

# Mathematics and Computer Science for Modeling

## Unit 1: Introduction to Programming in Python

Stephan Sehring

based on materials by Jan Tekülve and Daniel Sabinasz

Institut für Neuroinformatik, Ruhr-Universität Bochum

23.09.2024

## Why this course?

- ▶ Anyone with a Bachelor's degree in any of the cognitive sciences can start this course
- ▶ Not all of you will have the same level of background knowledge for all of the lectures
- ▶ The preparatory courses are here to help you bridge that gap
- ▶ Goal: Bring you on a similar level regarding mathematics and computer science skills
- ▶ ... which will hopefully make it easier for you to get through the Master programme
- ▶ The course is not mandatory, but recommended

# Course concept

- ▶ The course is split into lecture parts and exercise parts

# Exam

- ▶ At the end of the course, there will be a written exam (04.10. at 12 am)
- ▶ The exam is graded, but this is only for your feedback and won't enter into your average grade

# Dates

1. Mon 23.09. 15-17:30
2. Tue 24.09. 15-17:30
3. Wed 25.09. 15-17:30
4. Thu 26.09. 15-17:30
5. Fri 27.09. 15-17:30
6. Mon 30.10. 15-17:30
7. Tue 01.10. 15-17:30
8. Wed 02.10. 15-17:30
9. Fri 04.10. 15-17:30

# Course Structure

Unit	Title	Topics
1	Intro to Programming in Python	<i>Variables, if Statements, Loops, Functions, Lists</i>
-	Full-Time Programming Session	<i>Deepen Programming Skills</i>
2	Functions in Math	<i>Function Types and Properties, Plotting Functions</i>
3	Linear Algebra	<i>Vectors, Trigonometry, Matrices</i>
4	Calculus	<i>Derivative Definition, Calculating Derivatives</i>

# Course Structure

Unit	Title	Topics
5	Integration	<i>Geometrical Definition, Calculating Integrals</i>
6	Differential Equations	<i>Properties of Differential Equations</i>
-	04.10.24: Test	

# Lecture Slides/Material

Use the following URL to access the lecture slides:

[https://www.ini.rub.de/teaching/courses/preparatory\\_course\\_mathematics\\_and\\_computer\\_science\\_for\\_modeling\\_winter\\_term\\_2024](https://www.ini.rub.de/teaching/courses/preparatory_course_mathematics_and_computer_science_for_modeling_winter_term_2024)



# Questions

- ▶ Email me with any questions you might have:  
`stephan.sehring@ini.rub.de`

# The Python Programming Language

## Why Python?

- ▶ It is simple
- ▶ It is high-level
- ▶ It is interpreted “on the fly”
- ▶ It is widespread in the scientific community

## Helpful Resources

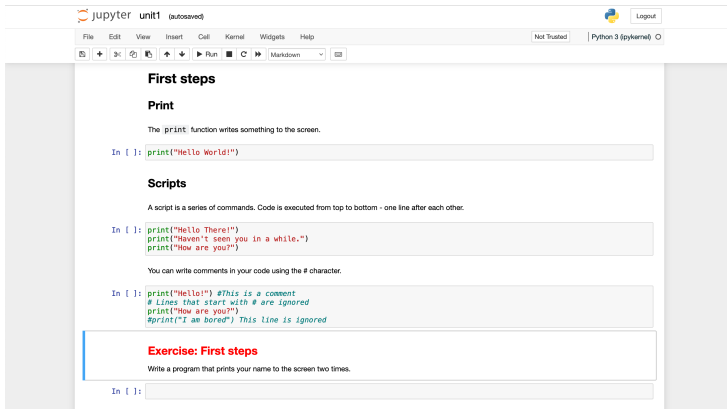
- ▶ The Anaconda Distribution contains all necessary software:  
<https://www.anaconda.com/distribution/>
- ▶ You can find helpful documentation here: <https://docs.python.org/3/>

# Getting Started

- ▶ Install Anaconda: <https://www.anaconda.com/>
- ▶ Download the document “Jupyter notebook” for Unit 1 (filename “unit1.ipynb”) from the course website
- ▶ Start the program “Anaconda-Navigator”. Find the application “Jupyter Notebook” and click on “launch”.
- ▶ (Alternative: Start the program “Anaconda Prompt”. Wait for a prompt to appear and then enter “jupyter notebook” into that prompt)
- ▶ Navigate to the directory where you saved the “unit1.ipynb” file and click on that file

# Getting Started

- ▶ You are now presented with a so-called Jupyter Notebook, a document that allows you to execute existing Python code and write your own Python code while being guided by narrative text

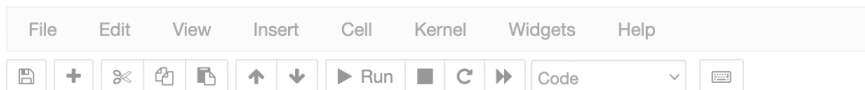


The screenshot displays a Jupyter Notebook interface. At the top, the title bar reads 'jupyter unit1 (autosaved)' with a 'Logout' button on the right. Below the title bar is a menu bar with options: File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. To the right of the menu bar are status indicators: 'Not Trusted' and 'Python 3 (ipykernel)'. Below the menu bar is a toolbar with icons for file operations, cell navigation, and execution. The main content area is titled 'First steps' and contains the following sections:

- Print**  
The `print` function writes something to the screen.  
In [ ]: `print("Hello World!")`
- Scripts**  
A script is a series of commands. Code is executed from top to bottom - one line after each other.  
In [ ]: `print("Hello There!")`  
`print("Haven't seen you in a while.")`  
`print("How are you?")`  
You can write comments in your code using the `#` character.  
In [ ]: `print("Hello!")` *#This is a comment*  
*# Lines that start with # are ignored*  
`print("How are you?")`  
*#print("I am bored") This line is ignored*
- Exercise: First steps**  
Write a program that prints your name to the screen two times.  
In [ ]:

# Getting Started

jupyter unit1 (unsaved changes)



## First steps

**1**  
click on a code cell  
to select it

## Print

**2**  
click "Run" to  
execute the code

The `print` function writes something to the screen.

**3**  
observe  
the output of the  
code below the cell

```
In [1]: print("Hello World!")
```

Hello World!