HAMK
Häme University of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin osarahoittama

Workshop 1: Python Programming and Data Analysis

The workshop is part of EU co-funded Robo CO. - Robots and AI as future colleagues –project

HAMK
Häme University of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin osarahoittama

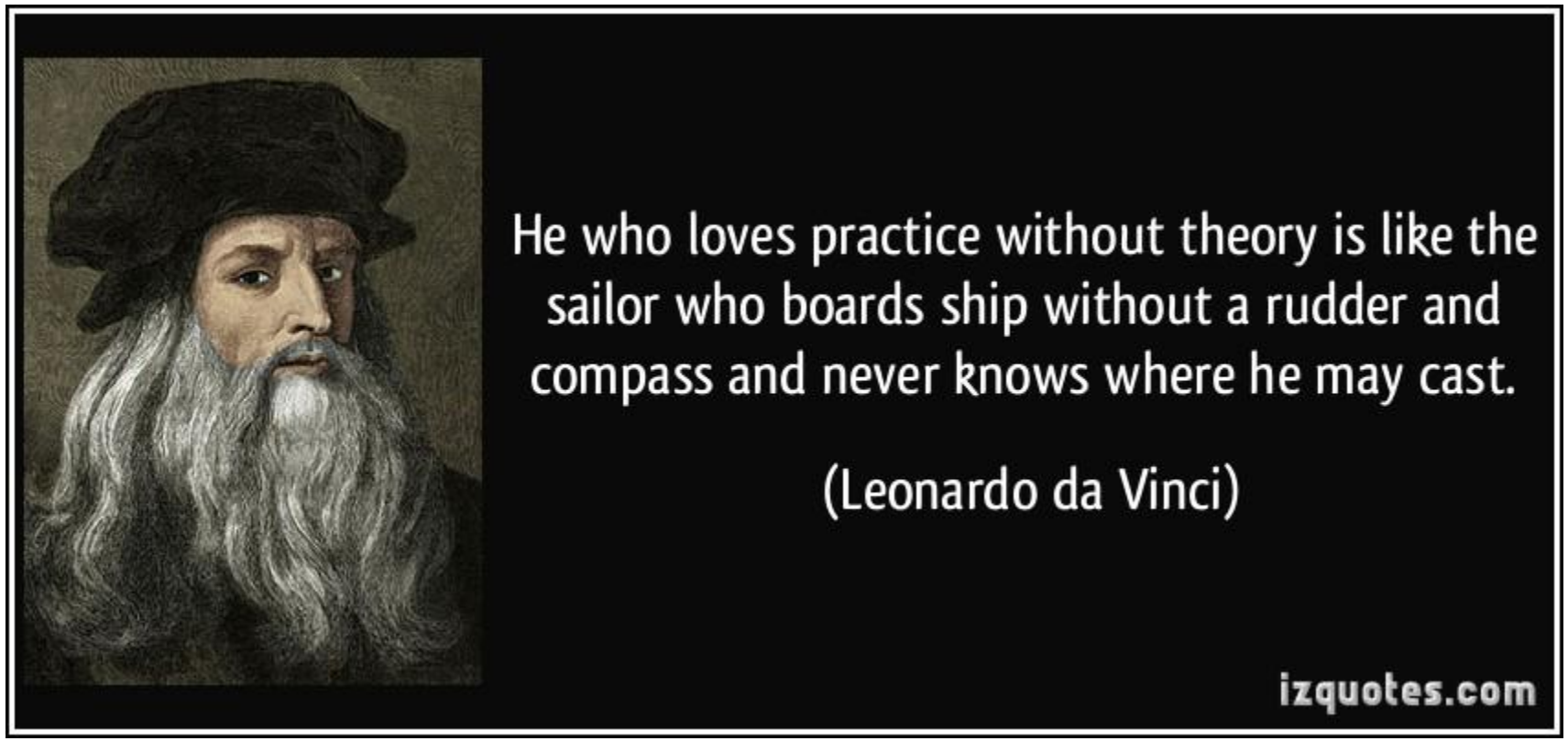
Anastasiia Stepanova
Project engineer
HAMK Tech Robotics
anastasiia.stepanova@hamk.fi

HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Small intro



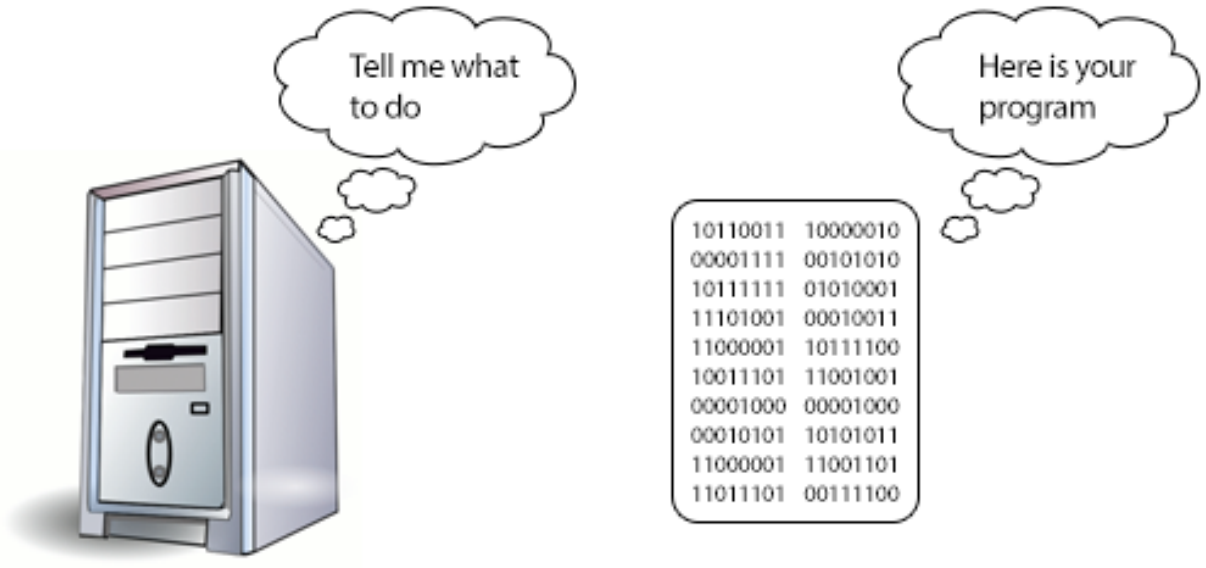
HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Programming and Program

- **Programming**: designing a program to be executed by a device.
- A computer **program:** computing task with a set of instructions that can be executed by a computer. The instructions are usually commands for the CPU of the computer.



High-level languages

- High-level languages resemble natural languages, such as English or Finnish, but they have strict grammar rules and limited vocabulary to facilitate the automatic transformation of the program into machine language with a compiler or an interpreter.
- The ambiguity of words and sentences as well as the complexity of grammar are only some of the reasons that have so far prevented computer programming in natural languages. There are tens of well-known high-level languages (**C, C++, Java and Python** are especially well-established ones). The term high-level language is vague: Languages such as C are clearly more computer-oriented than languages such as Python and Java, which offer higher levels of abstraction.

```
print("Welcome to the Python Basics Workshop!")
```

HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

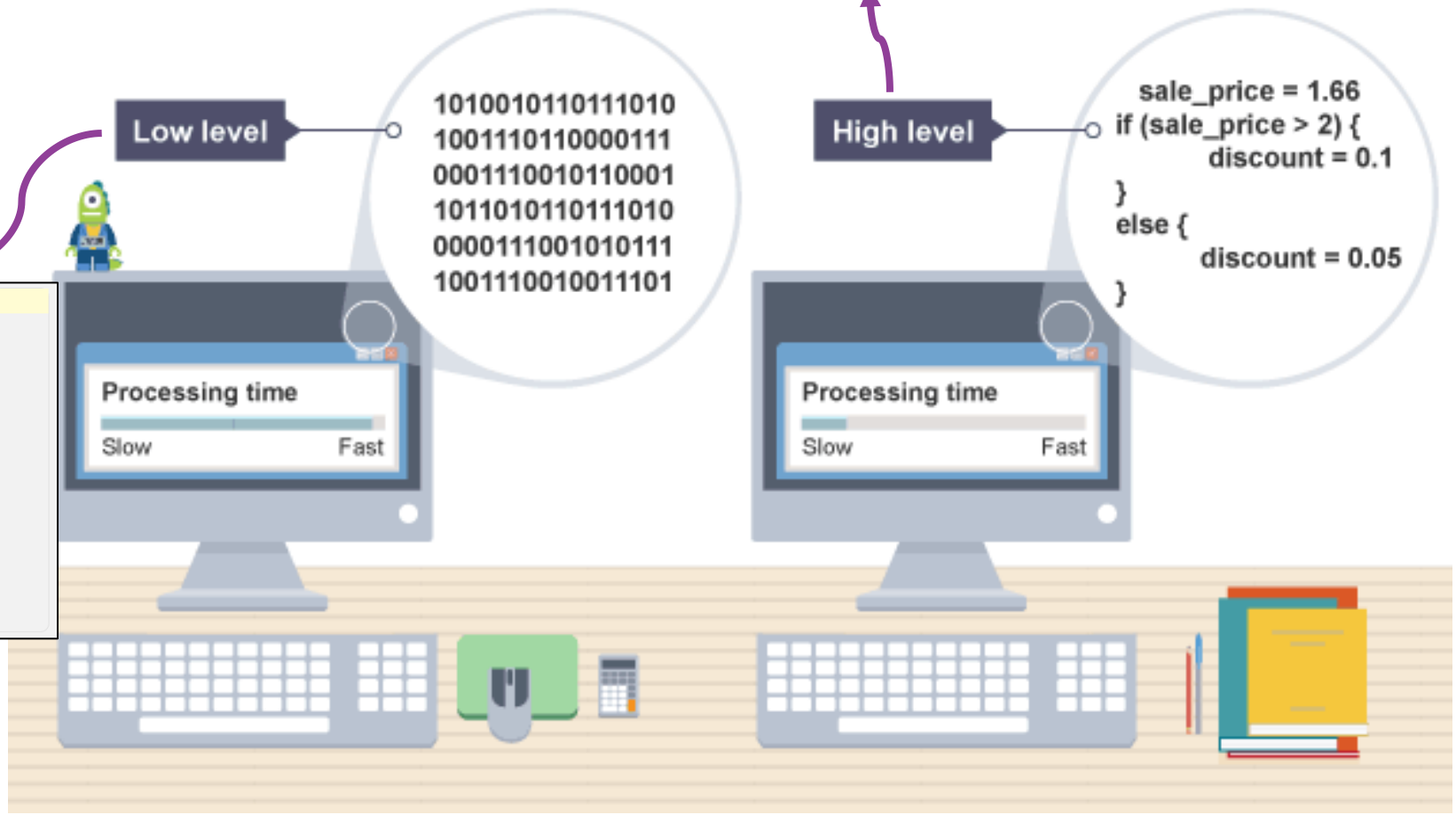
Low-level vs High-level

python

```python
print("Hello, World!")
```

assembler

```
.MODEL SMALL
.STACK 100h
.DATA
    HelloMessage DB 'Hello, World!',13,10,'$'
.CODE
START:
    mov ax,@data
    mov ds,ax
    mov ah,9
    mov dx,OFFSET HelloMessage
    int 21h
    mov ah,4ch
    int 21h
END START
```



Compiled vs interpreted languages

- **Compiled**: e.g. C, C++
  - **Translates entire source code into machine code before execution.**
  - **Faster** execution once compiled (machine code directly executed by CPU).
  - Compilation can take significant time, but this is a one-time cost.
  - Detects syntax and semantic errors during the compilation phase and all errors are reported together after the entire code is analyzed.
- **Interpreted**: e.g. Python, JavaScript
  - **Translates and executes source code line-by-line or statement-by-statement at runtime.**
  - Generally **slower** execution time **because translation occurs concurrently with execution.**
  - **Immediate execution** without the need for a separate compilation step.
  - Detects errors at runtime. Stops execution when it encounters an error and reports it immediately.
  - Allows for quicker debugging in some scenarios because you can test parts of the code interactively.

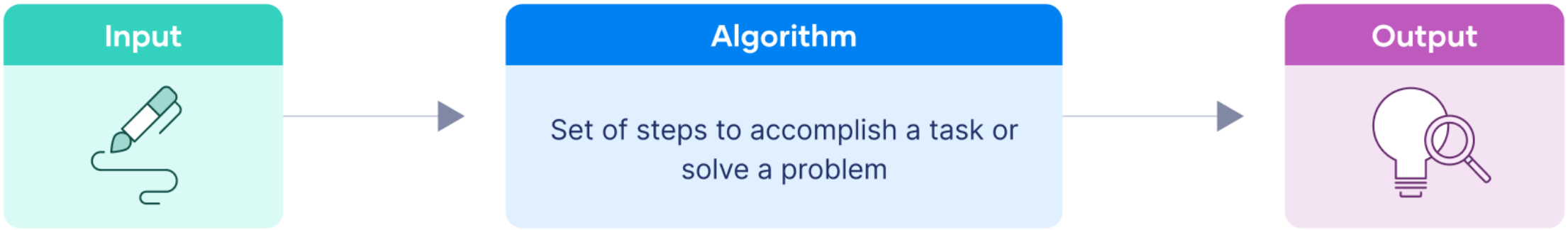
HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Algorithm

- Algorithm is a stepwise representation to solve a problem in a finite time.
- Algorithm is a wider concept than a program although these terms are often used interchangeably.
- Every stepwise task can be formulated as an algorithm, but only a part of the algorithms can be implemented as a program.



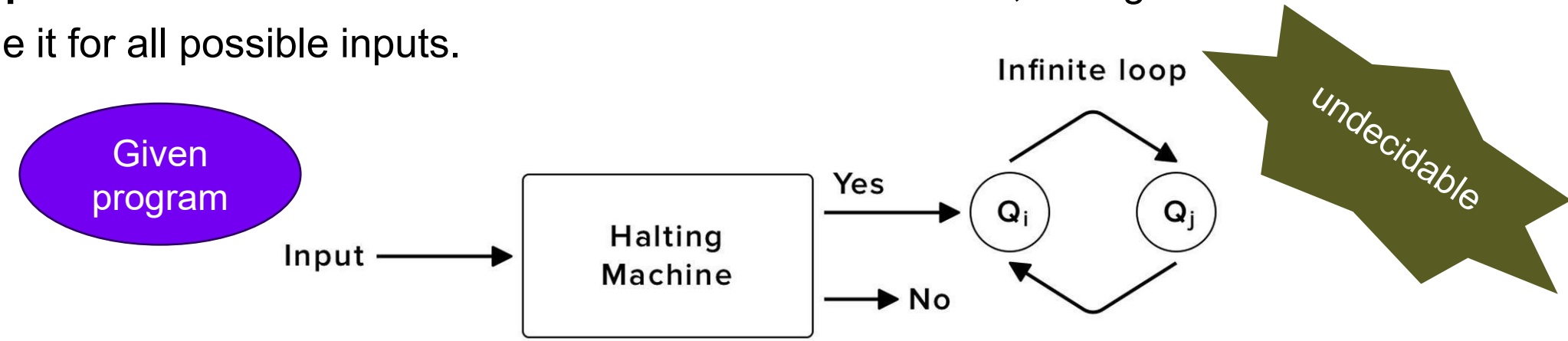
HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

What algorithm can not be implemented as a program?

- Tasks described by an algorithm may be too **vague** to be translated into a program
- **Resource constraints:** memory, processing power.
- **Too complex:** inefficient for practical implementation.
- **Non-computable:** cannot be solved within a finite amount of time, no algorithm can decide it for all possible inputs.



HAMK
Häme University
of Applied Sciences

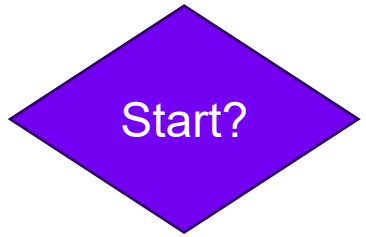
REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Flowchart

- Flowchart is a **graphical language** to formalize algorithms.
- It consist of graphical symbols connected with arrows.
- Rectangles denote actions.

- Rhombus (diamonds) represent decisions.
  Decisions allow the branching off to mutually exclusive paths.



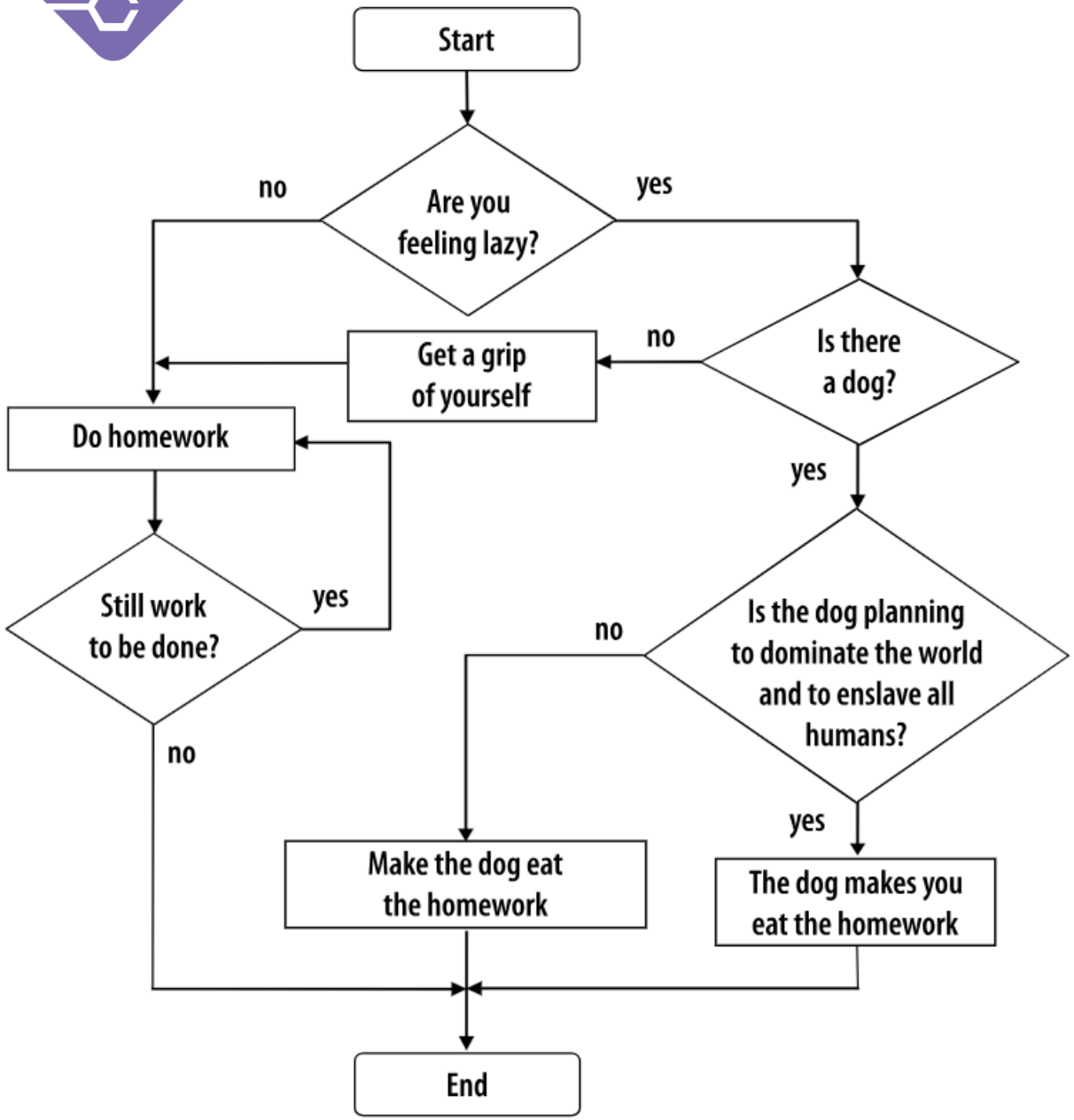
- A flowchart is executed from the start symbol and following the arrows until the end symbol is reached.

HAMK
Häme University of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin osarahoittama

Avoiding homework flowchart



More about algorithms and programs

- Only **exact enough** algorithms can be implemented as a program.
- There are several problems in this sense with the previous example:
- First, the execution of this flowchart requires common sense, while a program must be executed automatically with any knowledge or understanding of the surrounding world.
- Second, the actions and decisions of the algorithm are too ambiguous. For example, action "Do homework" is too vague to be expressed in a programming language.
- Lastly, the flowchart should be so comprehensive that even remotely possible cases have been considered. The previous example does not take into account that the homework may be already ready.

Crossing language barrier

- **Tools are needed to automatically translate source code, that is, a program written in a high-level language, into machine language.**
- A **compiler** translates the source code into machine language specific for the computer and saves the machine language as an executable file, while an **interpreter** sends the machine language directly to central processing unit for execution.
- Interpreted programs are usually slower than compiled ones, because the translation needs to be done during the execution. On the other hand, the interpreted programs are transferrable: they can be executed in any environment that has the interpreter. For example, C and C++ programs are compiled, while **Python is an interpreted language**. Java programs are both compiled and interpreted. Successful translation process requires that the program is strictly grammatically correct.

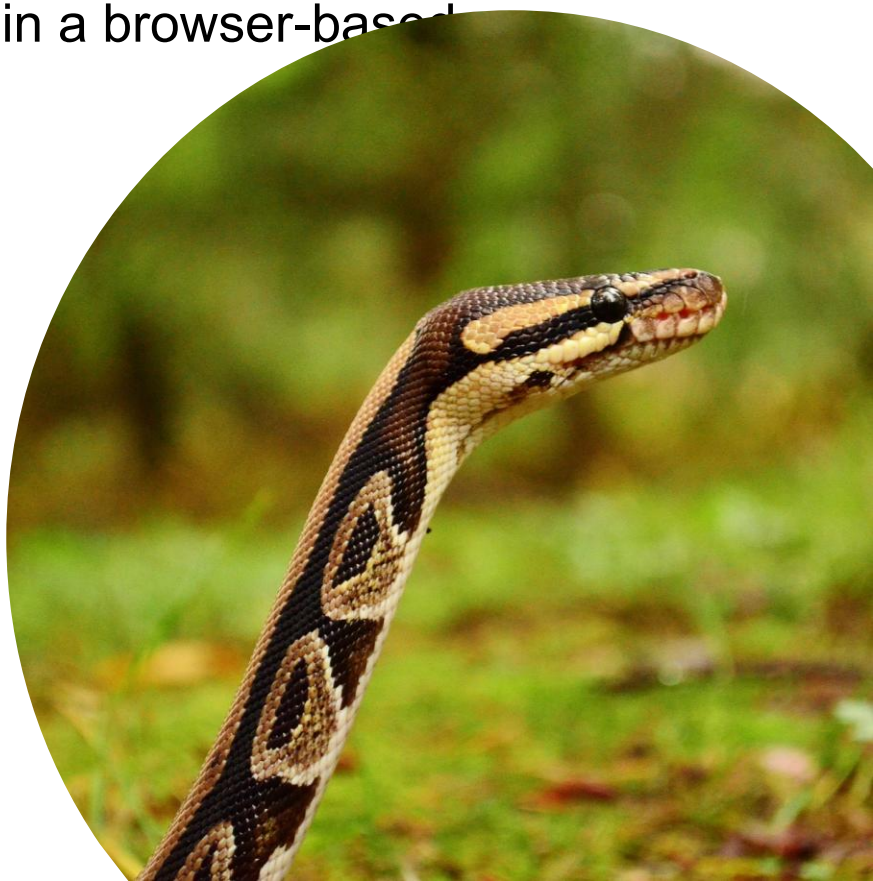
HAMK
Häme University of Applied Sciences

RUN REGIONAL UNIVERSITY NETWORK EUROPEAN UNIVERSITY

Euroopan unionin osarahoittama

Environment for this course

- **Google Colaboratory Notebook**: cloud-based user-friendly platform provided by Google that allows users to write and execute Python code in a browser-based interactive environment.
  - Free to use
  - Integrated with Google Drive
  - High-performance computing
  - Interactive visualization
  - Sharing and collaboration



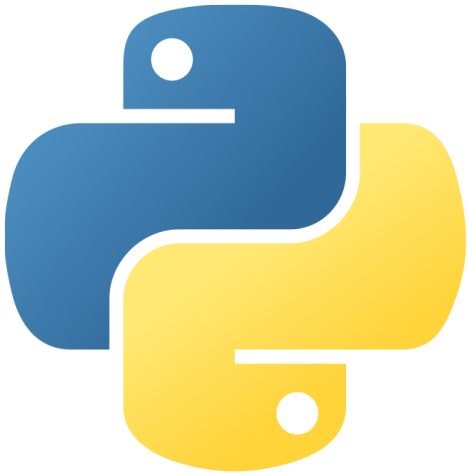
HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

What is Python?

- Python is a **high-level, interpreted programming language** known for its simplicity and readability.
- It was created by Guido van Rossum and first released in 1991.
- Python is widely used in various domains, including web development, data science, machine learning, artificial intelligence, scientific computing, automation, and more.



HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Why Python?

- **Easy to learn and read:** Python syntax is clean and easy to understand, making it ideal for beginners.
- **Versatile:** Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming.
- **Large standard library:** Python comes with a rich set of libraries and modules for performing various tasks, **reducing the need to write code from scratch.**
- **Community support:** Python has a large and active community of developers who contribute to its ecosystem, providing libraries, frameworks, and resources.
- **Cross-platform:** Python runs on all major operating systems (Windows, macOS, Linux), making it highly portable.

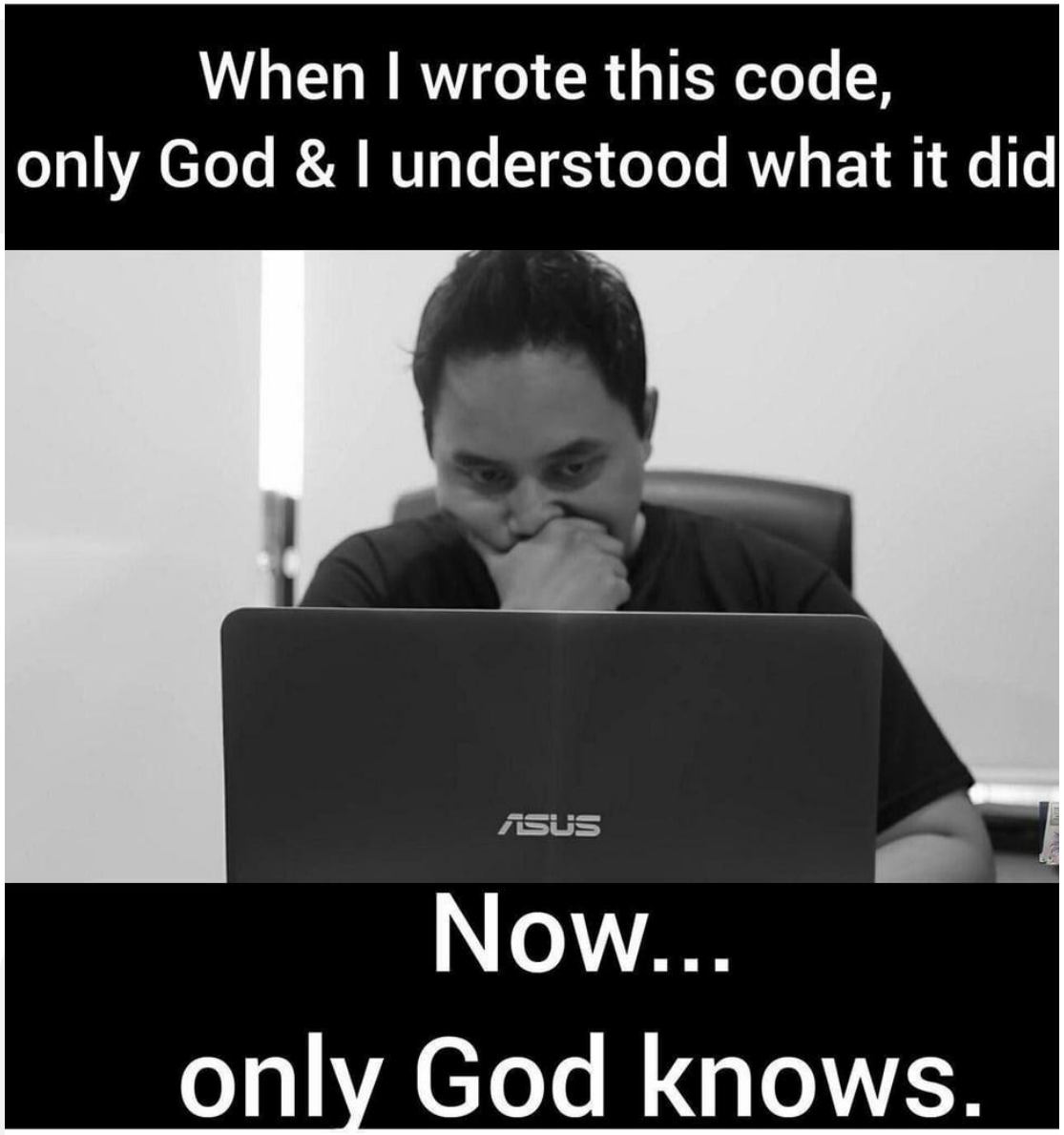
HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Basics of Python

- The provided code tutorial covers basic Python concepts such as variables, data types, arithmetic operations, conditional statements, loops, lists, functions, and even classes.
- You will run the code cell by cell to see the output and learn from it.



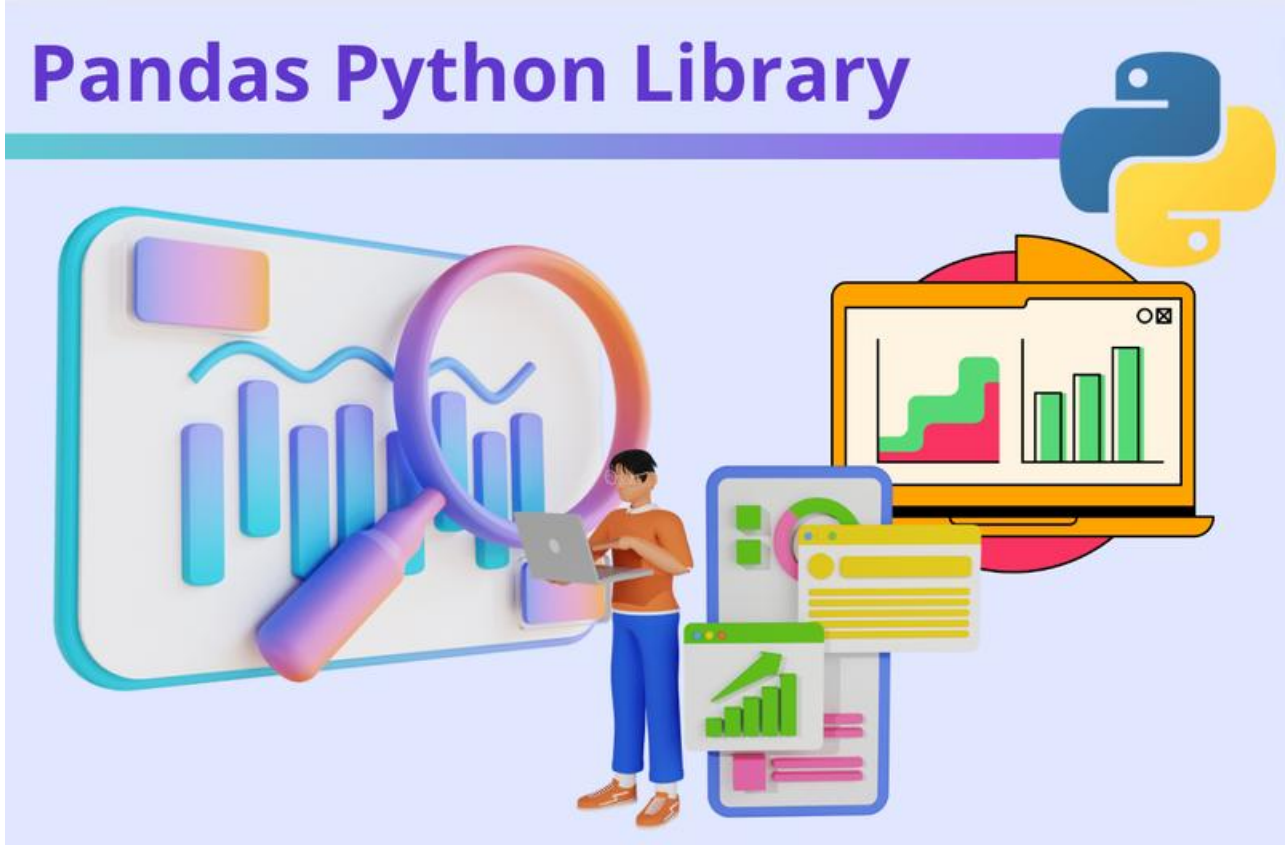
HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Introduction to data analysis

- Train simple data analysis tasks to structure the given data into a Pandas DataFrame, get summary about the data, filter and sort the data, add new columns to the original data.

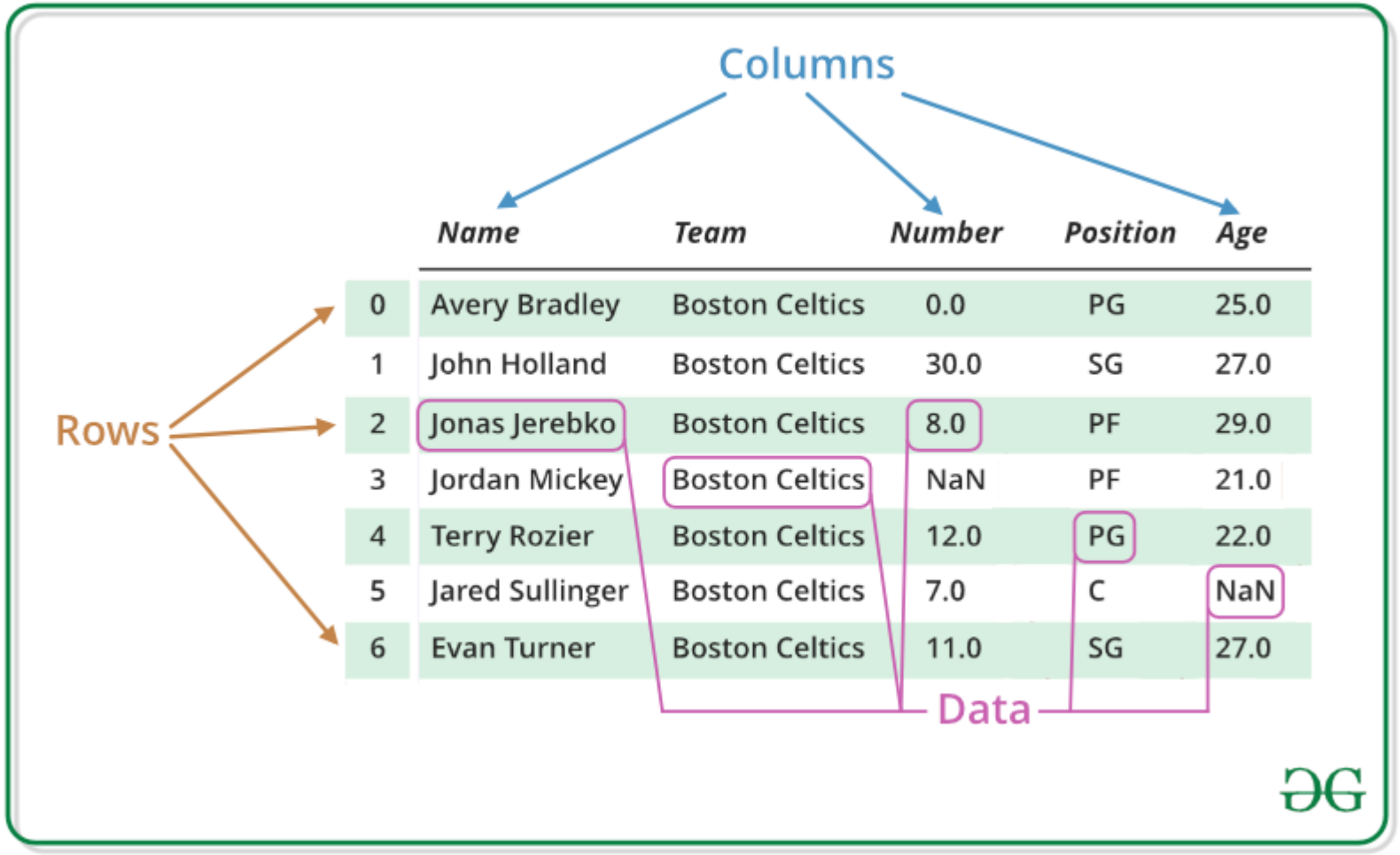


HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Pandas library



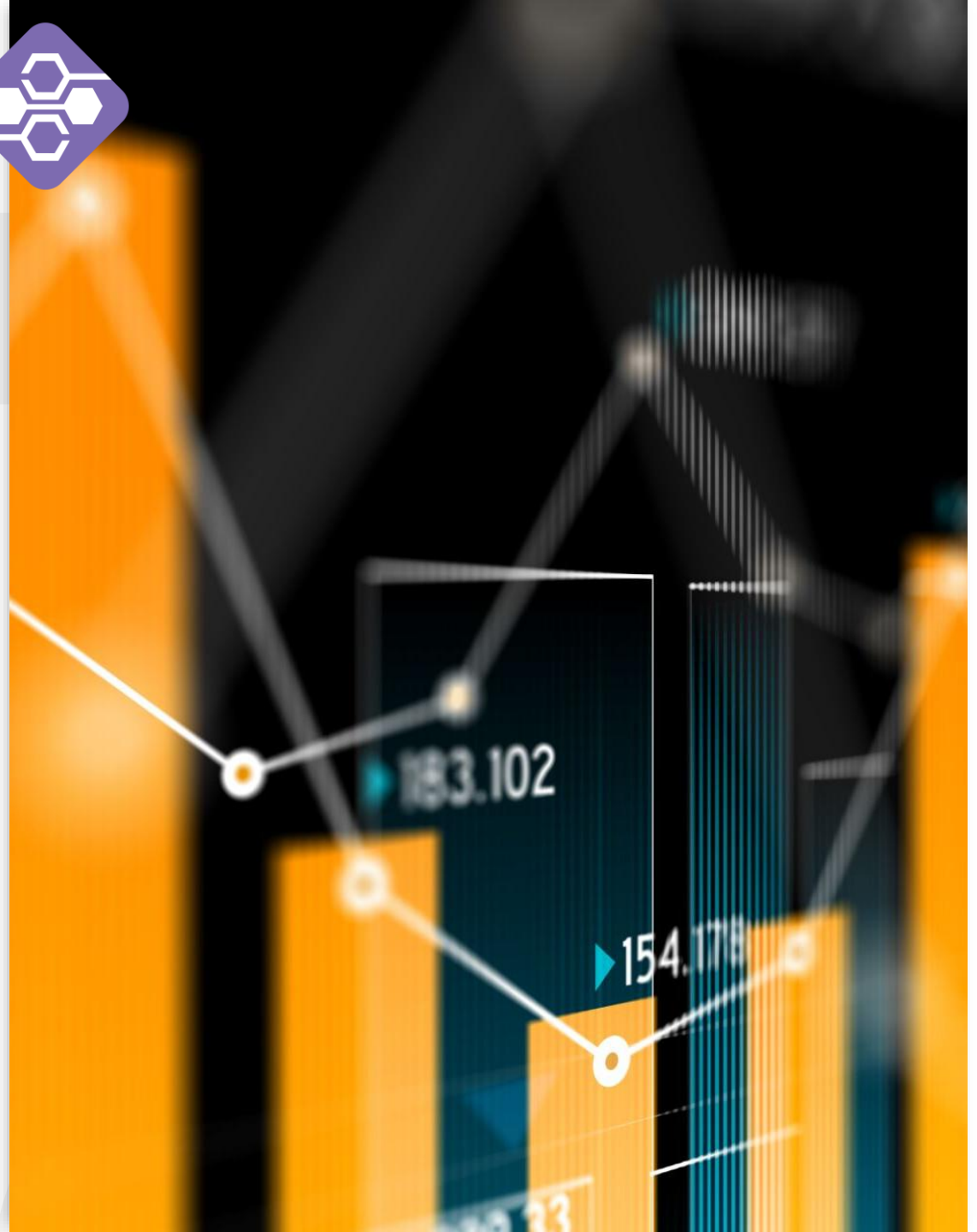
HAMK
Häme University
of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin
osarahoittama

Visualization

- Load CSV dataset file to get visualizing plots and tables.



HAMK
Häme University of Applied Sciences

REGIONAL UNIVERSITY NETWORK
EUROPEAN UNIVERSITY

Euroopan unionin osarahoittama

Thank you!