Boolean logic

Camille Duquesne
1ère IB

Any questions/difficulties you would like to share about last session’s content ?

B2 Programming

B2.1.1 Construct and trace programs using a range of global and local variables of various data types.
- Data types: Boolean value, char, decimal, integer, string

B2.3.3 Construct programs that utilize looping structures to perform repeated actions.
- Types of loops, including counted loops and conditional loops, and appropriate use of each type
- Conditional statements within loops, using Boolean and/or relational operators to govern the loop's execution

B1.1.4 Trace flowcharts for a range of programming algorithms.
- Use of standard flowchart symbols to depict processes, decisions and flows of control
- Standard flowchart symbols: Connector, Decision, Flowline, Input/Output, Process/Operation, Start/End
- Flowcharts for execution flow, to track changes in variables and to determine output

Generative AI - Traffic Light Guide

🟢Green - For this assignment using Generative AI is authorised and encouraged to help you learn and discover new ideas. Prompts must always be cited in the following way: “Text of prompt” prompt. *ChatGPT*, Day Month version, OpenAI, Day Month Year, chat.openai.com. You should also share the discussion you had with the generative AI if you used many prompts.

🟠Orange - For this assignment using Generative AI is not recommended as it will not make you practice valuable research and thinking skills. If you are finding the assignment difficult try to use online searches instead, come to office hours, send emails to your teacher or ask your peers.

🔴Red - For this assignment using Generative AI is not allowed and goes against academic integrity rules. If Generative AI is detected you are exposing yourself to academic sanctions.

Introduction

We saw in our last class how to ask for some user input and how to perform some operations on them. But most of the time we want to perform some operations based on some conditions.

For example if we accept the cookies, turn targeted advertisement on, else deny them.

Relational operators

A relational operator compares two values and determines the relationship between them. These operators will only work for primitive data types representing numbers

Operator	Description	Example
`==`	Checks if the values of two operands are **equal** or not	`(1 == 2)` is not true.
`!=`	Checks if the values of two operands are **not equal**	`(1 != 2)` is true.
`>`	Checks if the value of left operand is **greater than** the value of right operand.	`(1 > 2)` is not true.
`<`	Checks if the value of left operand is **less than** the value of right operand	`(1 < 2)` is true.
`>=`	Checks if the value of left operand is **greater than or equal** to the value of right operand	`(1 >= 2)` is not true.
`<=`	Checks if the value of left operand is **less than or equal** to the value of right operand	`(1 <= 1)` is true.

Boolean operators

If we want to combine relational operations together we can use the following boolean operators:

AND

A	B	A && B
true	true	true
true	false	false
false	true	false
false	false	false

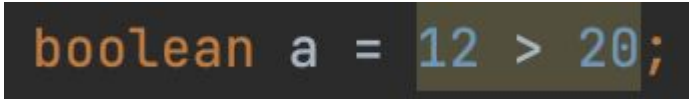
OR

A	B	A || B
true	true	true
true	false	true
false	true	true
false	false	false

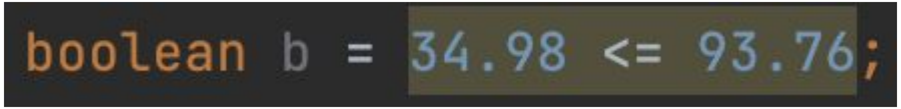
NOT

A	!A
true	false
false	true

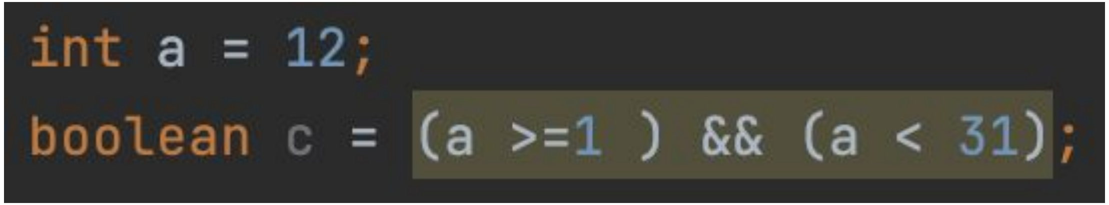
Boolean expression examples



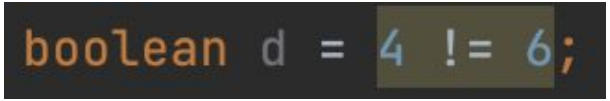
```
boolean a = 12 > 20;
```



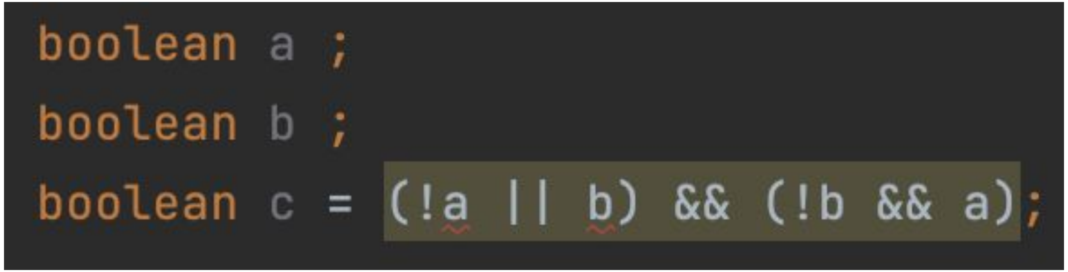
```
boolean b = 34.98 <= 93.76;
```



```
int a = 12;
boolean c = (a >=1 ) && (a < 31);
```

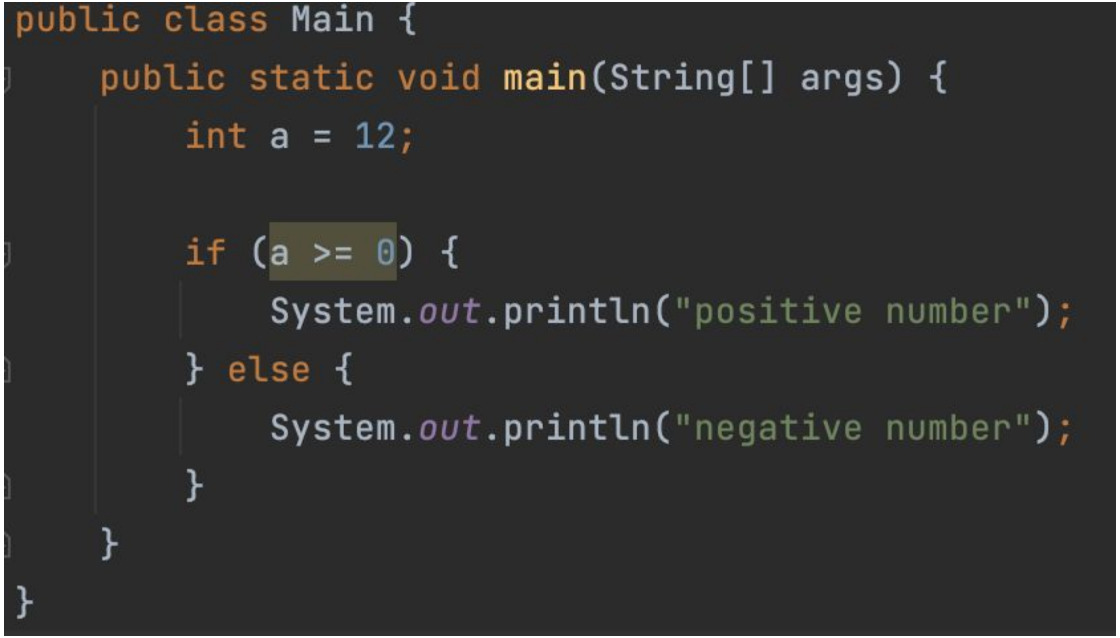


```
boolean d = 4 != 6;
```



```
boolean a ;
boolean b ;
boolean c = (!a || b) && (!b && a);
```

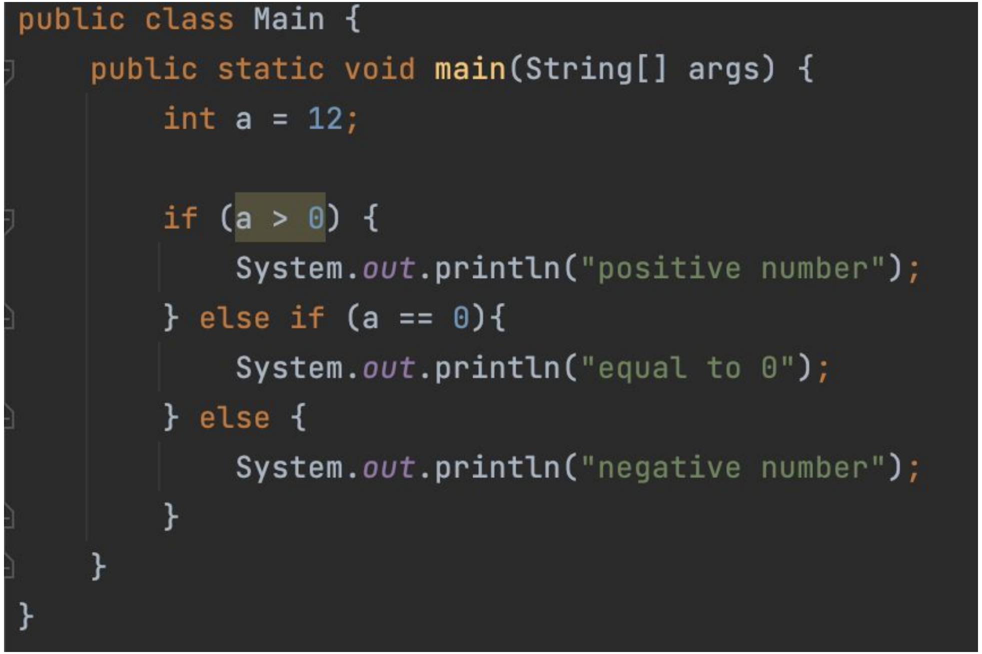
If (condition) {} else {}



```
public class Main {
    public static void main(String[] args) {
        int a = 12;

        if (a >= 0) {
            System.out.println("positive number");
        } else {
            System.out.println("negative number");
        }
    }
}
```

- If the first condition (always in parentheses) is true then the first block is executed
- Else (= if the first condition is false) the second block is executed
- Blocks are defined by curly braces `{}`
- The `if` is always first and then the `else`
- The else clause is optional

If (condition) {} else if (condition) {} else {}

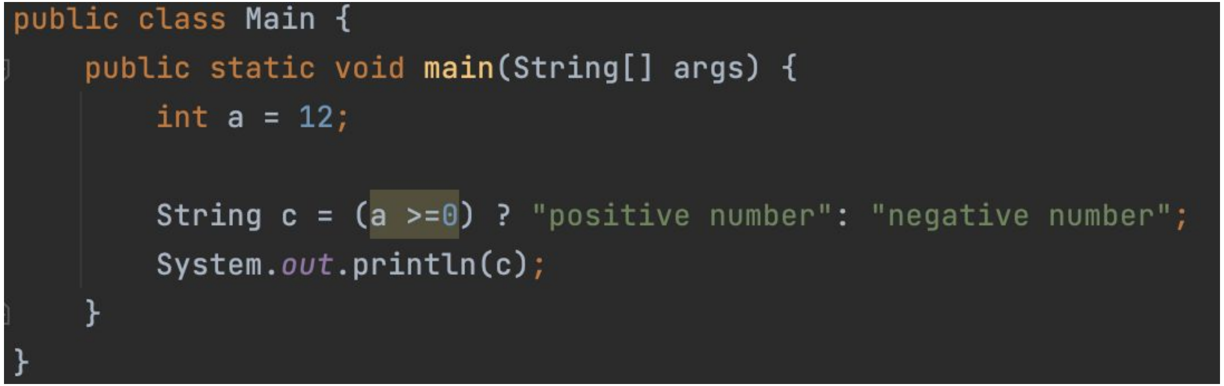


```java
public class Main {
    public static void main(String[] args) {
        int a = 12;

        if (a > 0) {
            System.out.println("positive number");
        } else if (a == 0){
            System.out.println("equal to 0");
        } else {
            System.out.println("negative number");
        }
    }
}
```

- If the first condition is true then the first block is executed
- If the first condition is false then the second condition is evaluated.
- If the second condition is true then the second block is executed
- If the second condition is false then the third block is executed.
- `if`, `else if`, `else` are always in that order
- There can be as many `else if` conditions as we want

Ternary operator

Java has one ternay operator, which acts the same way as an if else statement.



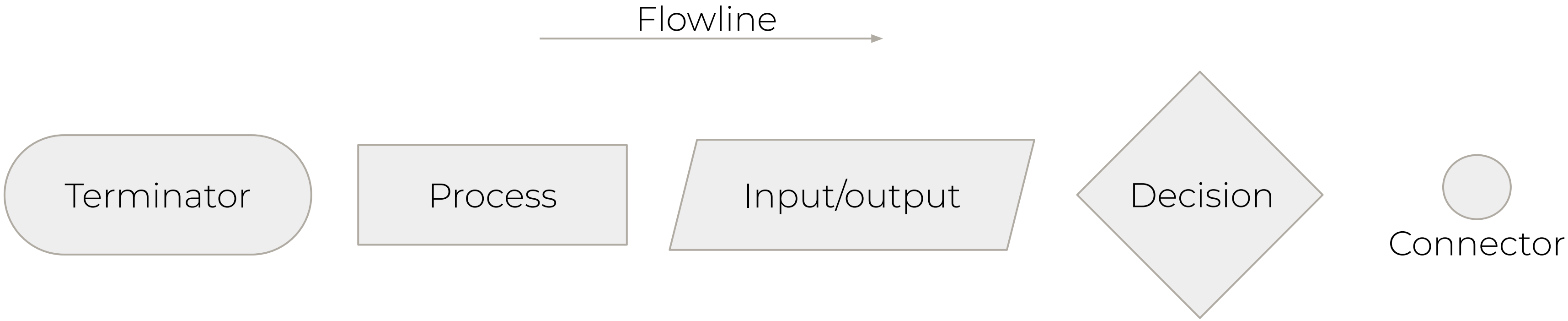
```
public class Main {
    public static void main(String[] args) {
        int a = 12;

        String c = (a >=0) ? "positive number": "negative number";
        System.out.println(c);
    }
}
```

What does ternay mean ? Do unary operators exist ?

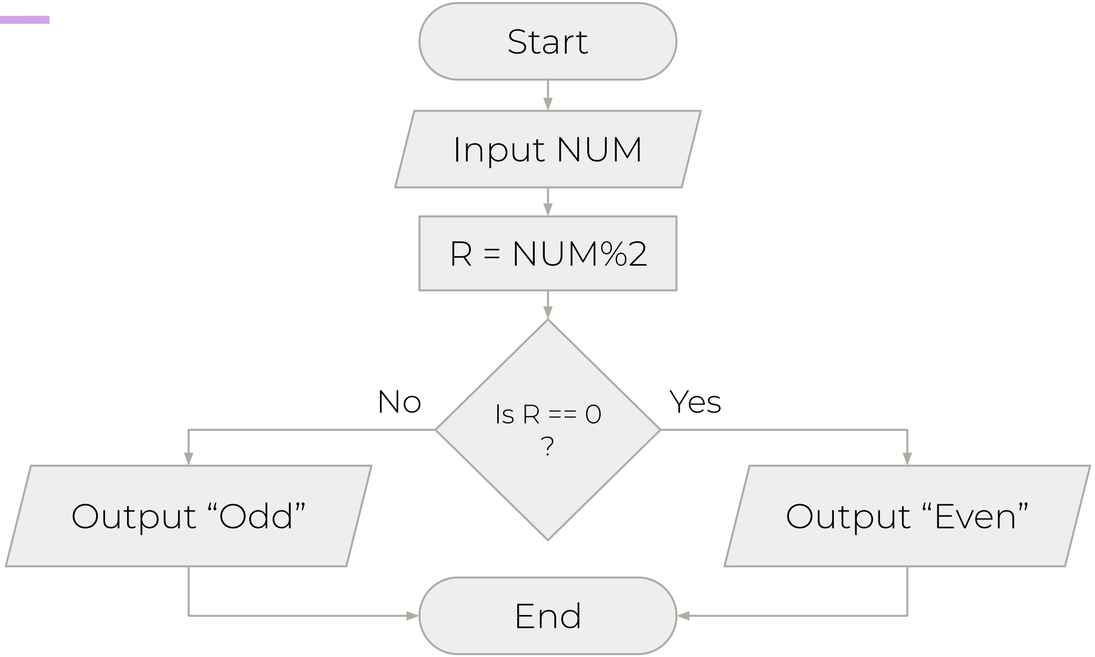
What are flowcharts ?

Flowcharts are a visual representation of the flow of execution of a program/an algorithm.

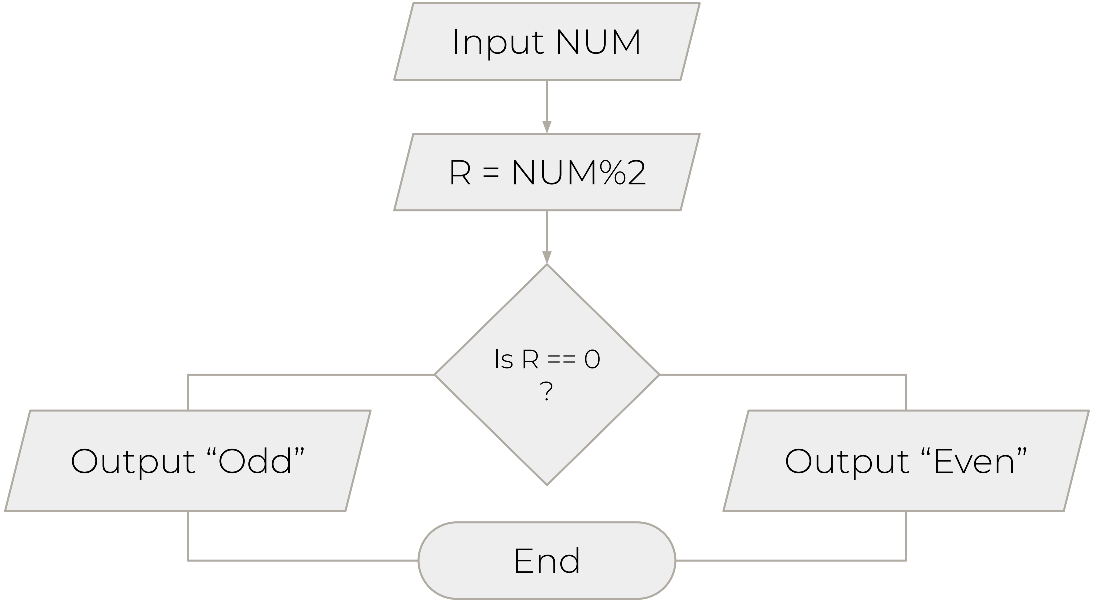
Flowchart symbols are standardised and are connected together using flowlines



Flowchart example - checking if a number is even or odd



Flowchart example - Spot the 4 errors !



Input NUM

R = NUM%2

Is R == 0
?

Output “Odd”

Output “Even”

End

What do you think ?

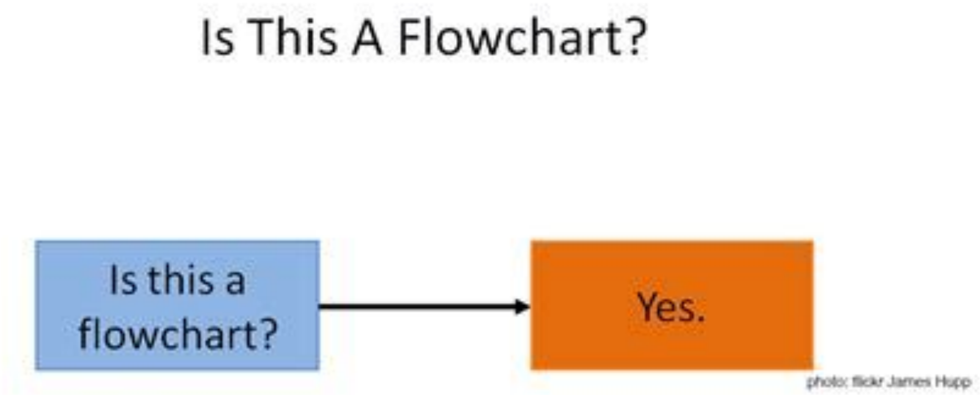
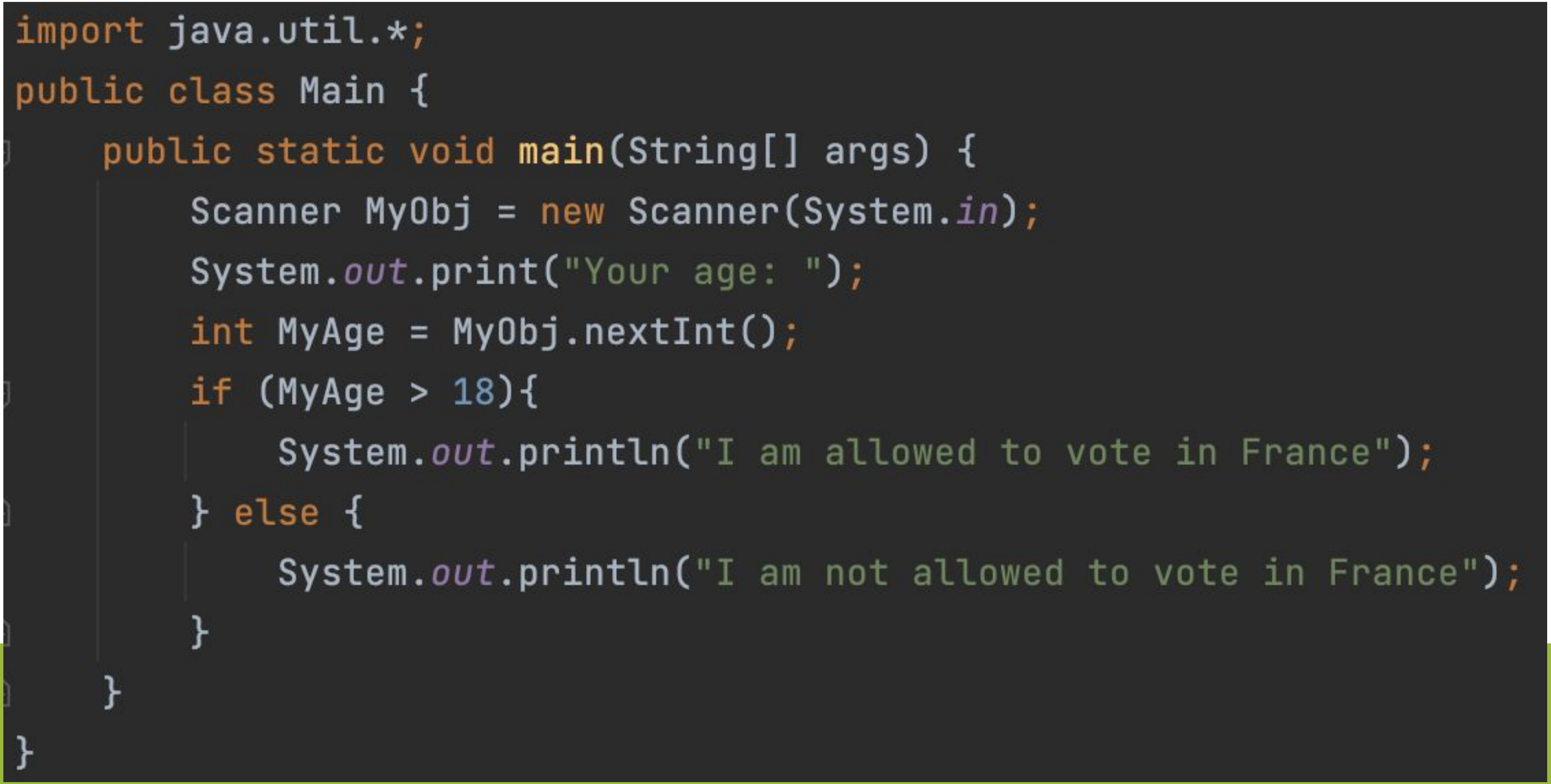


photo: flickr James Hupp

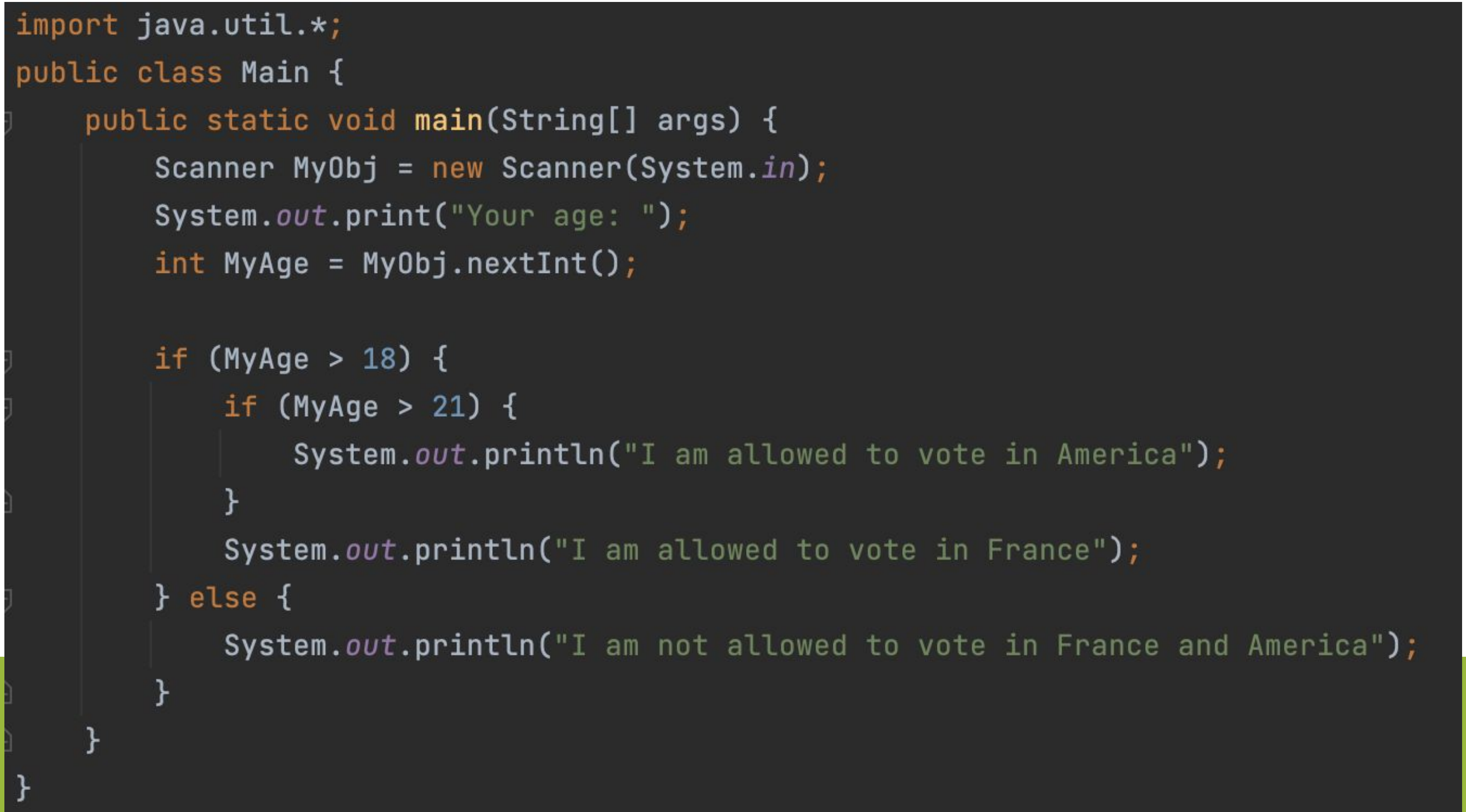
https://nulab.com/learn/design-and-ux/20-funny-flowcharts-help-navigate-lifes-toughest-decisions/

What does this code output ?



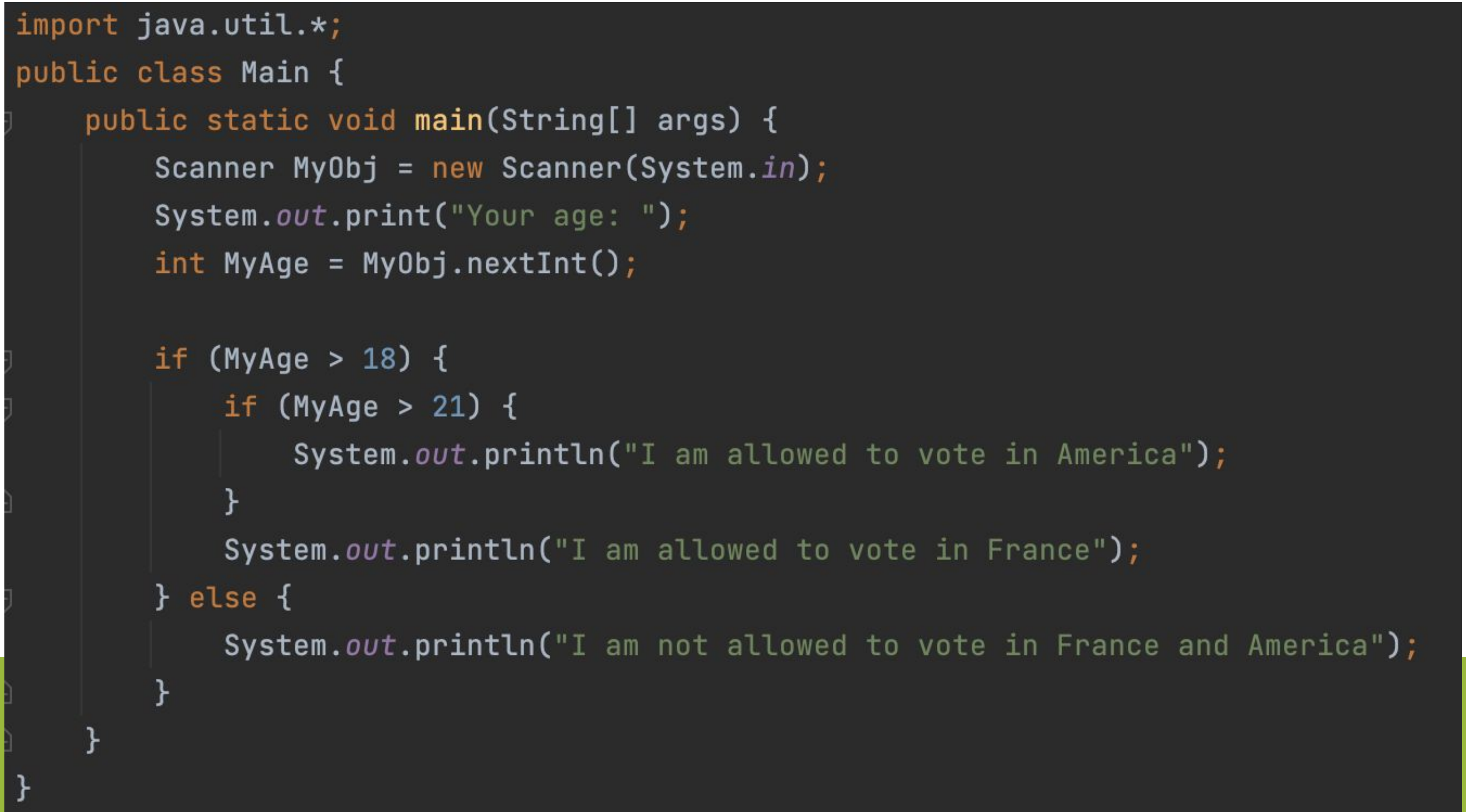
```java
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner MyObj = new Scanner(System.in);
        System.out.print("Your age: ");
        int MyAge = MyObj.nextInt();
        if (MyAge > 18){
            System.out.println("I am allowed to vote in France");
        } else {
            System.out.println("I am not allowed to vote in France");
        }
    }
}
```

What does this code output ?



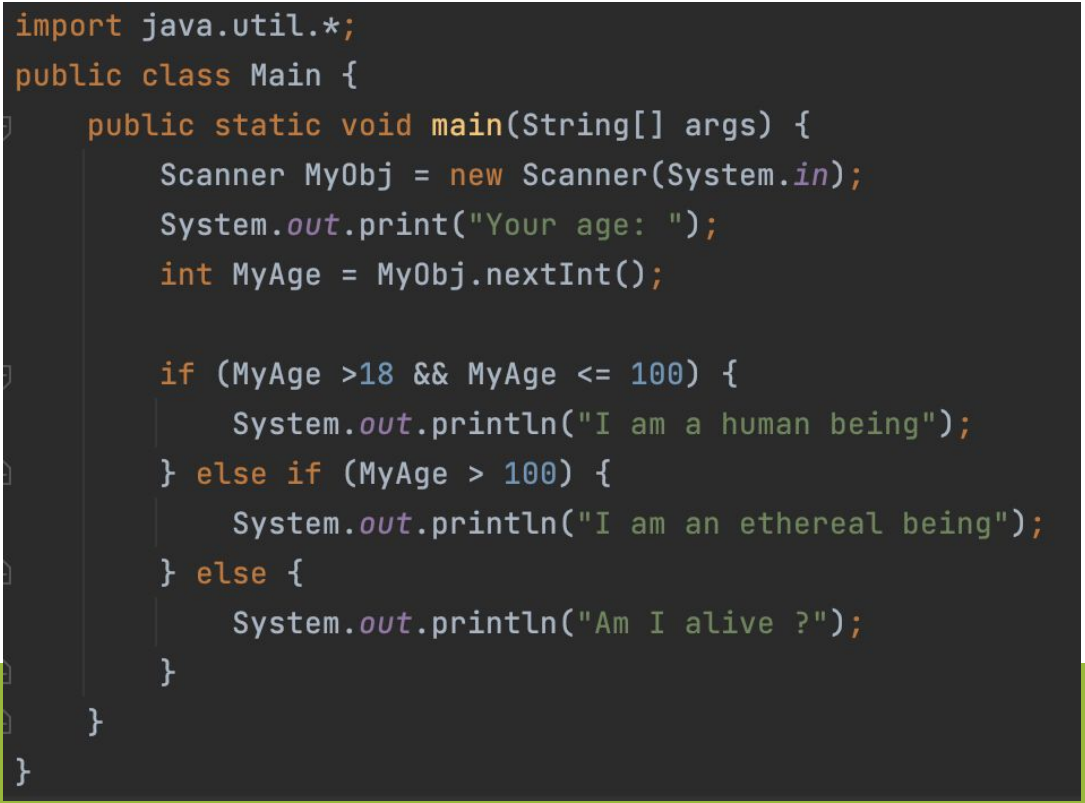
```java
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner MyObj = new Scanner(System.in);
        System.out.print("Your age: ");
        int MyAge = MyObj.nextInt();

        if (MyAge > 18) {
            if (MyAge > 21) {
                System.out.println("I am allowed to vote in America");
            }
            System.out.println("I am allowed to vote in France");
        } else {
            System.out.println("I am not allowed to vote in France and America");
        }
    }
}
```

How would the flowchart for the following process look like ?



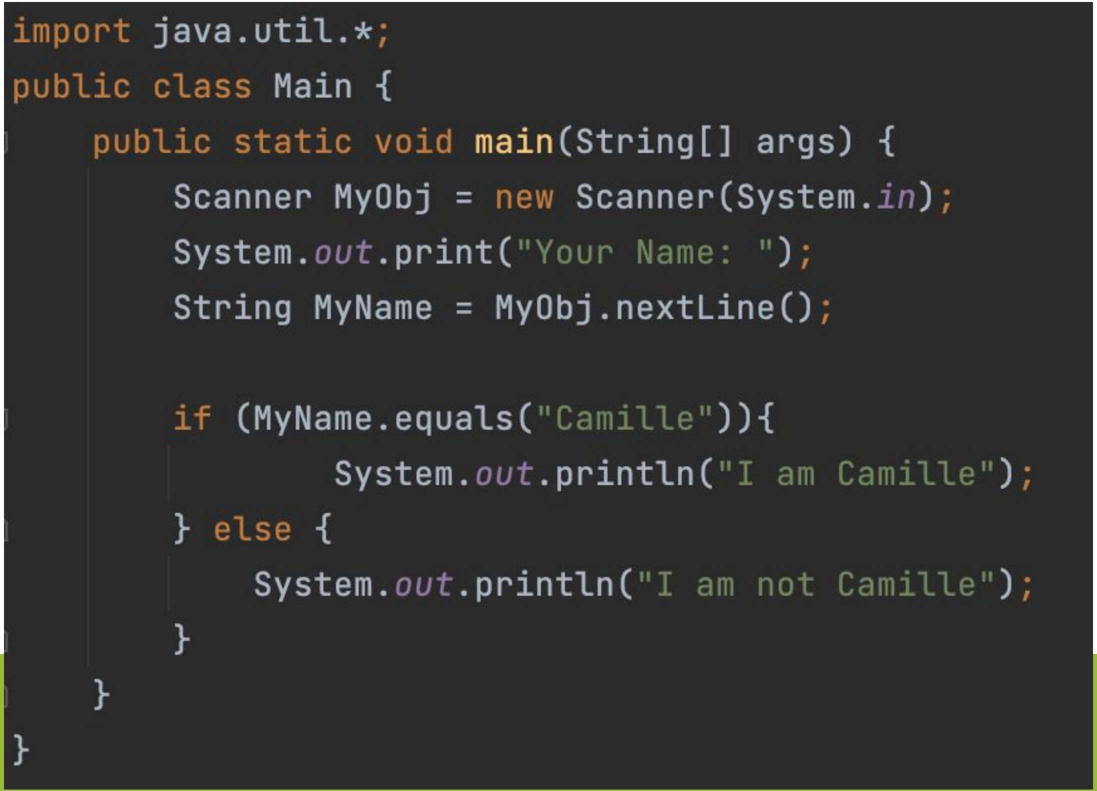
```java
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner MyObj = new Scanner(System.in);
        System.out.print("Your age: ");
        int MyAge = MyObj.nextInt();

        if (MyAge > 18) {
            if (MyAge > 21) {
                System.out.println("I am allowed to vote in America");
            }
            System.out.println("I am allowed to vote in France");
        } else {
            System.out.println("I am not allowed to vote in France and America");
        }
    }
}
```

What does this code output ?



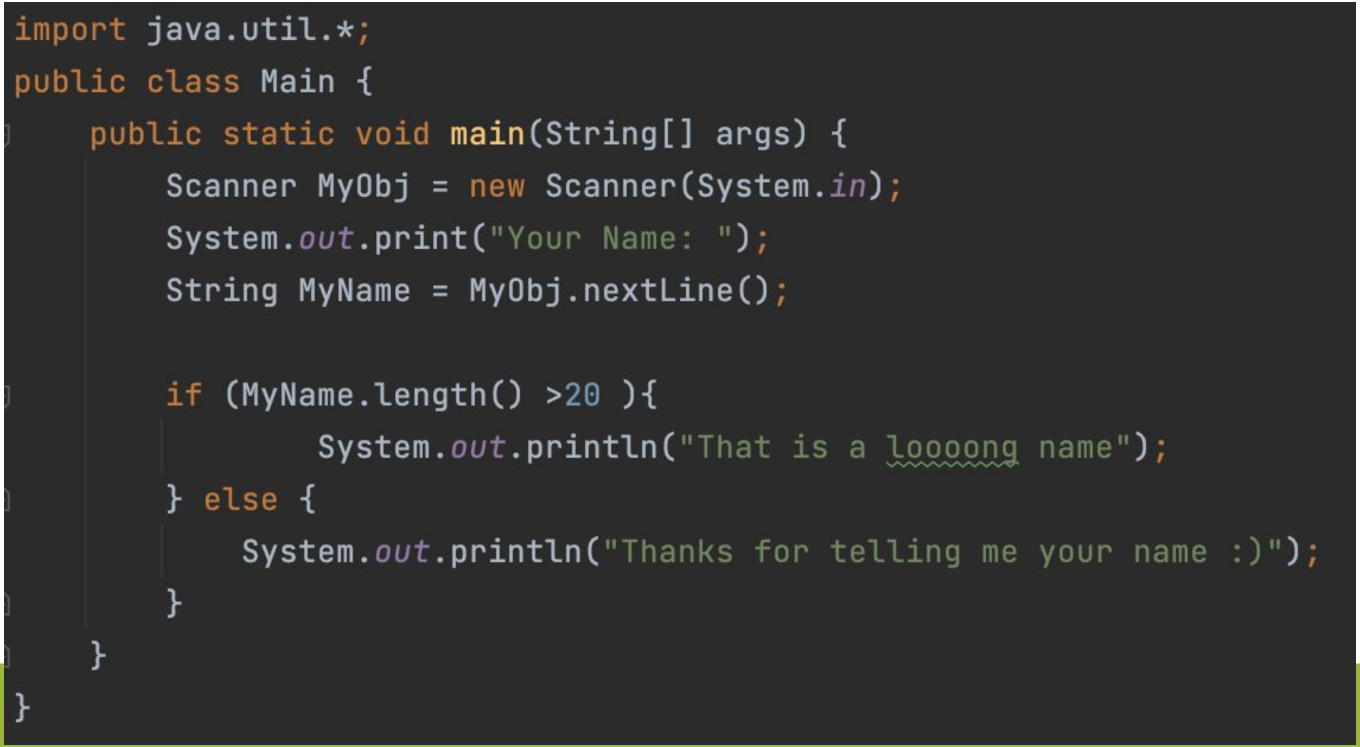
```java
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner MyObj = new Scanner(System.in);
        System.out.print("Your age: ");
        int MyAge = MyObj.nextInt();

        if (MyAge >18 && MyAge <= 100) {
            System.out.println("I am a human being");
        } else if (MyAge > 100) {
            System.out.println("I am an ethereal being");
        } else {
            System.out.println("Am I alive ?");
        }
    }
}
```

What does this code output ?



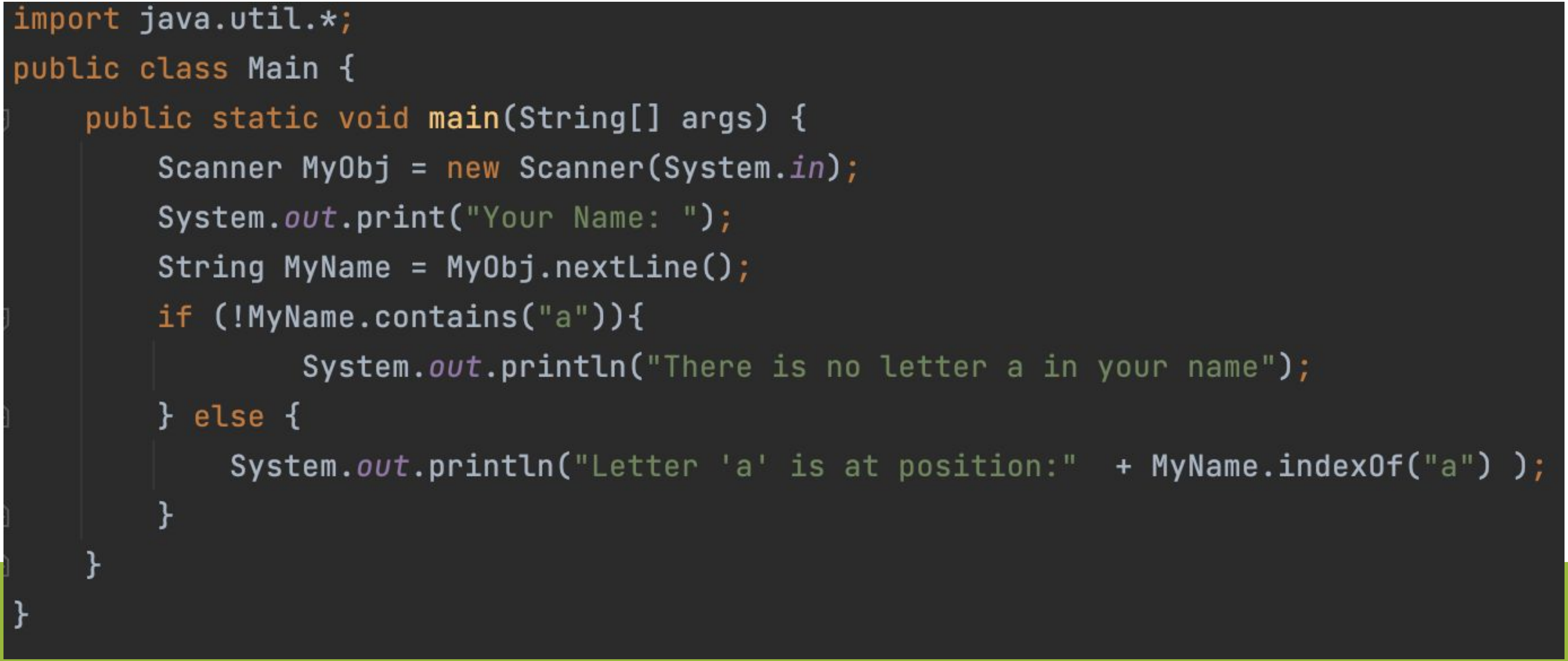
```java
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner MyObj = new Scanner(System.in);
        System.out.print("Your Name: ");
        String MyName = MyObj.nextLine();

        if (MyName.equals("Camille")){
                System.out.println("I am Camille");
        } else {
            System.out.println("I am not Camille");
        }
    }
}
```

What does this code output ?



```
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner MyObj = new Scanner(System.in);
        System.out.print("Your Name: ");
        String MyName = MyObj.nextLine();

        if (MyName.length() >20 ){
                System.out.println("That is a loooong name");
        } else {
            System.out.println("Thanks for telling me your name :)");
        }
    }
}
```

What does this code output ?



```java
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner MyObj = new Scanner(System.in);
        System.out.print("Your Name: ");
        String MyName = MyObj.nextLine();
        if (!MyName.contains("a")){
                System.out.println("There is no letter a in your name");
        } else {
            System.out.println("Letter 'a' is at position:"  + MyName.indexOf("a") );
        }
    }
}
```

Pause & Recall

Created by Victoruler
from Noun Project

Close your eyes and try to recall as many things as possible that were covered during this lesson.

Alternatively, you can keep your eyes open and write down as many things you remember on a piece of paper.

This will help strengthen your memory of key concepts 💪

Exercise 1 *(GenAI 🟠 Orange)*

Write a program where the user inputs a number and the program outputs a string saying if the number is even or odd.

Write the code a first time with an if else statement and then a second time with the ternary operator.

Exercise 2 *(GenAI 🟠 Orange)*

Draw a flowchart that describes the process to check if a year, that the user entered, is a leap year or not.

https://www.mathsisfun.com/leap-years.html

Then code the program and check if it works using different test cases.

Exercise 3 *(GenAI 🟠 Orange)*

We want to buy train tickets for our next vacation. The train company applies a discount of 10% if the ticket price is above or equal to 100€. If we have a membership card we get 20% discount instead of 10%. If the price is below 100€, no discount is applied.

Draw a flowchart that illustrates the process to calculate the price to pay given the ticket price and membership of the traveler.

Then code the program and check if it works using different test cases.

Exercise 4 *(GenAI 🟠 Orange)*

Write a java program that takes three numbers from the user and prints the greatest number of those three.

Exercise 5 *(GenAI 🟠 Orange)*

Upgrade your program from exercise 2 to check if the user input are only numeric characters. This will make sure that the rest of the statements are executed safely and that there will be no errors.

Exercise 6 *(GenAI 🟠Orange/🟢Green)*

Write a script that validates whether the given password from the user is valid or not with the following conditions:

- the password must contain a capital letter
- the password must contain at least one lower case letter
- the password must contain at least one number
- the password must be longer than 8 characters

Exercise 7 *(GenAI 🟠 Orange)*

Hide your java solutions to your previous exercises. Try to solve them again in a flowchart form.

Exercise 8 *(GenAI 🟠 Orange/🟢 Green)*

Compare your answers to previous exercises with someone's else's answers.

What are similarities/differences betweens your codes ?

Which one is more readable/understandable faster ? Why ?

Translate your classmate's code into flowcharts and vice versa

Homework

Finish all the exercises

Create flashcards for the following terms:

Operator
Else
Else if
Else
Flowchart
And
Or
Not
Precedence of operators
Input
Output
Process