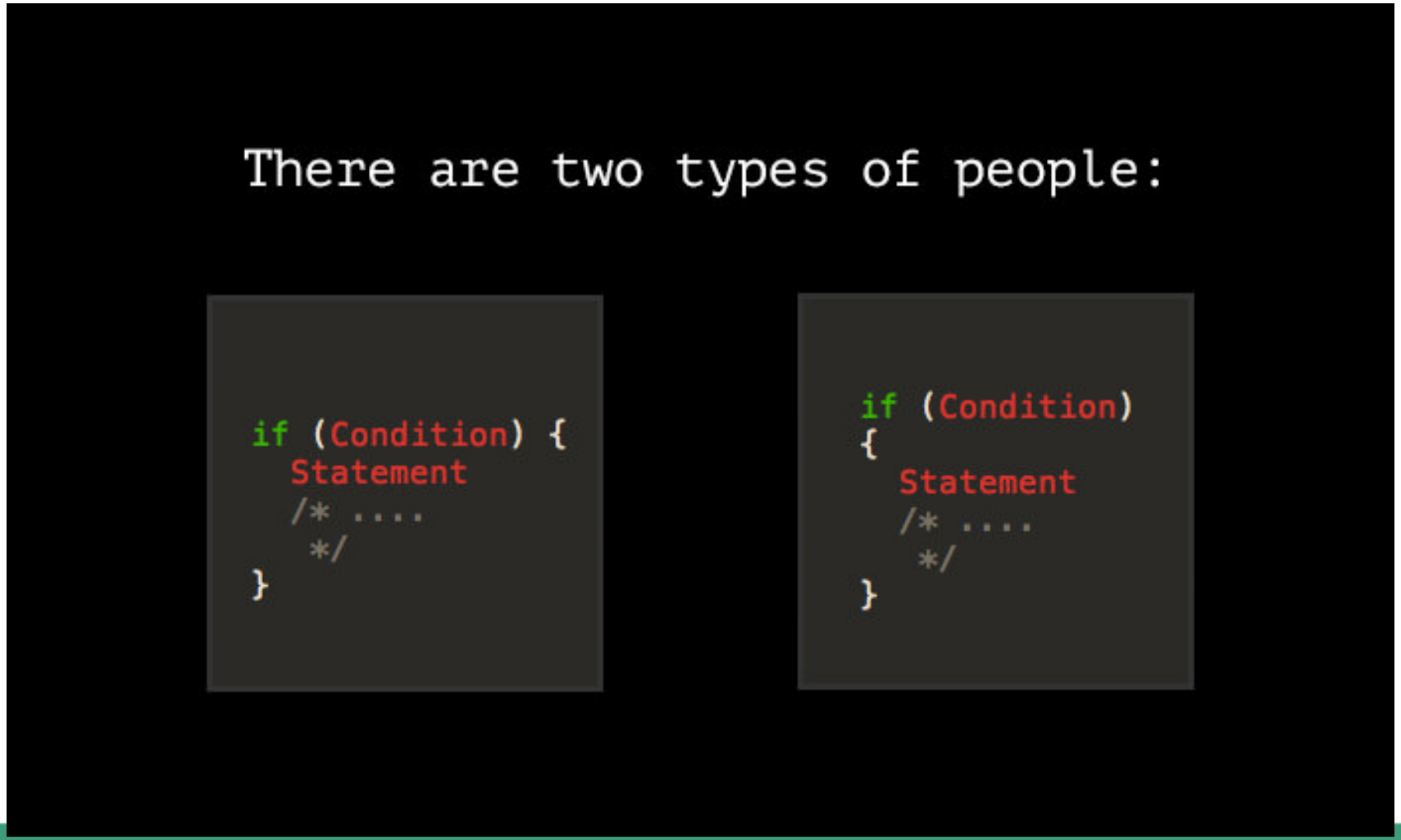


Stijl

variabelen steeds met kleine letter! laat een blok code inspringen !

```
import java.util.Scanner;
public class KennisMaking2 {

    public static void main(String [] args) {
    System.out.println("Zeg, geef eens je naam :");
    Scanner SCAN = new Scanner(System.in); // FOUT !
    String NAAM = SCAN.next(); // FOUT !!
     // variabele naam van type String
    System.out.println("Dag "+ NAAM);
    System.out.println("Enne, hoe oud ben jij
        eigenlijk?");
    int leeftijd = SCAN.nextInt();
     // variabele leeftijd van type int
    } // Gebruik indentatie !!
}
```

Stijl : indentatie !



Stijlfouten

- Stijlregel 1 : Naam van een klasse steeds met een Hoofdletter beginnen! Is ook het geval voor **ALLE** andere klassen in de java klassenbibliotheek!!! – (bvb. System, Math, String, . . .)
- Stijlregel 2 : Laat elke Java methodenaam beginnen met een kleine letter – Is ook het geval voor **ALLE** andere methoden in de java klassenbibliotheek!!! – (bvb. println, . . .)
- Stijlregel 3 : Laat elke Java variabele beginnen met een kleine letter
- Stijlregel 4 : Gebruik CamelCasing voor lange namen : klasse **K**ennis**M**aking, **E**nergie**V**erbruik, methode **b**eschrijf**J**ezelf, ...
- er volgen nog stijlregels voor indentatie, constanten, ...

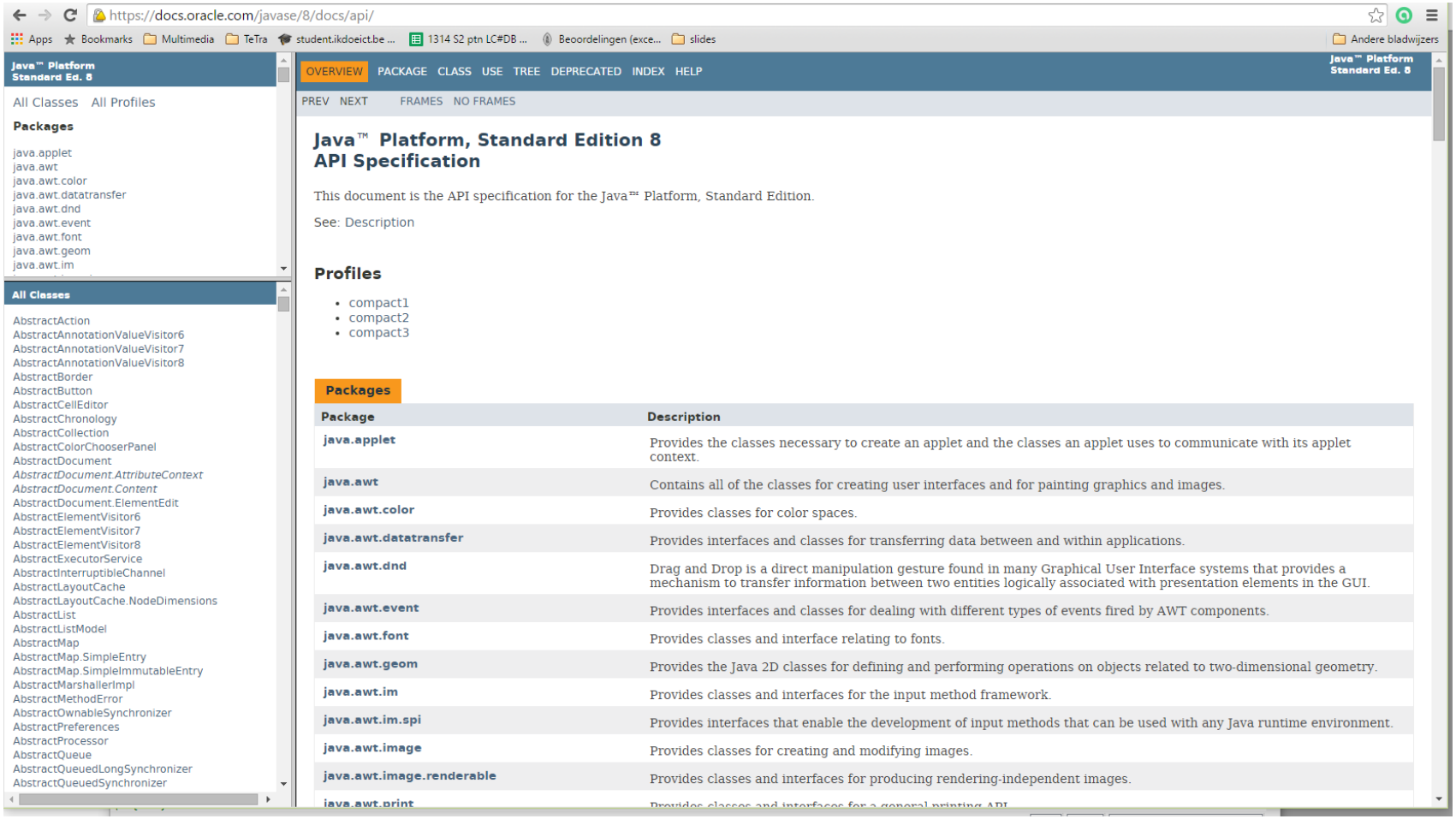
Commentaar

```
/*
   Commentaar Stijl 1 : meerdere lijnen
   Dit is een eenvoudig Java voorbeeld.
*/

public class Hallo {
    // Commentaar Stijl 2 : slechts 1 lijn

    public static void main(String[] args) {

        System.out.println("Dag 1 EO-ICT !!!");
    }
}
```

Javadoc

Javadoc is een programma, bijgeleverd bij de jdk, waarmee je html-documentatie pagina's kan genereren gebruik makend van **javadoc** tags. Merk op, de volledige java API is eveneens op deze manier gedocumenteerd : `https://docs.oracle.com/javase/8/docs/api/` zie ook boek, appendix B

De tags die wij alvast hanteren voor elke gemaakt oefening zijn **@author** en **@version** om de header van elk programma te beschrijven. Later zullen meer tags gebruikt worden.


https://docs.oracle.com/javase/8/docs/api/
Java™ Platform Standard Ed. 8
All Classes All Profiles
Packages
java.applet
java.awt
java.awt.color
java.awt.datatransfer
java.awt.dnd
java.awt.event
java.awt.font
java.awt.geom
java.awt.im
All Classes
AbstractAction
AbstractAnnotationValueVisitor6
AbstractAnnotationValueVisitor7
AbstractAnnotationValueVisitor8
AbstractBorder
AbstractButton
AbstractCellEditor
AbstractChronology
AbstractCollection
AbstractColorChooserPanel
AbstractDocument
AbstractDocument.AttributeContext
AbstractDocument.Content
AbstractDocument.ElementEdit
AbstractElementVisitor6
AbstractElementVisitor7
AbstractElementVisitor8
AbstractExecutorService
AbstractInterruptibleChannel
AbstractLayoutCache
AbstractLayoutCache.NodeDimensions
AbstractList
AbstractListModel
AbstractMap
AbstractMap.SimpleEntry
AbstractMap.SimpleImmutableEntry
AbstractMarshallerImpl
AbstractMethodError
AbstractOwnableSynchronizer
AbstractPreferences
AbstractProcessor
AbstractQueue
AbstractQueuedLongSynchronizer
AbstractQueuedSynchronizer
OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP
PREV NEXT FRAMES NO FRAMES
Java™ Platform, Standard Edition 8
API Specification
This document is the API specification for the Java™ Platform, Standard Edition.
See: Description
Profiles
compact1
compact2
compact3
Packages
Package
Description
java.applet
Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt
Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color
Provides classes for color spaces.
java.awt.datatransfer
Provides interfaces and classes for transferring data between and within applications.
java.awt.dnd
Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI.
java.awt.event
Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font
Provides classes and interface relating to fonts.
java.awt.geom
Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.
java.awt.im
Provides classes and interfaces for the input method framework.
java.awt.im.spi
Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.
java.awt.image
Provides classes for creating and modifying images.
java.awt.image.renderable
Provides classes and interfaces for producing rendering-independent images.
java.awt.print

```java
/**
* Deze klasse print een welkomstwoord
* op het scherm
* @author Katja Verbeeck
* @version september 2015
*/

// let op een javadoc header start steeds
   met /** en eindigt met */

public class Header {
    public static void main(String[] args) {
        System.out.println(" De eerste
            javadoc-tags ");
    }
}
```

Javadoc header in IntelliJ