

# Programming for psychologists

**Lecture 1: Introduction**

Matthias Nau

**Welcome to  
this course!**



# Welcome to this course!

## Who are we?

**Matthias Nau**

Course coordinator



Assistant Professor

[m.nau@vu.nl](mailto:m.nau@vu.nl)

**Anna van Harmelen**

Teaching Assistant

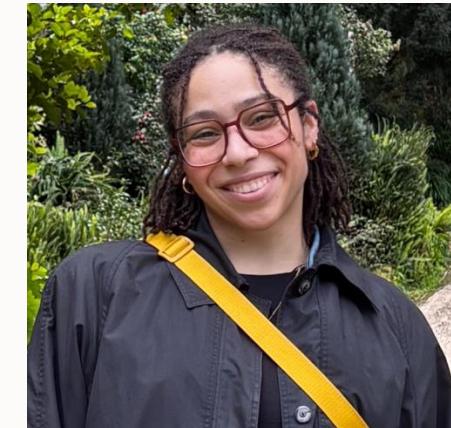


PhD Candidate

[a.m.van.Harmelen@vu.nl](mailto:a.m.van.Harmelen@vu.nl)

**Camilla U. Enwereuzor**

Teaching Assistant



PhD Candidate

[c.u.enwereuzor@vu.nl](mailto:c.u.enwereuzor@vu.nl)

**Feel welcome to Email us! When you do, please put [Programming] in the Email subject!**

# What is this course about?

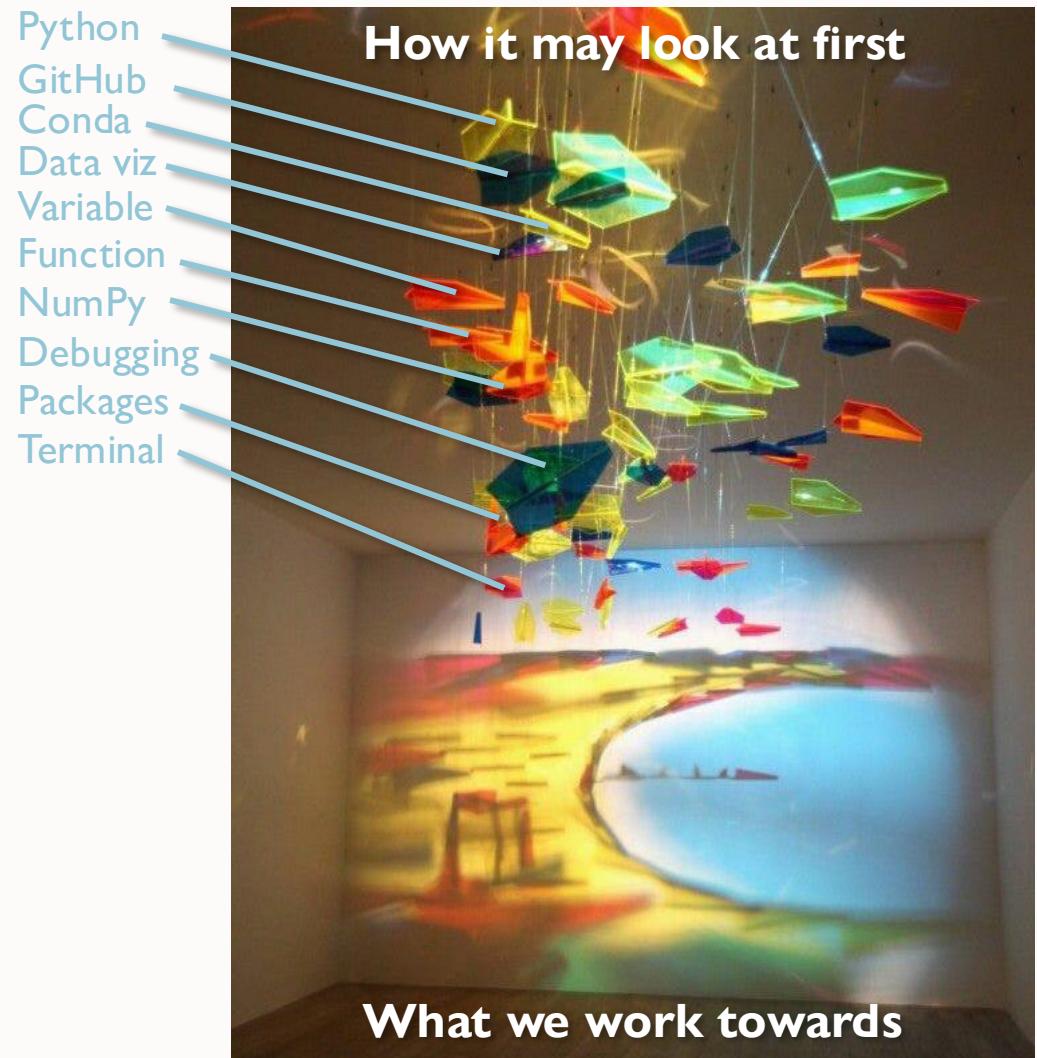
**Programming** is the process of telling a computer which **tasks** to perform in order to **solve problems**.

We create **instructions** in a language that computers understand (i.e. code).

---

Focus on the **Big Picture!**

Once you find the right **perspective**, you will learn everything you need to learn through **practice**.



# Course outline

**The course is organized into 7 modules, each comprising 1 lecture and 2 practicals**  
Each module addresses a different aspect of programming relevant to psychologists.

## Lectures focus on conceptual understanding

Why is programming important for psychologists? What is Python and how do I use it?  
What are best practices and what does Open Science have to do with them?  
*Lectures will be recorded, but expect delays before these recordings are posted.*

## Practicals focus on skills

Programming basics and typical workflows. Using Python for data analysis and visualization.  
Simple experimental design. How to share and review code.  
*Practicals will not be recorded, unless they contain shorter lectures.*

# Grading

## I) Exam

What: Answer questions about programming and about code that we provide (e.g., What does this code do?)

Weight: 50% of the final grade

Time: End of December

## 2) Home assignment

What: Define research interest, download corresponding data, write code to visualize the data, upload the code to an online platform and provide feedback on someone else's code.

Weight: 25% of the final grade

Time: Work in your own time from October to December

## 3) Quizzes

What: Answer multiple choice or multiple answer questions about the course content.

Weight: 25% of the final grade

Time: Weekly, always available for one week following the lecture (deadline: Sunday night).

# What to expect?

## **Understand the “programming mindset”**

Learn a new way of thinking about goals, problems, solutions, collaboration etc.



## **Demystifying key terms and concepts**

Get a good overview on the topic and let go of intimidation

## **Internalize best practices**

Learn about the values of Open Science with regards to code sharing and code review

## **Get to know workflows of a research psychologist**

Understand how programming can help you in (almost) every step of your research

## **Learn about tools for continued learning**

We will help you learn how to program beyond the course by introducing helpful resources.

## **Familiarize yourself with the Python programming language**

Acquire basics in Python for experimental design, data analysis, and visualization

# What NOT to expect?

**Do not expect to be fluent in Python before using it**  
Learning a new (programming) language takes practice.



**Do not expect to be Python experts by the end of this course**  
But you will know how to use it and how you can continue learning!

**Do not expect others to succeed or struggle with the same things as you**  
Everyone is different. Help others, and be open to learn from them.

**Do not expect us to be perfect or all-knowing**  
This course is still quite new and we are trying out a few things. Please help us improve by giving feedback. It will make the course better for you and everyone else.

I came here to  
study **psychology**.

Why would I want to  
learn **programming**?



# Programming is **EMPOWERING**

You will **learn to use a tool** that allows you **to build tools** (i.e. programs that work for you).

With it, many things you thought were impossible for you suddenly become possible.

From experimental design to statistical inference, you gain **full control** over your research.

## *Prediction*

*You will find yourself one beautiful Sunday morning wondering how to use programming to solve all sorts of life tasks :)*



**Coding is your new superpower!**

# Programming is **FUN**

Coding is among the most **creative** components of your work as a research psychologists.

Unlike many other academic tasks, it involves **direct feedback** and daily rewards.

**Trackable progress** on short and long time scales (e.g., lines of code written for a project).



## **Prediction**

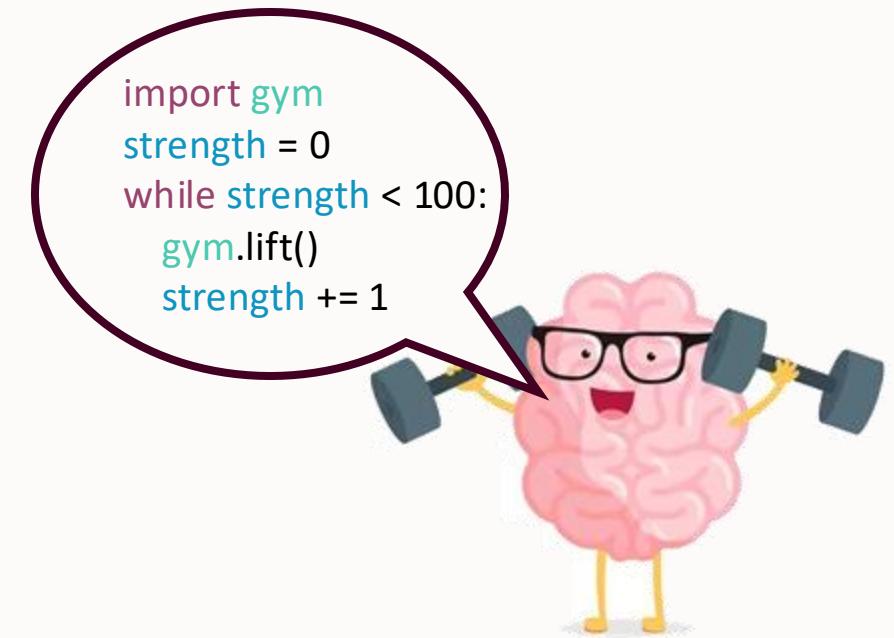
*One day, a program you have written will solve a problem you and others have had for a long time, and it will feel great! :)*

# Programming changes your **MINDSET**

Knowing how to program informs your thinking beyond work (like learning a new language).

It teaches you how to **embrace problems** and **find solutions** in a goal-oriented manner.

It requires **discipline** and **practice**, but you will feel stronger and stronger as time goes on.



## ***Prediction***

*You will start to see parallels between programming and all sorts of other things in life (e.g., cooking, or putting together IKEA furniture)*

# Programming is **EFFICIENT**

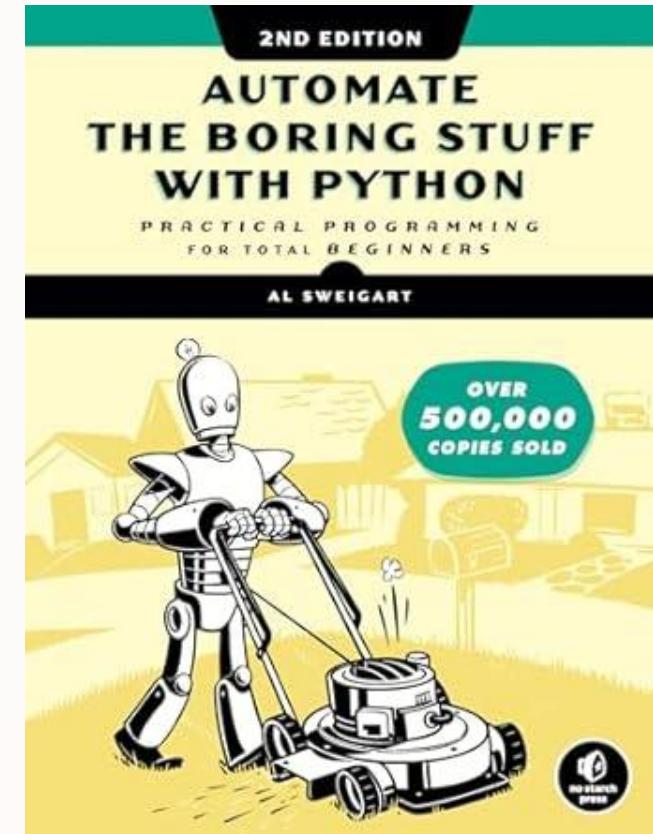
**Your time** is the most **precious currency** you have.  
**Don't waste it** on annoying or repetitive tasks!

By **automating processes**, programming can  
**save A LOT of time** and make you more **efficient**.

## *Prediction*

*You will want to automate more and more tasks in your work, from moving files between folders to data analysis*

Recommended reading:  
[automatetheboringstuff.com](http://automatetheboringstuff.com)



# Programming is a **TRANSFER SKILL**

Coding skills are **widely applicable** independent of the topic or method you choose for your research.

This course focuses specifically on skills that translate directly to **better experiences and grades** in many other courses.

Coding will allow you to **make the most of your studies**.



“I can do anything”

## *Prediction*

*No matter what topic you choose for your MSc thesis, you will look back to this course and think “glad I know how to do some of this already!”.*

# Programming STAYS INTERESTING

You will never be finished with learning how to program;  
**the path is the goal**, and no programmer knows it all.

There is **always something new** to be learned, a function you have never used, a new way of structuring code...

This **continuous learning** will remain to be **rewarding**



## *Prediction*

*One day you will look back and think “I wish I had more time for coding and data analysis”.*

# Programming drives **REPRODUCABLE RESEARCH**

Psychology is in a **replication crisis**.  
Many established results cannot be reproduced.

Coding allows you to create **fully reproducible analysis pipelines** that other researchers can use and assess for **accuracy**.

Good programming skills are one building block towards a **better future for science**.

## *Prediction*

*Many publishers will require you to share your code, and funding agencies & employers will value code sharing more.*

The image shows two overlapping news articles from the journal *Science*. The top article is titled "Psychology's Replication Crisis Is Running Out of Excuses" by David B. Allison, Andrew W. Brown, Brandon J. George, and Kathryn A. Kaiser, published on 03 February 2016. It includes a sub-headline "Reproducibility: A tragedy of errors" and author information. The bottom article is titled "The legacy of the reproducibility crisis" by Christopher Alibertyn, published on 09 March 2018. It includes a sub-headline "The replication crisis devastated psychology. This group is looking to rebuild it." and author information. Both articles mention the Psychological Science Accelerator and its potential to sustain the field. A callout box at the bottom right says "+Many more".

SCIENCE  
Psychology's Replication Crisis Is Running Out of Excuses  
... project has found that only half of studies can be repeated.  
Comment | Published: 03 February 2016  
**Reproducibility: A tragedy of errors**  
David B. Allison , Andrew W. Brown, Brandon J. George & Kathryn A. Kaiser  
Reproducibility  
**The legacy of the reproducibility crisis**  
Christopher Alibertyn   
SCIENCE  
The replication crisis devastated psychology. This group is looking to rebuild it.  
The Psychological Science Accelerator could be the future of the field around the globe — if they can sustain it.  
+Many more

# Programming **STRENGTHENS YOUR RESUME**

Because of all the benefits they bring, good coding skills are highly sought after in **academia & industry**.

Being able to demonstrate good coding skills will significantly **strengthen your resume**, and make you a valuable asset in **any research environment**.



## **Prediction**

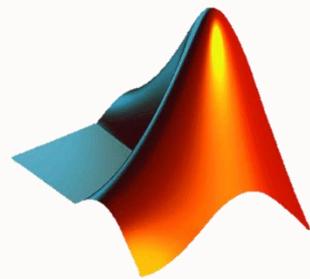
*Whether you stay in academia or not, many of you will be asked about programming skills when applying for positions later on in your career.\**

\*one of the reasons I got this job was because I could develop and teach this course!

**Great! Now that we  
know what we want,  
how do we do it?**



# Popular programming languages in Psychology



MatLab



Julia



R



JavaScript



Python

This is what we will be using

# What is **Python** and why do we use it?

**Python** is a **powerful, general-purpose** programming language that is **widely used** in academia, medical research, and industry

## Reasons for using Python include:

- Simple rules that are easy to learn
- It is popular within psychology and beyond
- Huge existing code base (e.g., many toolboxes)
- It has a strong online support community (e.g., *Stackoverflow.com*)
- Python is free for everyone (Open Science)



```
# Define each part of the word
part1 = "Psy"
part2 = "cho"
part3 = "logy"

# Combine the parts to form the word "Psychology"
word = part1 + part2 + part3

# Print the word
print(word)
```

✓ 0.0s

Psychology

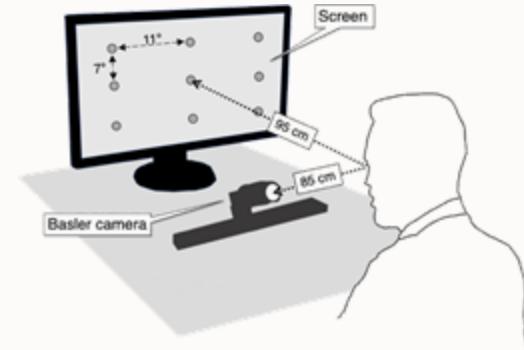
Python

**Example code**

# Course outline



- Oct 27 – Oct 31: Module 1: **Welcome**
- Nov 03 – Nov 07: Module 2: **Python basics**
- Nov 10 – Nov 14: Module 3: **Python advanced**
- Nov 17 – Nov 21: Module 4: **Data visualization**
- Nov 24 – Nov 28: Module 5: **Code sharing & collaboration**
- Dec 01 – Dec 05: Module 6: **Experimental design**
- Dec 08 – Dec 12: Module 7: **Course recap & workflow**
- Dec 15 – Dec 18: End of course: **Exam, home assignment**



## Attention:

We might make changes to this schedule along the way depending on how everyone is doing!

# Announcements:

**Next up: Practical 1.1 tomorrow at 10am in room NU-5B21.**

**The times on canvas or communicated via email count,  
not the one on rooster.vu.nl!**

**Attention 1: You will need your laptops!**

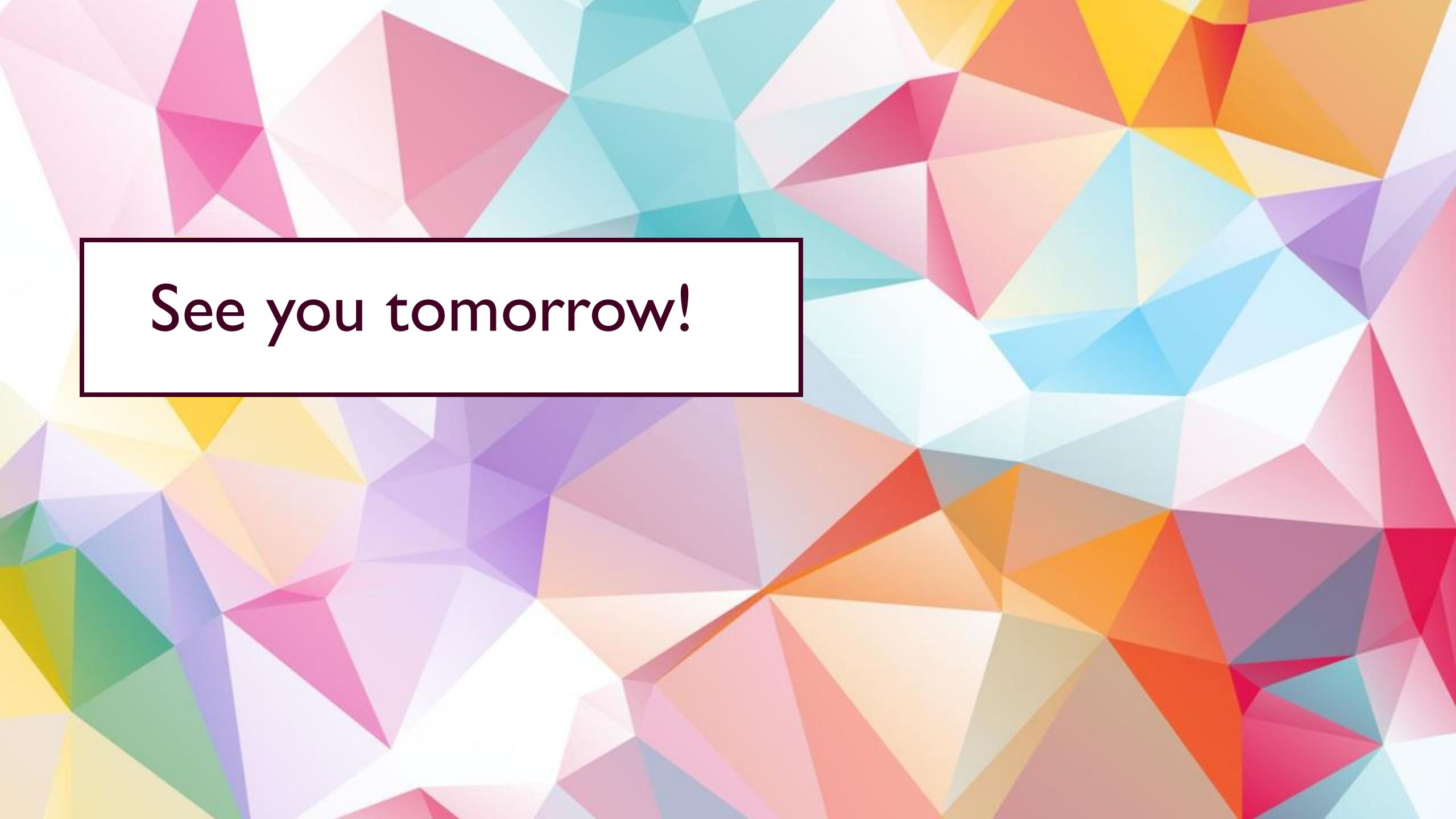
This is true for all practicals except the one tomorrow (Practical 1.1)

**Attention 2: Watch the recording of Practical 1.2!**

There will be NO in-person practical this upcoming Friday.

Instead, please watch the video on canvas and follow all steps described in it.

If you get stuck, please consult the FAQ page on canvas or reach out.



A vibrant, abstract background composed of numerous overlapping triangles in a variety of colors, including shades of pink, purple, yellow, blue, and orange. The triangles are of different sizes and orientations, creating a dynamic and textured pattern across the entire frame.

See you tomorrow!