

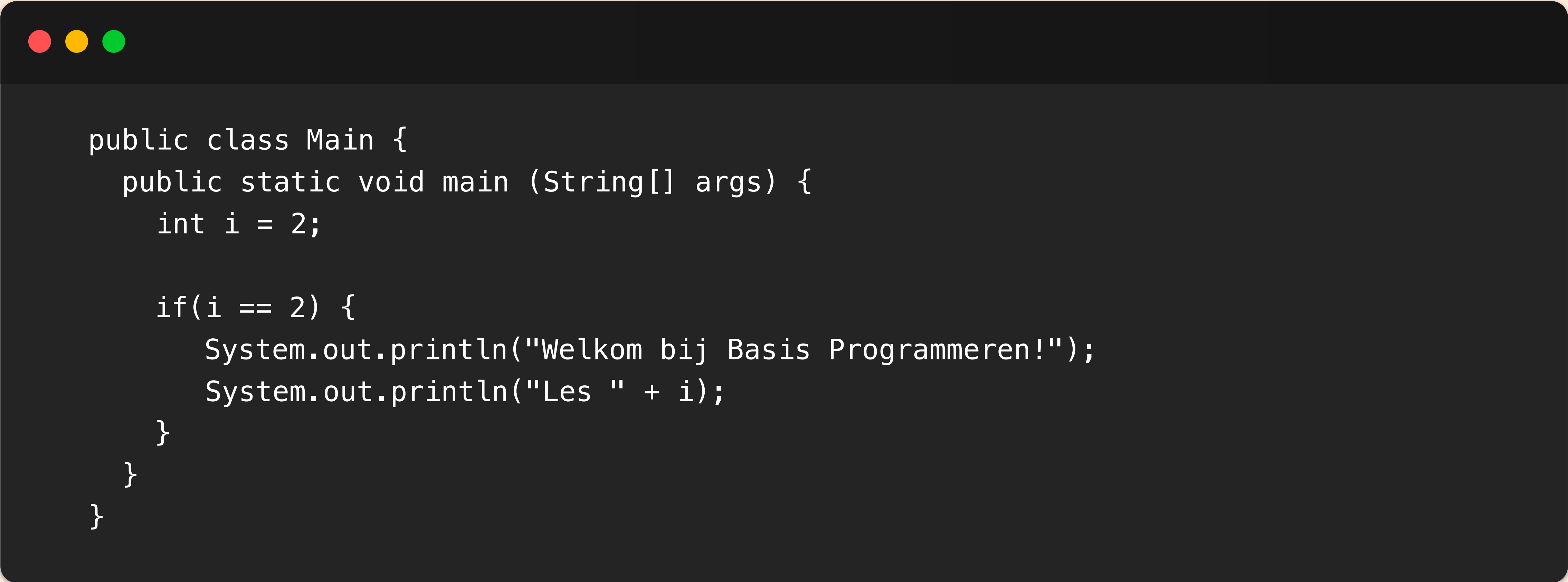
Full Stack Developer Bootcamp

Basis Programmeren

Les 2

Basis Programmeren

Les 2



```
public class Main {  
    public static void main (String[] args) {  
        int i = 2;  
  
        if(i == 2) {  
            System.out.println("Welkom bij Basis Programmeren!");  
            System.out.println("Les " + i);  
        }  
    }  
}
```

Lesplanning

- Terugblik les 1
- Herhalingsconstructies
- Java demonstratie
- *Opdracht: Euro/dollar converter - deel 2*

Terugblik les 1

Software Installatie

IntelliJ IDEA & WebStorm

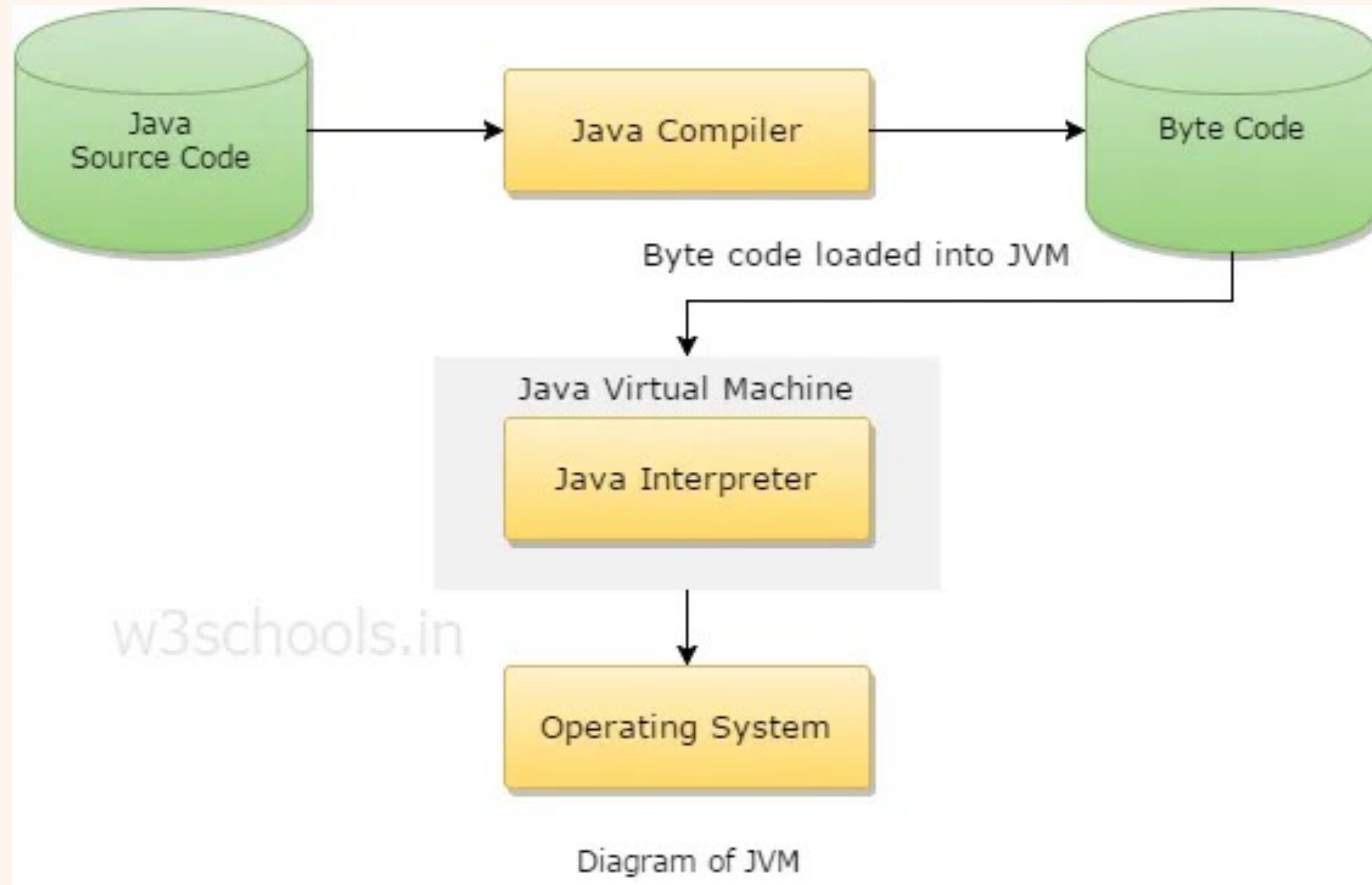
- IntelliJ geïnstalleerd met JDK. Eventueel ook WebStorm geïnstalleerd
- Daarnaast ook JetBrains licentie aangevraagd met @novi-education.nl mail.
- Handige plugin voor de back-end opleiding, **SequenceDiagram**:
<https://plugins.jetbrains.com/plugin/8286-sequencediagram>
- Tip van student uit vorige groep, **Rainbow Brackets**:
<https://plugins.jetbrains.com/plugin/10080-rainbow-brackets>
- Zie documentatie IntelliJ:
<https://www.jetbrains.com/help/idea/managing-plugins.html>

Java

- Java 1.0 in 1995 uitgebracht door SUN
 - Hedendaags Java 18 (uitgebracht in maart 2022)
 - Er wordt veel gewerkt met Java versie 11 of 16 vanwege lange support
- Object georiënteerde programmeertaal
- Java is platform onafhankelijk, het draait op Windows, Apple, Linux, Android, etc.
- Java-code moet gecompileerd worden naar byte-code
- Gecompileerde byte-code wordt door de Java Virtual Machine ingeladen
- Voor de cursus Basis Programmeren zal Java versie 16 of versie 17 beide geschikt zijn.

Java Virtual Machine

JVM



→ Bron: <https://www.w3schools.in/java/java-virtual-machine>

Java

Theorie

- .java-bestanden voor Java broncode
- .class-bestanden voor gecompileerde Java-code (= bytecode)

- Diverse soorten applicaties mogelijk met Java:

 - Console (command line interface) applicaties

 - Grafische desktop applicaties

 - Web applicaties

 - Android applicaties

- Afkortingen:

 - IDE = Integrated Development Environment, JVM = Java Virtual Machine

 - JDK = Java Development Kit (compileren), JRE = Java Runtime Environment (uitvoeren)

Informatie opslaan

Variabelen

→ Informatie wordt opgeslagen in variabelen

Denk aan wiskunde, bijv stelling van Pythagoras $a^2 + b^2 = c^2$

→ Variabelen hebben bij Java een **type**, een **naam** en een **waarde**

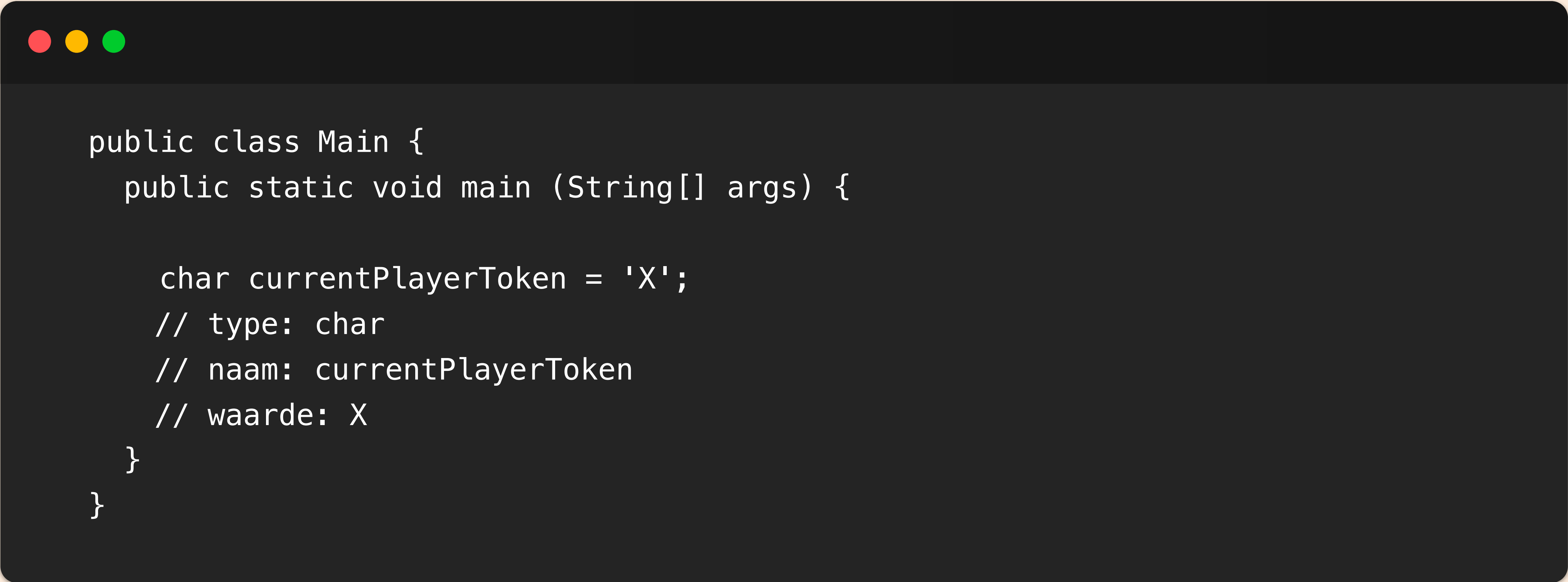
→ type geeft aan wat voor informatie in die variabele wordt opgeslagen

→ naam beschrijft de informatie die opgeslagen wordt

→ waarde is de opgeslagen informatie zelf

Informatie opslaan

Voorbeeld-code



```
public class Main {  
    public static void main (String[] args) {  
  
        char currentPlayerToken = 'X';  
        // type: char  
        // naam: currentPlayerToken  
        // waarde: X  
    }  
}
```

Primitieve typen

Variabelen

- Gehele getallen: byte, short, **int**, long
- Reëel getallen: float, **double**
- 1 karakter: char
- Logisch: boolean (true / false)

Boolean

Variabelen

- Datatype Boolean is vernoemd naar **George Boole**
- 19-eeuwse Britse wiskundige en logicus
- Geboren in 1815, lang voordat programmeren in Java een ding was
- Uitvinder van de Booleaanse logica:

Iets kan WAAR (1) of ONWAAR (0) zijn

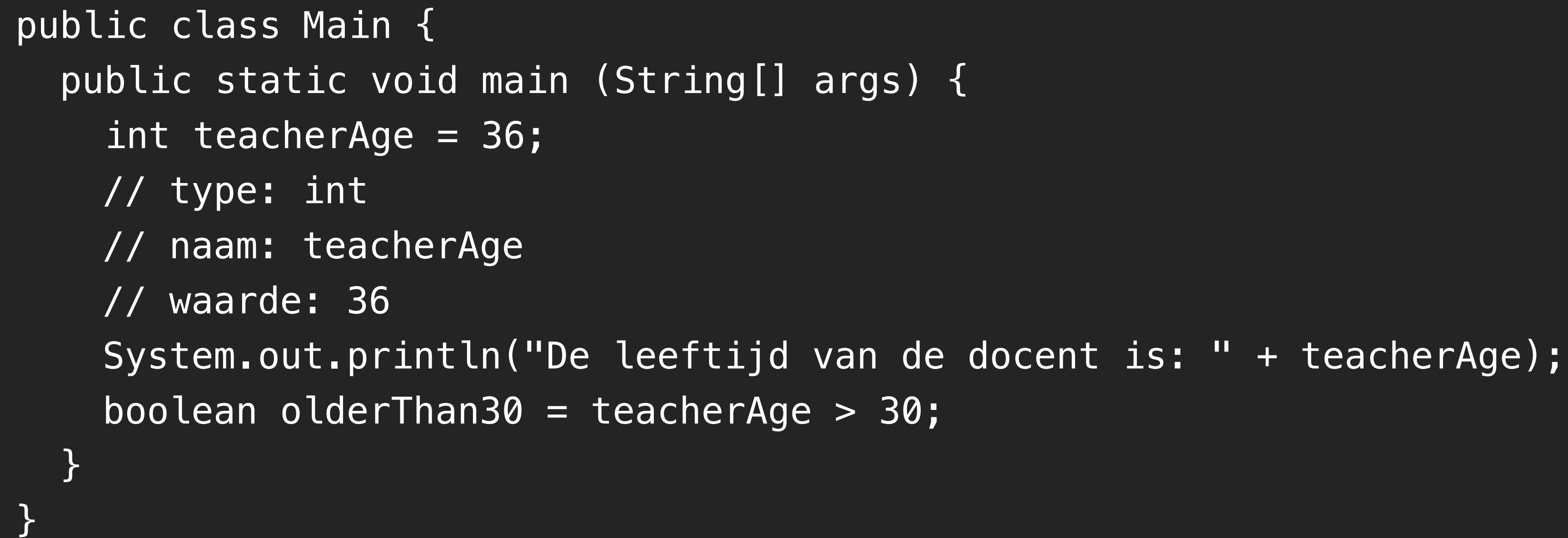
Bijvoorbeeld de stelling: “De zon schijnt” kan waar of onwaar zijn

De logische operators AND, OR en NOT komen ook uit de booleaanse logica



Informatie printen

Voorbeeld-code



```
public class Main {  
    public static void main (String[] args) {  
        int teacherAge = 36;  
        // type: int  
        // naam: teacherAge  
        // waarde: 36  
        System.out.println("De leeftijd van de docent is: " + teacherAge);  
        boolean olderThan30 = teacherAge > 30;  
    }  
}
```

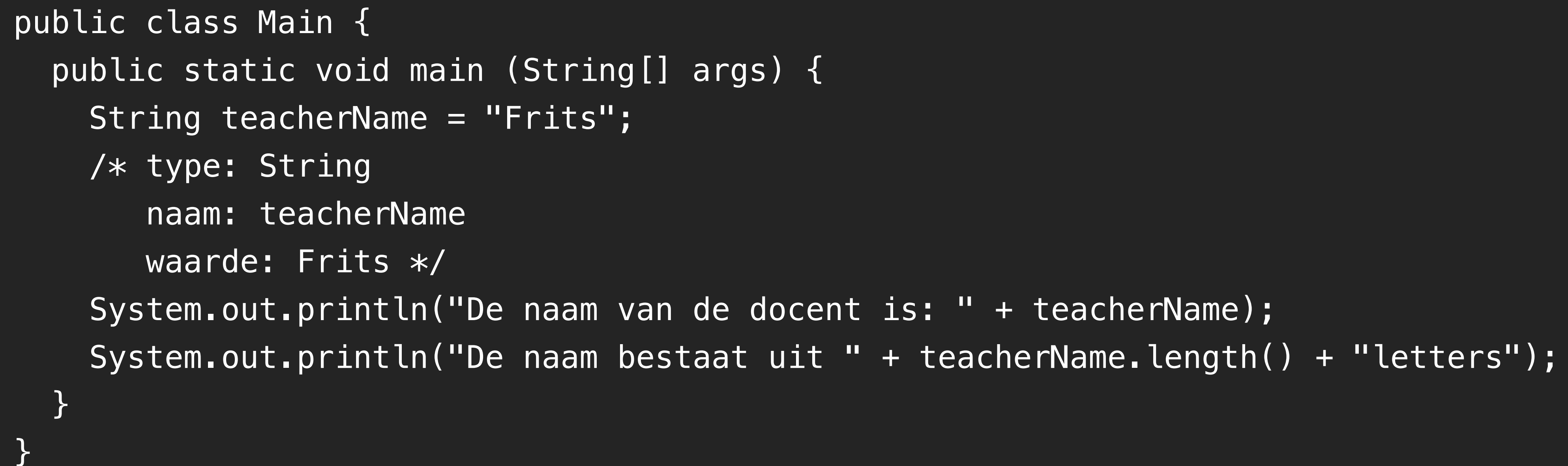
Tekst opslaan

String-variabelen

- Tekst wordt opgeslagen in een variabele van het type **String**
- String is geen primitieve datatype
- Bij een String-variabele worden objecten gebruikt van het type String
- Daardoor is het mogelijk om methoden uit te voeren, bijvoorbeeld om de lengte van de tekst op te vragen (*length*) of om de tekst om te zetten naar hoofdletters (*toUpperCase*)
- Zie String-methoden op: https://www.w3schools.com/java/java_ref_string.asp

Tekst opslaan en printen

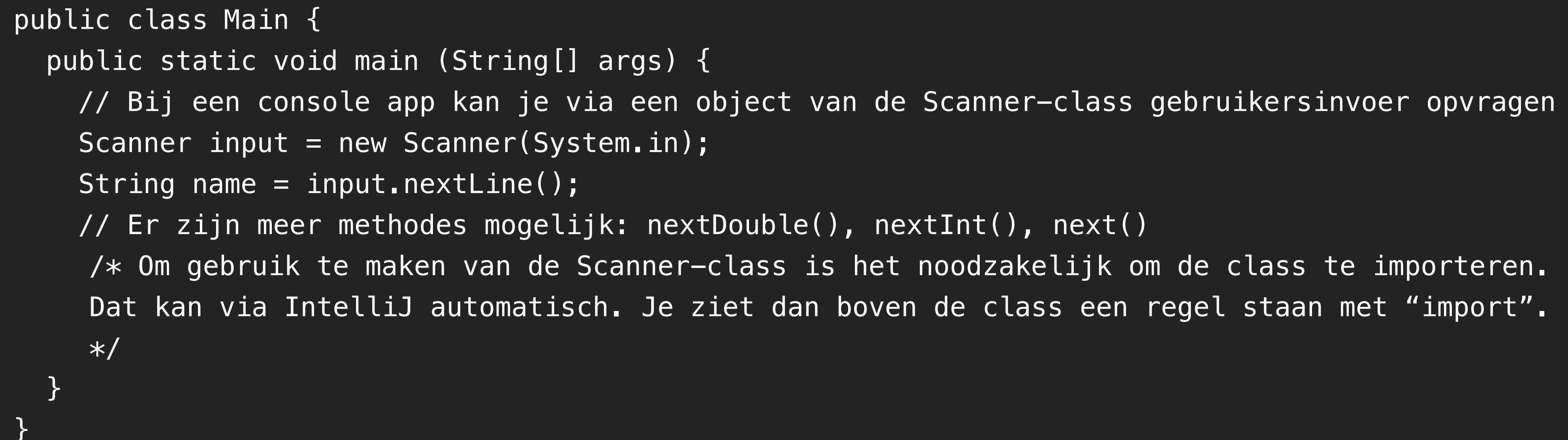
Voorbeeld-code



```
public class Main {  
    public static void main (String[] args) {  
        String teacherName = "Frits";  
        /* type: String  
           naam: teacherName  
           waarde: Frits */  
        System.out.println("De naam van de docent is: " + teacherName);  
        System.out.println("De naam bestaat uit " + teacherName.length() + "letters");  
    }  
}
```

Gebruikersinvoer verwerken

Voorbeeld-code



```
public class Main {  
    public static void main (String[] args) {  
        // Bij een console app kan je via een object van de Scanner-class gebruikersinvoer opvragen  
        Scanner input = new Scanner(System.in);  
        String name = input.nextLine();  
        // Er zijn meer methodes mogelijk: nextDouble(), nextInt(), next()  
        /* Om gebruik te maken van de Scanner-class is het noodzakelijk om de class te importeren.  
        Dat kan via IntelliJ automatisch. Je ziet dan boven de class een regel staan met "import".  
        */  
    }  
}
```


Keuzeconstructies

If/else statement vs **switch** statement

- Zie code op: <https://replit.com/@NOVI/Huisnummers>
- Klik op 'Show Files' om de code van Main.java te bekijken
- Druk op de groene Run-knop om de code uit te voeren
- Vergelijk de code van bovenstaande URL met onderstaande URL:
<https://replit.com/@NOVI/Switch-statement>

Opdracht les 1

Euro/Dollar converter

GELUKT?? Vandaag deel 2 van deze opdracht.

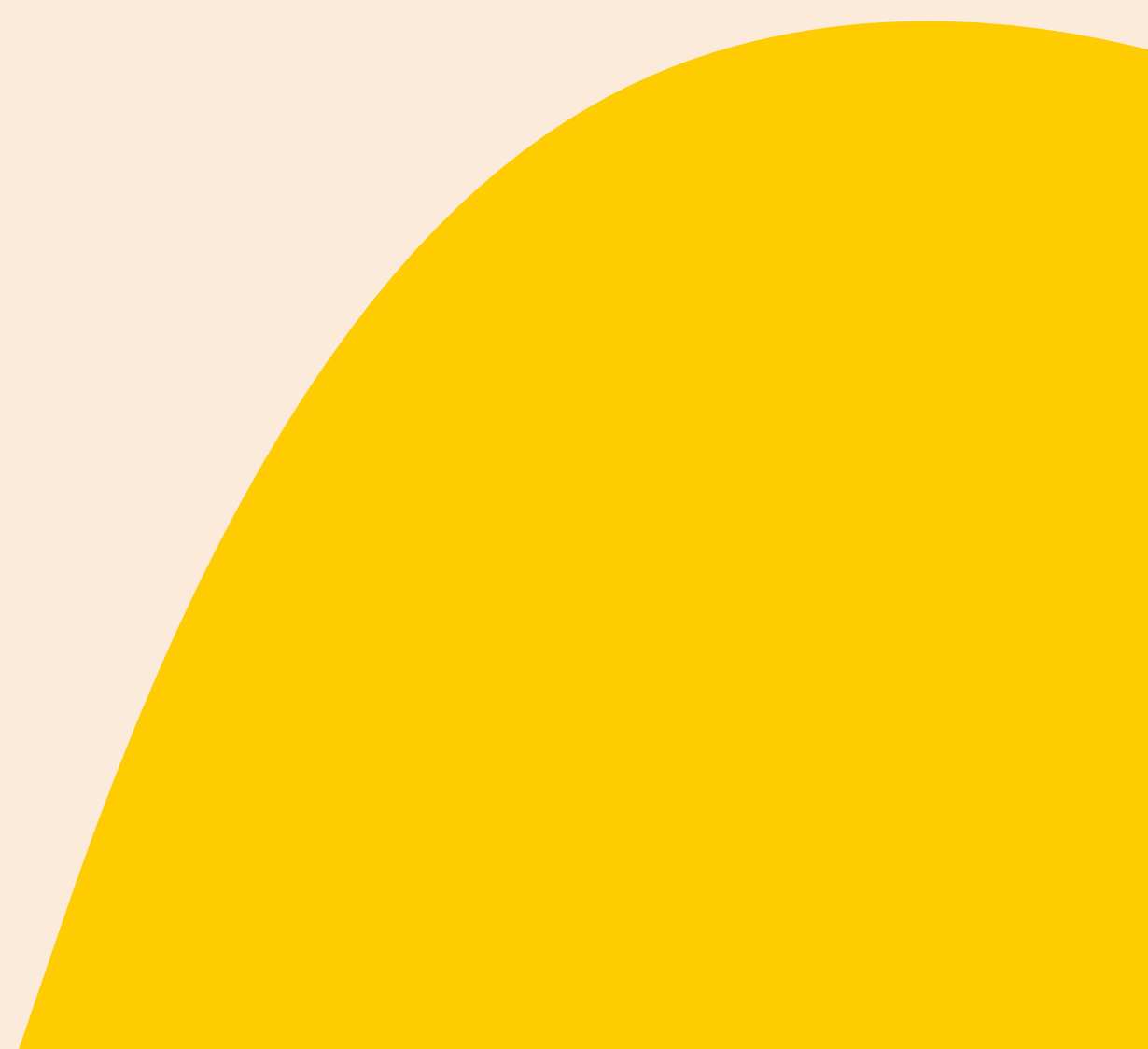
- Zoek koers euro/dollar op
- Maak een nieuw project aan in IntelliJ
- Creëer enkele variabelen binnen de *main*-methode om informatie op te slaan o
De huidige koers, het euro-bedrag en het dollar-bedrag
- Print de gegevens op het scherm met behulp van `System.out.println()`
- *Bonus*: Zorg voor gebruikersinvoer via het `Scanner`-object

Herhalingsconstructies

For, while, do-while

Herhalingsconstructies

Theorie

- Herhalingen, lussen, loops, iteraties
 - For
 - While
 - Do/While
 - *ForEach / enchanted For*
- 
- A large, solid yellow shape in the bottom right corner of the slide, resembling a quarter-circle or a stylized wave.

Herhalingsconstructies

For, while, do-while

for
loop

The Java for loop is used to iterate a part of the program several times. If the number of iteration is fixed, it is recommended to use for loop.

while
loop

The Java while loop is used to iterate a part of the program several times. If the number of iteration is not fixed, it is recommended to use while loop.

do-while
loop

The Java do-while loop is used to iterate a part of the program several times. Use it if the number of iteration is not fixed and you must have to execute the loop at least once.

For-loop

Syntax

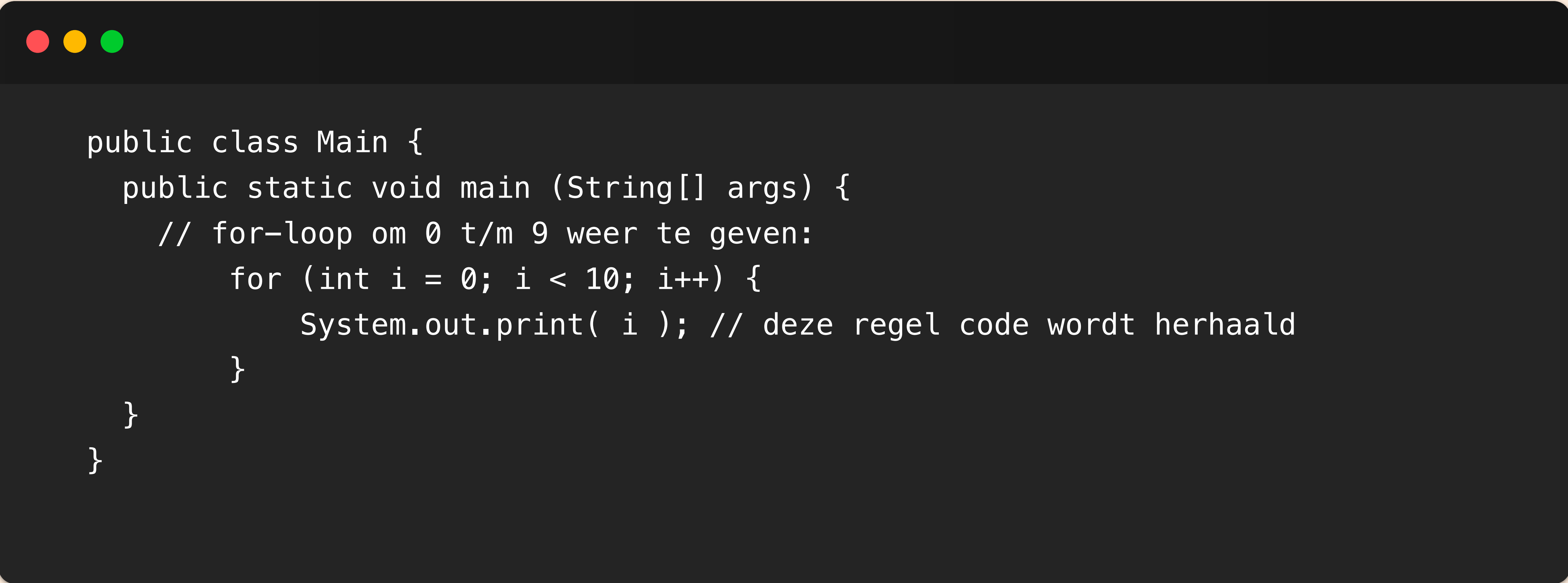
The diagram illustrates the syntax of a for-loop with three components: Declaring and Initializing loop control variable, Checking condition, and Incrementing loop control variable. These components are shown above the code snippet, with arrows pointing to their respective parts in the code.

```
for (int i =0; i<10 ; i++) {  
  
    // Loop statements to be executed  
  
}
```

Bron: <https://www.w3resource.com/java-tutorial/java-for-loop.php>

For-loop

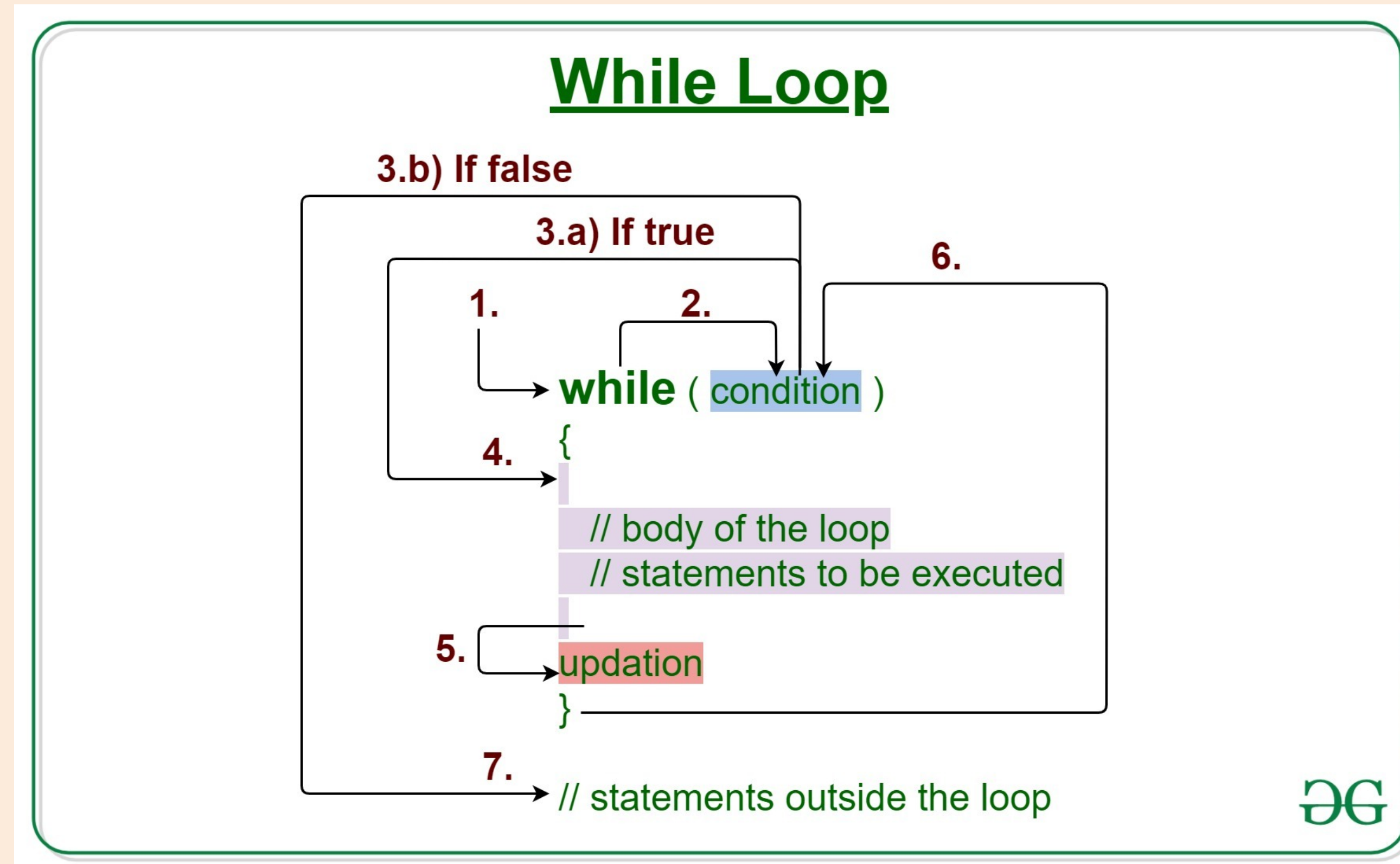
Voorbeeld-code



```
public class Main {  
    public static void main (String[] args) {  
        // for-loop om 0 t/m 9 weer te geven:  
        for (int i = 0; i < 10; i++) {  
            System.out.print( i ); // deze regel code wordt herhaald  
        }  
    }  
}
```


While-loop

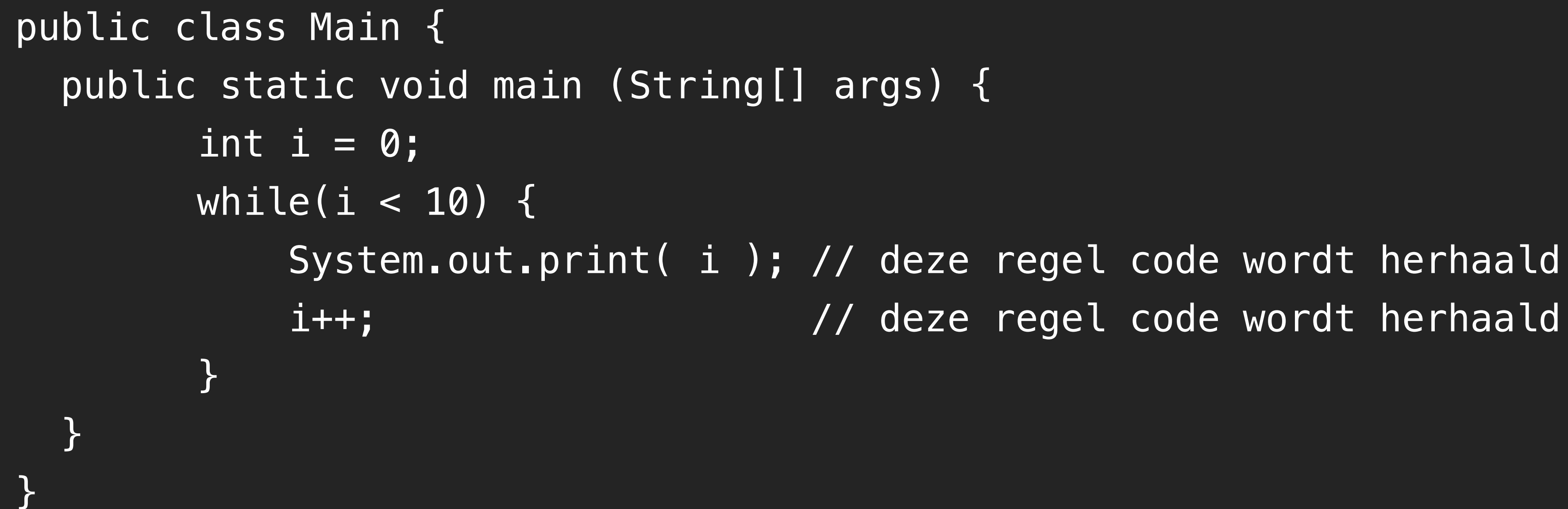
Syntax



Bron: <https://www.geeksforgeeks.org/java-while-loop-with-examples/>

While-loop

Voorbeeld-code



```
public class Main {  
    public static void main (String[] args) {  
        int i = 0;  
        while(i < 10) {  
            System.out.print( i ); // deze regel code wordt herhaald  
            i++;                    // deze regel code wordt herhaald  
        }  
    }  
}
```

Kahoot

Quiz

- Ga naar kahoot.it
- Voer Spel-PIN in
- Bekijk quiz-vragen op Teams en geef antwoord
op je eigen (mobiele) scherm

Java demonstratie

→ Herhalingsconstructies

→ for

→ while

→ Debuggen

→ Stel gerust vragen via chat

Groepsopdracht

Euro/Dollar converter

60 min

- Ga verder met de Java-code van de opdracht van gisteren. Jullie worden ingedeeld in groepjes. Een persoon deelt zijn scherm met IntelliJ. Overleg met mede-studenten over code; deel eventueel code via chat.
- Pas een **keuzeconstructie** (if/else-statement) en een **herhalingsconstructie** (bijv. while) toe zodat de invoer van de gebruiker wordt gecontroleerd (het programma vraagt net zolang om invoer totdat een goed getal is ingevoerd).
- Pas een **herhalingsconstructie** toe zodat de gebruiker vaker dan 1 keer een bedrag kan invoeren.
- *Bonus*: Zorg dat de gebruiker bij het begin kan kiezen tussen *euro-naar-dollar* of *dollar-naar-euro*.
- *Bonus 2*: Voeg meer valuta's toe: pond, yen, bitcoin, etc.

Quiz

Vragen over Java-code

→ Loops:

https://www.w3schools.com/java/exercise.asp?filename=exercise_while_loop1

→ If/else:

https://www.w3schools.com/java/exercise.asp?filename=exercise_conditions1

→ Switch:

https://www.w3schools.com/java/exercise.asp?filename=exercise_switch1

Vragen?

Loop je vast? Krijg je errors?

- Tijdens de les via de les-chat
- Na de les via Teams – Vragen/Vraagbaak-kanaal

- Plaats eventueel screenshots en/of code

Windows: Windows toets + Shift + S

Mac OS: Alt + Command + 4 (of 3)

Volgende les

Vrijdag 9.30 – 12.00 Basis Programmeren les 3

Vooruitblik

Basis Programmeren les 3 (vrijdag 9.30 – 12.00)

- Herhaling les 1 & les 2
- Verzamelingen van informatie (arrays)
- Methoden

Huiswerk

Edhub & opdrachten

- Neem **hoofdstuk 4** door van de cursus Programming Basics
- Rond de opdracht af die tijdens de les is besproken

Extra (optionele) opdrachten:

- Zie <https://www.w3resource.com/java-exercises/datatypes/index.php>

Opdracht 1 t/m 4, 9