Aalto University

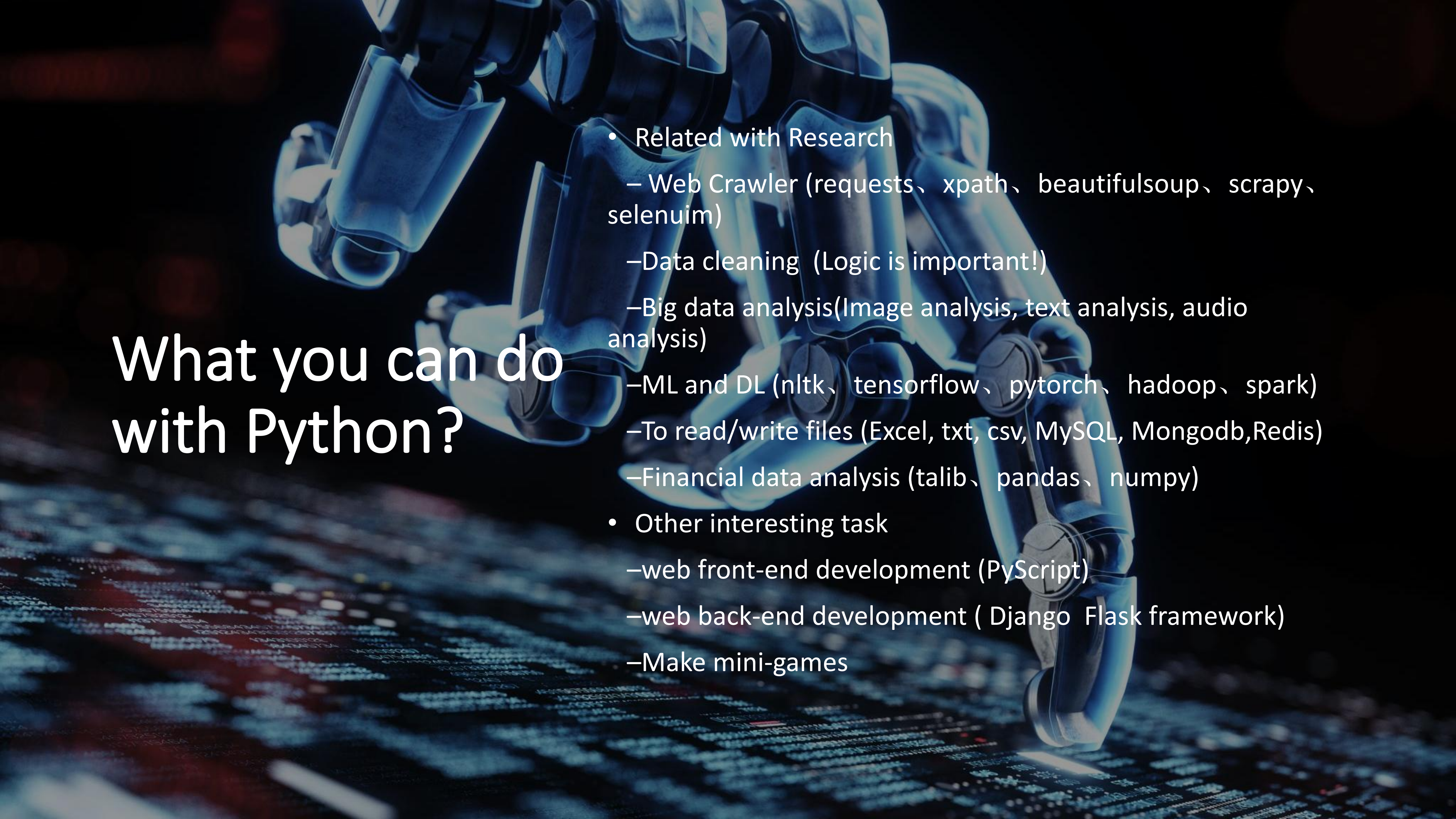
37C00400 Programming I

Introduction to the Course - October 25, 2023

***Yuting Jiang**, Doctoral Researcher, ISM*

Supporting teacher: Johanna Bragge, Principal University Lecturer

Some of the slides prepared originally by A. Frantsev and V. Marella



What you can do with Python?

- Related with Research

  – Web Crawler (requests、xpath、beautifulsoup、scrapy、selenuim)

  –Data cleaning (Logic is important!)

  –Big data analysis(Image analysis, text analysis, audio analysis)

  –ML and DL (nltk、tensorflow、pytorch、hadoop、spark)

  –To read/write files (Excel, txt, csv, MySQL, Mongodb,Redis)

  –Financial data analysis (talib、pandas、numpy)

- Other interesting task

  –web front-end development (PyScript)

  –web back-end development (Django Flask framework)

  –Make mini-games

Aalto University

Content

1. Practicalities
2. Getting started
3. Troubleshooting and getting help

1. Practicalities

Course overview

- This is an online self-paced course
  - That means no compulsory lectures and no weekly deadlines,

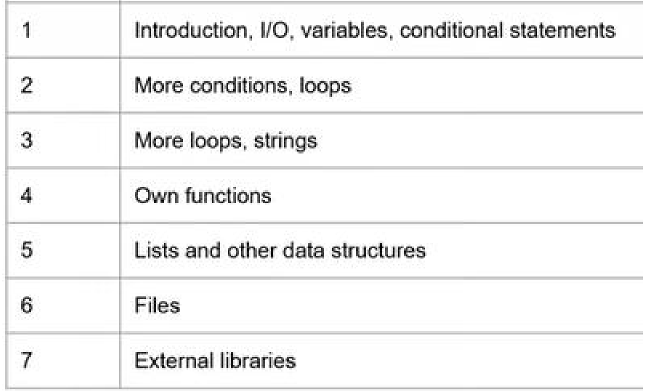
    but two deadlines for completing the 201 exercises

    Doing the exercises take a considerable amount of time!

    Around 10-20 hours per week depending on experience.
- The course is based on University of Helsinki's **Introduction to programming** with Python MOOC
  - However, we have our own exam, deadlines, support sessions and chat forum!

Exercises

- **7 Parts** of exercises to be completed in 7 weeks
- There are two (**2**) deadlines; otherwise you can work at your own pace
- Each seven part's exercises unlock only after completing **70%** of the previous part’s exercises
- Exercises from parts 1-4 must be completed by **November 15**, and from parts 5-7 by **December 11**, 2023 **(midnight 23:55)**
  - – even if you plan to take the second exam
  - – else, you get only the exercise points you have on Dec 11
    - Note that a scaling of 90% is applied to points from parts 1-4 if you submit them after the interim deadline on November 15 (and submitted by December 11 at the latest).



1	Introduction, I/O, variables, conditional statements
2	More conditions, loops
3	More loops, strings
4	Own functions
5	Lists and other data structures
6	Files
7	External libraries

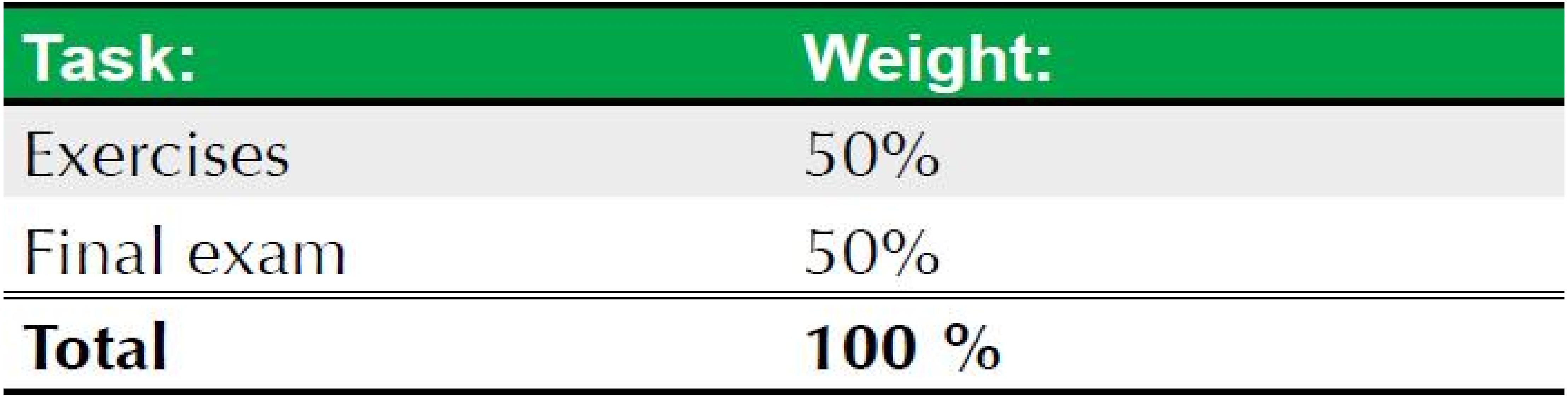
Exam

- *PRELIMINARY INFO, format of exam may change to a class-exam with own computers or to an e-Exam due to LLM's!*
- Exam
    - You will have max **3** hours (possibly less) for the exam
    - exam has previously consisted of similar tasks than in the course exercises (this might change)
- There are only two chances to take the exam **(Follow Sisu)**
    - Tuesday, **December 12, 2023 at 13-16 o'clock**
    - **February 12, 2024 at 16-19 o'clock**

Course workload estimate

Opening lecture (voluntary)	2h
Weekly help sessions (voluntary)	10h
Independent work with the programming exercises	128h
Exam and preparing for the exam	20h
Total	160h (6 cr)

Grade composition



Task:	Weight:
Exercises	50%
Final exam	50%
Total	**100 %**

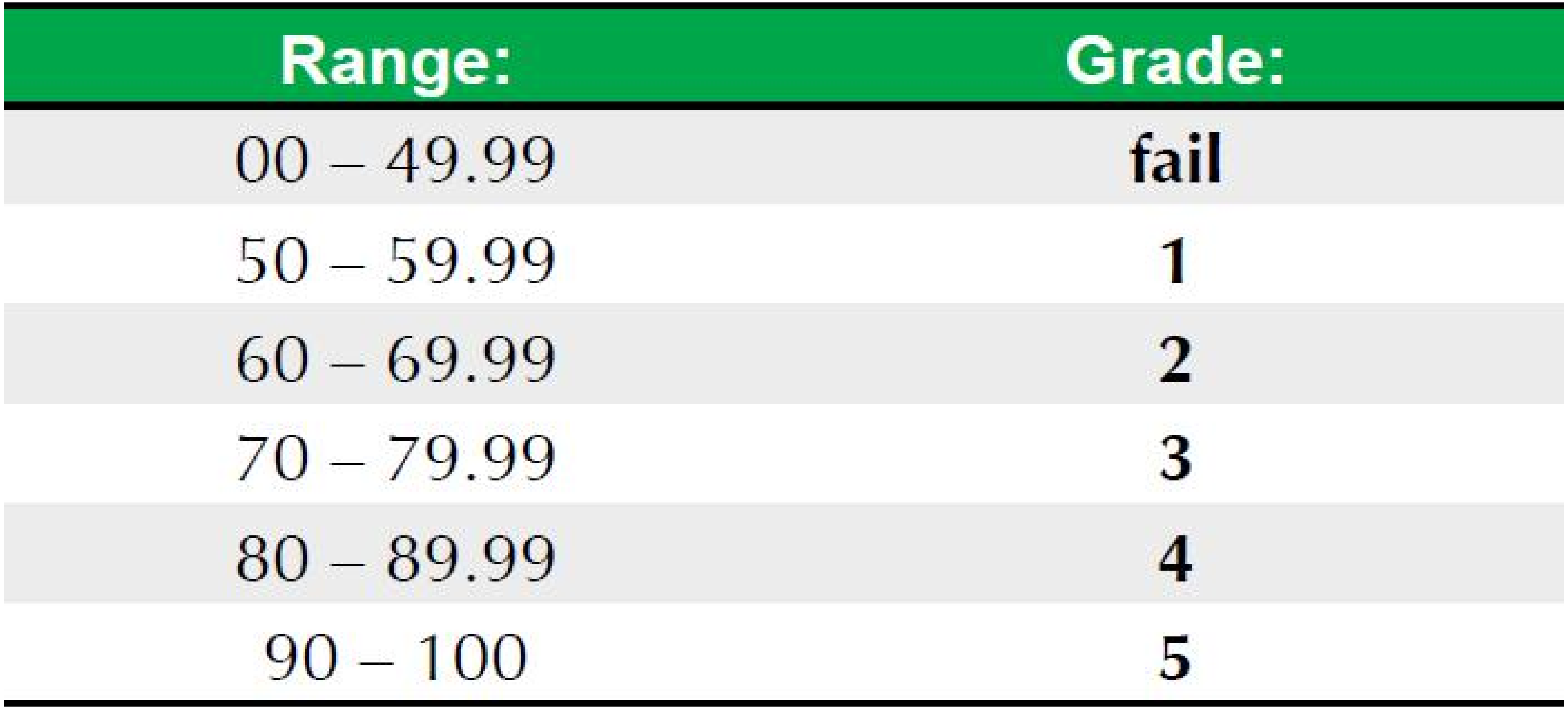
NOTE: You must obtain **at least 50%** of the points in both exercises and exam in order to pass the course, see example on next slide.

Examples how the grade is calculated

Exercise points	Exam Points	% Exercise	% Exam	% Total	Grade
201	30	100,00	100,00	100,00	5
101	15	50,00	50,00	50,00	1

Max points from the exercises is ca. 201 and 30 from the exam, so you must achieve at least 101 and 15 points, respectively, to pass the course.

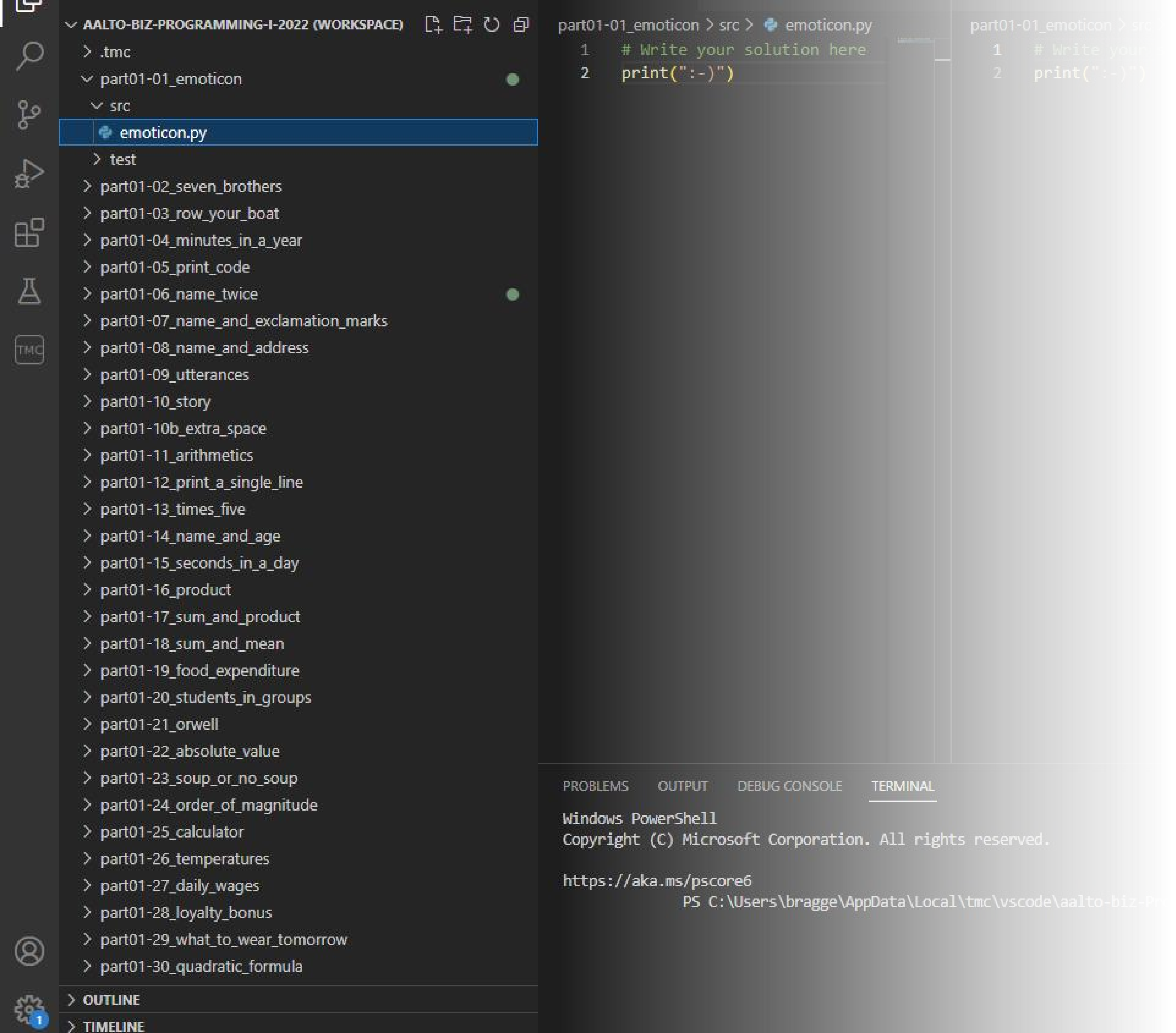
Grading Scale



Range:	Grade:
00 – 49.99	**fail**
50 – 59.99	**1**
60 – 69.99	**2**
70 – 79.99	**3**
80 – 89.99	**4**
90 – 100	**5**

Usage of conversational AI and its plugins

- In this introductory programming course you should learn the logics of programming. This can be achieved only by hands-on programming experience.
- Thus, you are not allowed to use ChatGPT or its plugins to produce the code in the exercises to start with.
  - However, you can use them for trouble-shooting when you have first tried yourself to code the solutions.
  - Use enough time for hands-on learning-by-doing!



```python
# Write your solution here
print(":-)")
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

https://aka.ms/pscore6
                    PS C:\Users\bragge\AppData\Local\tmc\vscode\aalto-biz-Pr
```

2. Getting Started

Step 1: Register in TMC (TestMyCode)

- Register a **new user account** at (or *use your **existing Mooc.fi** account*):
  - https://tmc.mooc.fi/org/aalto-biz

    Click the **Sign up** button and add your details.
    Submit your **answer choice** to the research done on the online courses.
    Confirm your account via email sent by MOOC.fi (check also your junk email!)
- **Return** to page https://tmc.mooc.fi/org/aalto-biz and **log in**

- **Add your Student ID *number* in the Organizational Identifier field**
  - this is important in order to differentiate participants' similar names!

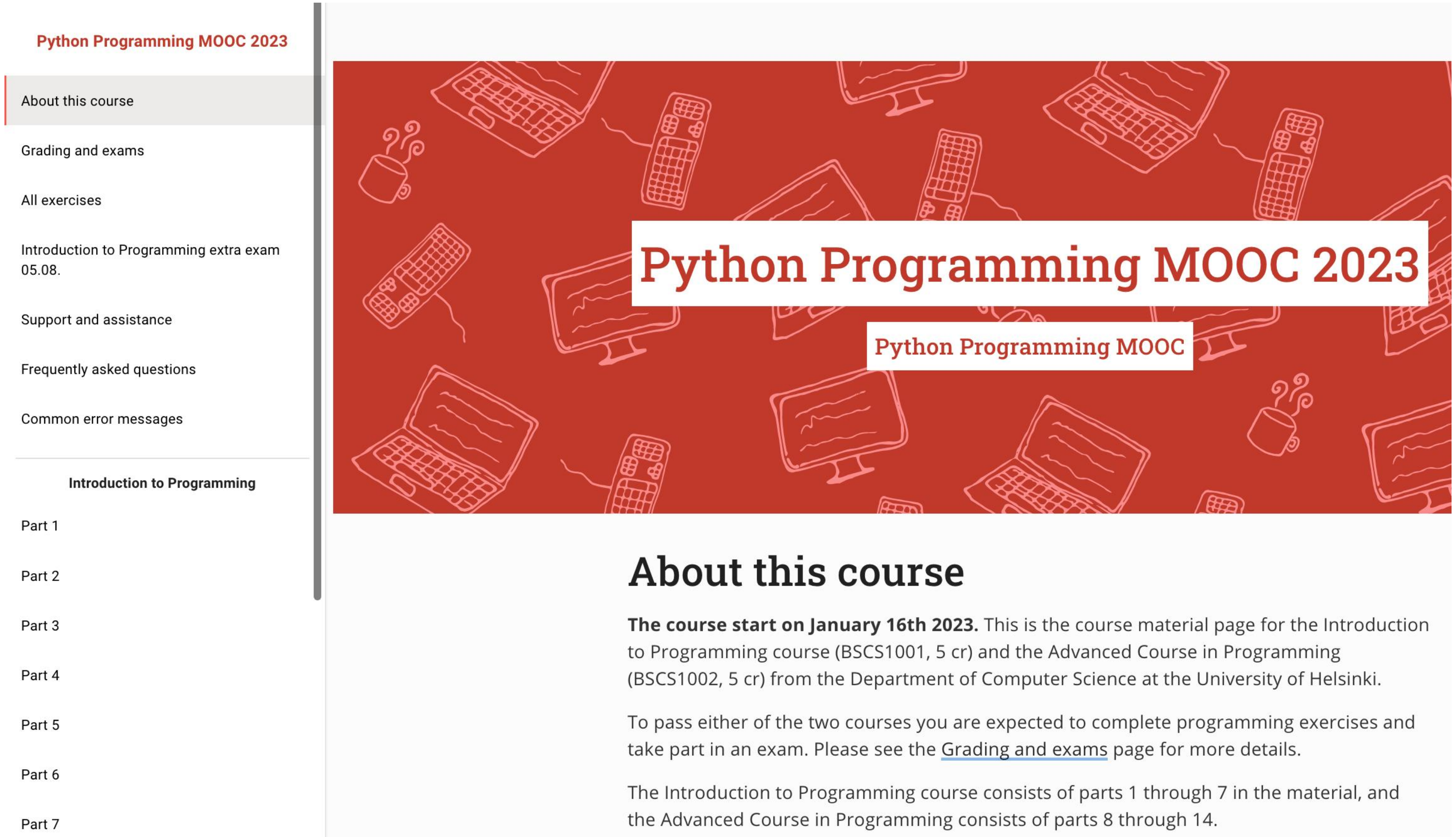
  The organizational identifier can be added under **My account > Settings tab** (click the account email box on top-right, select Settings, scroll down, enter ID & save).

Step 2: Go to https://programming-23.mooc.fi

This is the actual course page with the **exercise task instructions**

*NOTE: in BIZ course you will need to submit the exercises in **VS Code** environment right from the beginning and not in the browser! Otherwise, your points are not visible to BIZ course instance teachers.*

VS Code also helps you along when writing the code!



About this course

The course start on January 16th 2023. This is the course material page for the Introduction to Programming course (BSCS1001, 5 cr) and the Advanced Course in Programming (BSCS1002, 5 cr) from the Department of Computer Science at the University of Helsinki.

To pass either of the two courses you are expected to complete programming exercises and take part in an exam. Please see the Grading and exams page for more details.

The Introduction to Programming course consists of parts 1 through 7 in the material, and the Advanced Course in Programming consists of parts 8 through 14.

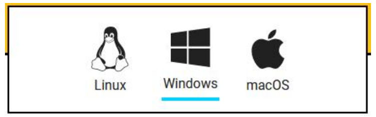
Aalto University

Step 3: Install *VS Code, TestMyCode extension & Python and its extension for VS Code*

To install the development environment, go to:

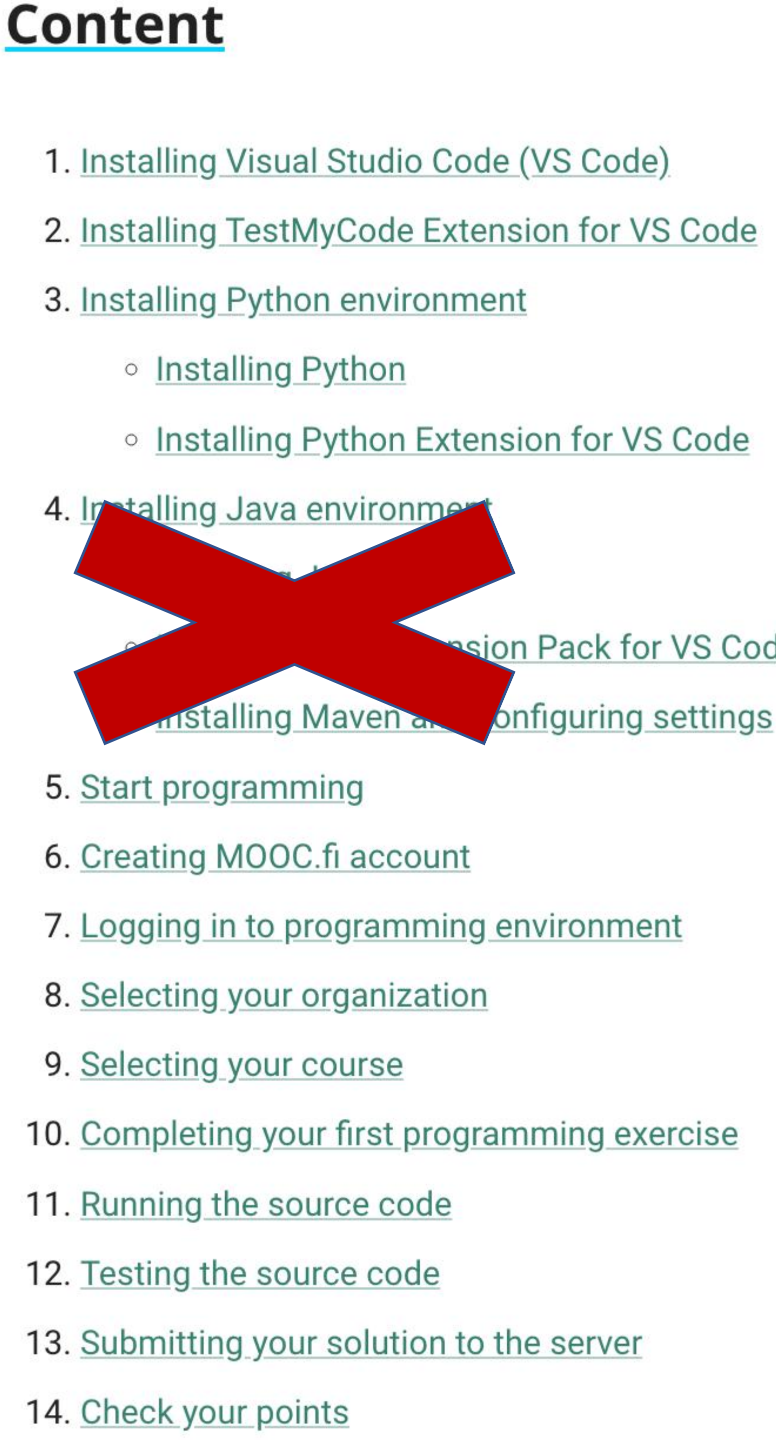
https://www.mooc.fi/en/installation/vscode/

- Select first your operating system:



- Follow the instructions in **points 1-3**

- ***Don't*** follow the 4. **Installing Java** instructions (it is needed for another/ Java language course)



Content

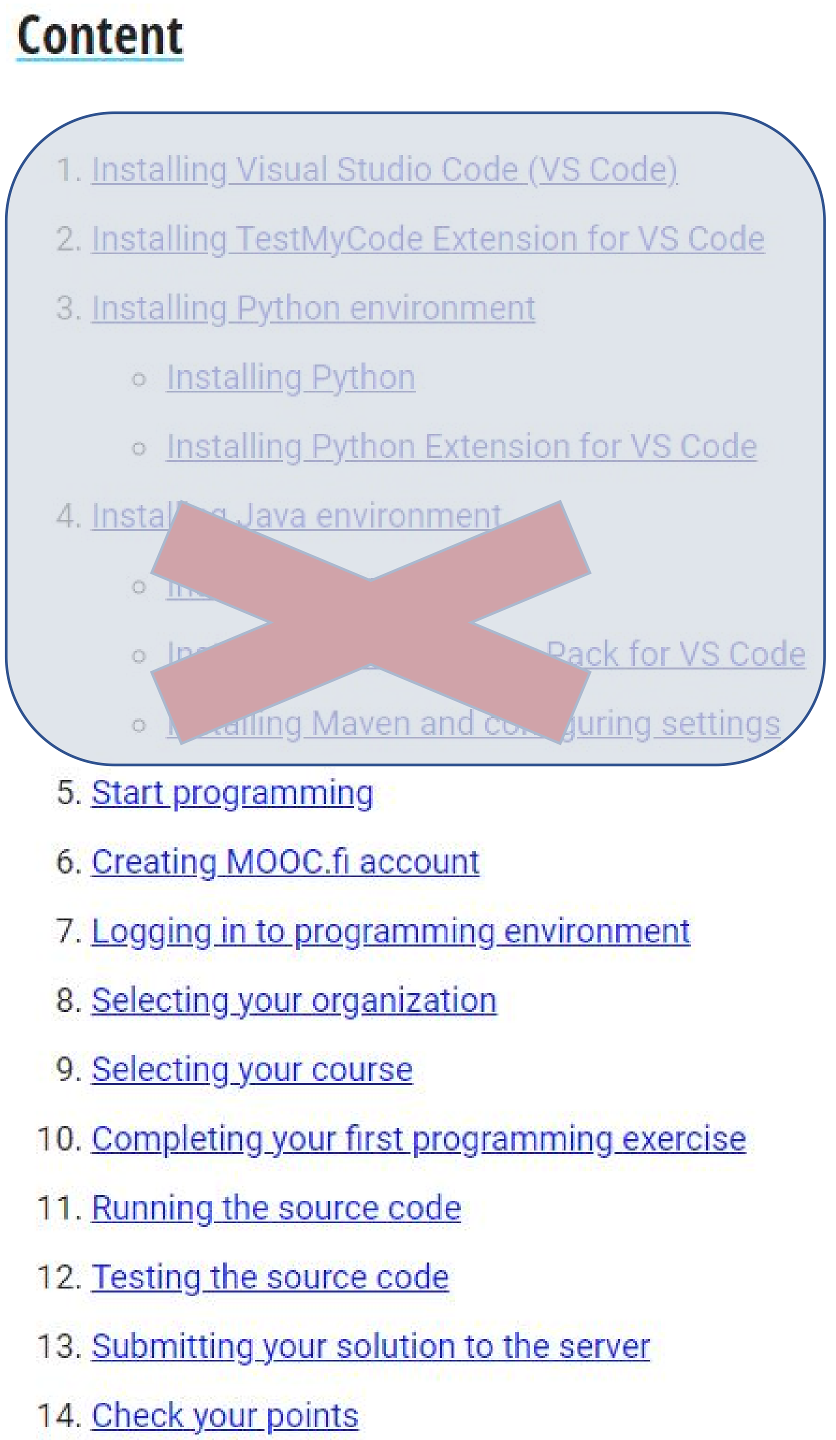
1. Installing Visual Studio Code (VS Code)
2. Installing TestMyCode Extension for VS Code
3. Installing Python environment
   - Installing Python
   - Installing Python Extension for VS Code
4. ~~Installing Java environment~~
   - ~~sion Pack for VS Cod~~
   - ~~nstalling Maven a onfiguring settings~~
5. Start programming
6. Creating MOOC.fi account
7. Logging in to programming environment
8. Selecting your organization
9. Selecting your course
10. Completing your first programming exercise
11. Running the source code
12. Testing the source code
13. Submitting your solution to the server
14. Check your points

Step 4: Continue the instructions from point 5: Start programming

Use the MOOC.fi account you created in Step 1 to log in to VS Code

Selected step-by-step pictures on the MOOC instructions page have been reproduced on the next slides for the BIZ-specific course instance!

Aalto University

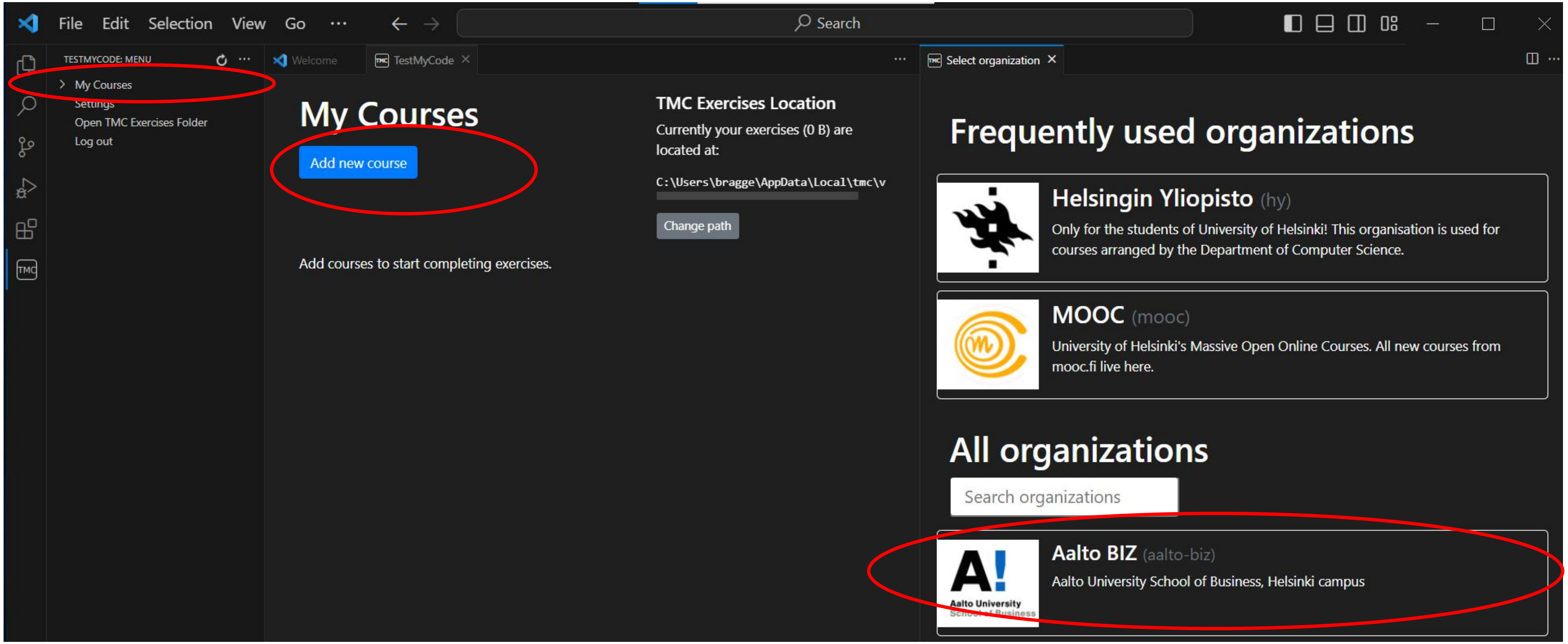


Content

1. Installing Visual Studio Code (VS Code)
2. Installing TestMyCode Extension for VS Code
3. Installing Python environment
    - Installing Python
    - Installing Python Extension for VS Code
4. Installing Java environment
    - In
    - In Pack for VS Code
    - Installing Maven and configuring settings
5. Start programming
6. Creating MOOC.fi account
7. Logging in to programming environment
8. Selecting your organization
9. Selecting your course
10. Completing your first programming exercise
11. Running the source code
12. Testing the source code
13. Submitting your solution to the server
14. Check your points

Adding the course to VS Code

- Click **My Courses** tab on the left and click Add new course
- Scroll list on right and select first **Aalto BIZ** from the organizations



My Courses

Add new course

Add courses to start completing exercises.

TMC Exercises Location

Currently your exercises (0 B) are located at:

`C:\Users\bragge\AppData\Local\tmc\v`

Change path

Frequently used organizations

Helsingin Yliopisto (hy)

Only for the students of University of Helsinki! This organisation is used for courses arranged by the Department of Computer Science.

MOOC (mooc)

University of Helsinki's Massive Open Online Courses. All new courses from mooc.fi live here.

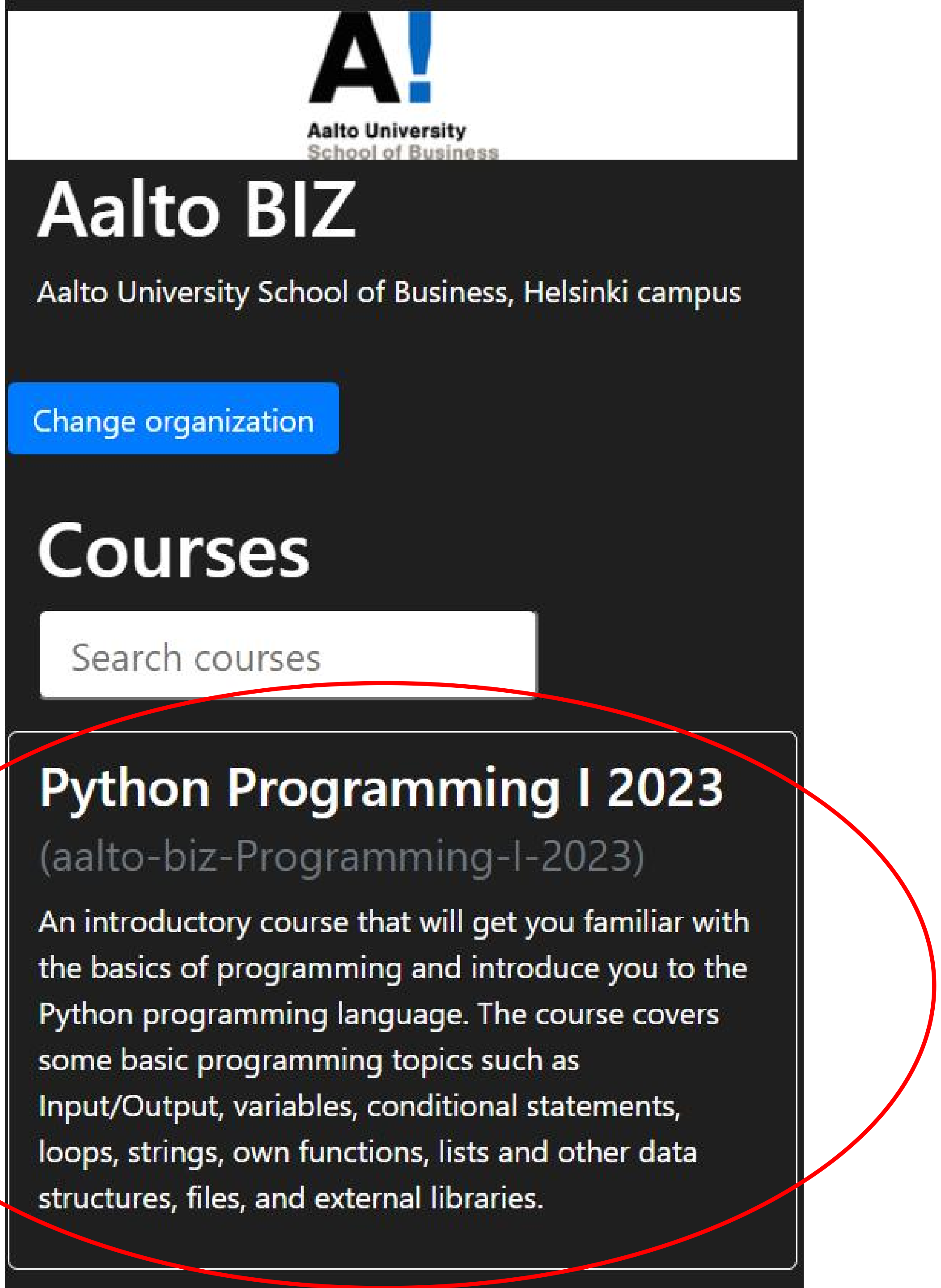
All organizations

Search organizations

Aalto BIZ (aalto-biz)

Aalto University School of Business, Helsinki campus

Next, search 2023 and select the course **Python Programming I 2023** (aalto-biz-Programming-I-2023)



Aalto BIZ

Aalto University School of Business, Helsinki campus

Courses

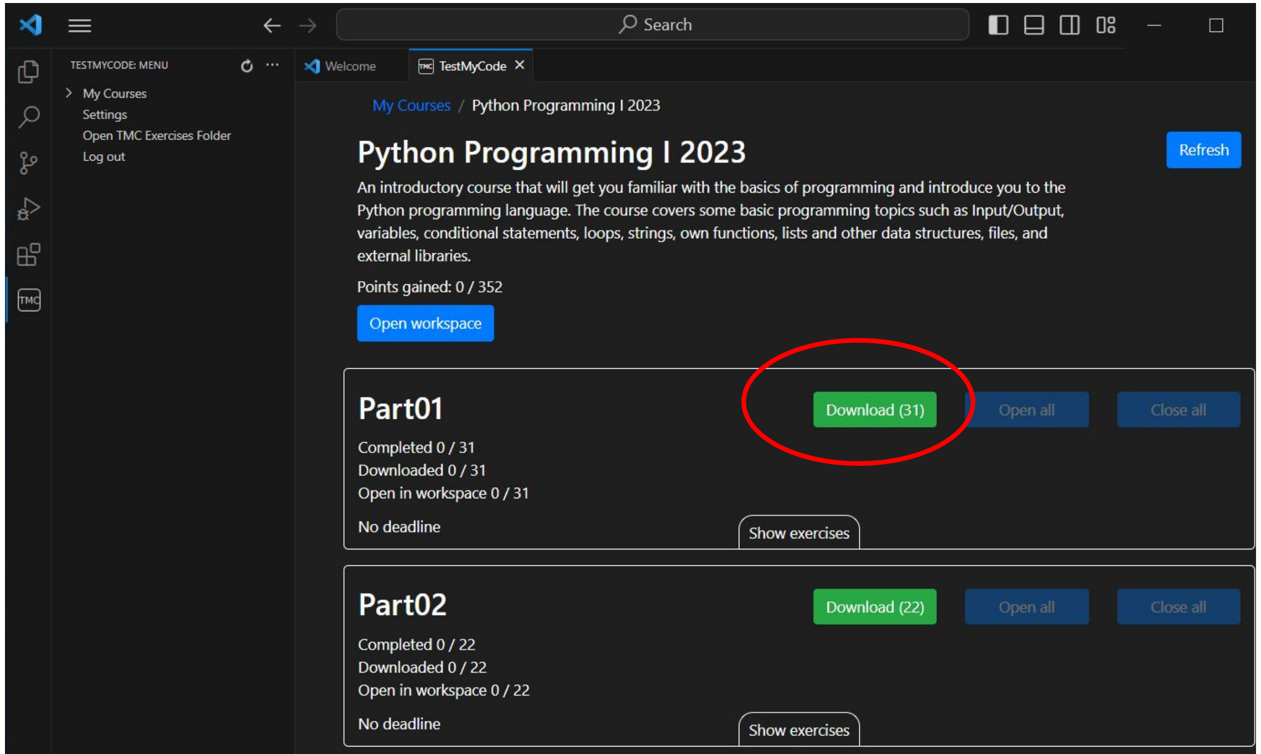
Python Programming I 2023

(aalto-biz-Programming-I-2023)

An introductory course that will get you familiar with the basics of programming and introduce you to the Python programming language. The course covers some basic programming topics such as Input/Output, variables, conditional statements, loops, strings, own functions, lists and other data structures, files, and external libraries.

Step 5: Downloading exercises

- Once you have selected the Python Programming I 2023 course, click the course title in the "My Courses" view.
  You will land into the course details page. Click on the **Download** button for **Part01**, which will start downloading the first part's exercises (the instructions on the MOOC page advise to download Part04, but download Part01 first)



My Courses / Python Programming I 2023

Python Programming I 2023

An introductory course that will get you familiar with the basics of programming and introduce you to the Python programming language. The course covers some basic programming topics such as Input/Output, variables, conditional statements, loops, strings, own functions, lists and other data structures, files, and external libraries.

Points gained: 0 / 352

Part01

Completed 0 / 31
Downloaded 0 / 31
Open in workspace 0 / 31

No deadline

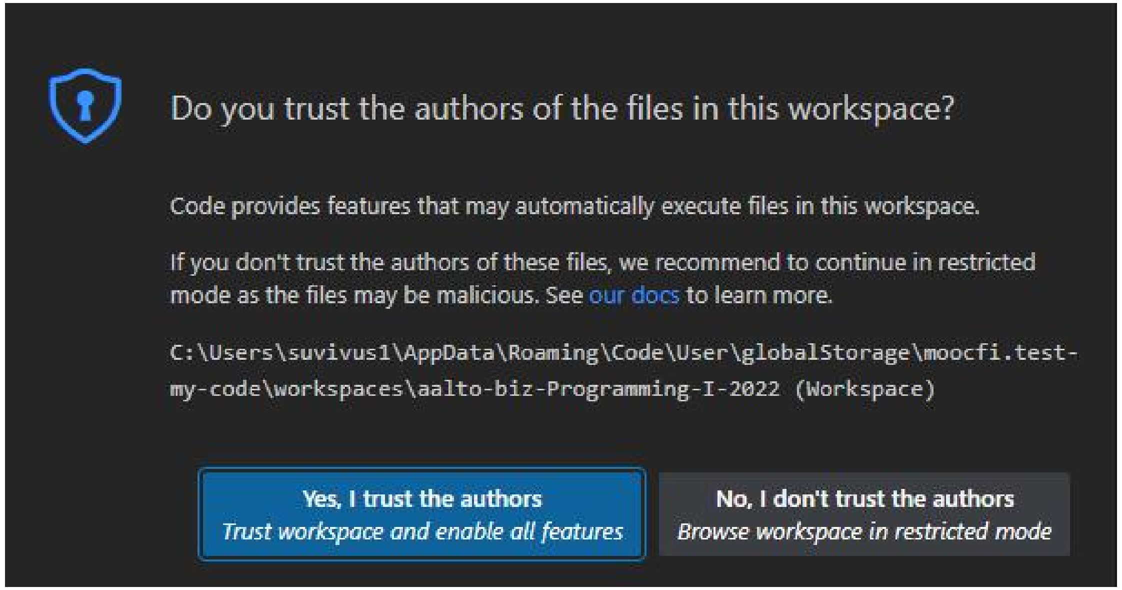
Part02

Completed 0 / 22
Downloaded 0 / 22
Open in workspace 0 / 22

No deadline

Downloading exercises (continued)

- You can follow the download process by checking the **Downloaded** counter or via the notification at bottom right, wait for the download process to finish before proceeding with the exercises.
- Note, downloading takes some time, and you may also see the following popup message – you can answer "Yes, I trust the authors".



Do you trust the authors of the files in this workspace?

Code provides features that may automatically execute files in this workspace.

If you don't trust the authors of these files, we recommend to continue in restricted mode as the files may be malicious. See our docs to learn more.

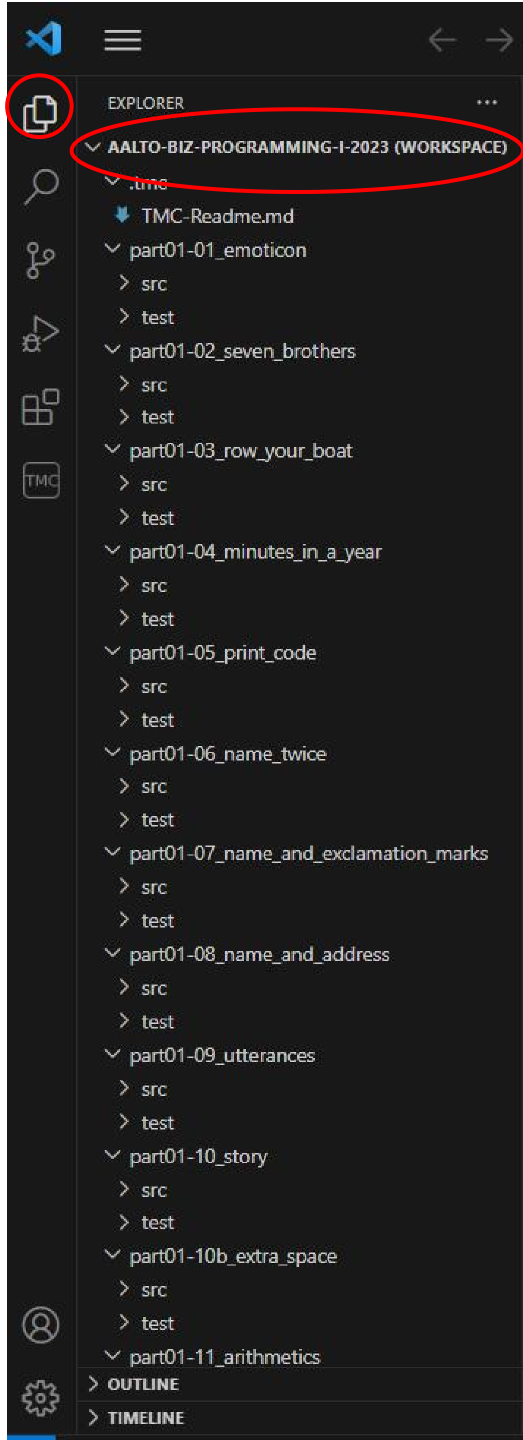
```
C:\Users\suvivus1\AppData\Roaming\Code\User\globalStorage\moocfi.test-my-code\workspaces\aalto-biz-Programming-I-2022 (Workspace)
```

Yes, I trust the authors
Trust workspace and enable all features
No, I don't trust the authors
Browse workspace in restricted mode

Downloading exercises (continued)

- When the download finishes, you should see the first part’s 30 exercises under the **AALTO-BIZ-PROGRAMMING-I-2023** course workspace ->

- *In case the exercises are not shown after you have waited a couple of minutes, close VS Code and open it again, and try downloading the exercises again.*

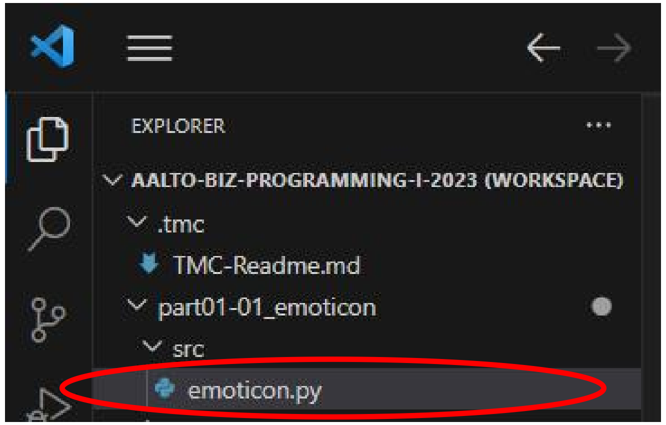


```
AALTO-BIZ-PROGRAMMING-I-2023 (WORKSPACE)
  .tmc
    TMC-Readme.md
  part01-01_emoticon
    src
    test
  part01-02_seven_brothers
    src
    test
  part01-03_row_your_boat
    src
    test
  part01-04_minutes_in_a_year
    src
    test
  part01-05_print_code
    src
    test
  part01-06_name_twice
    src
    test
  part01-07_name_and_exclamation_marks
    src
    test
  part01-08_name_and_address
    src
    test
  part01-09_utterances
    src
    test
  part01-10_story
    src
    test
  part01-10b_extra_space
    src
    test
  part01-11_arithmetics
```

Step 6: Get started with the exercises

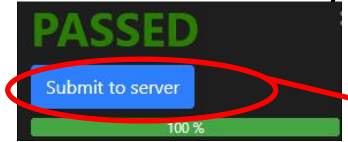
- All **course material and detailed instructions to the exercise tasks can be found in this page:** **https://programming-23.mooc.fi/**
- **The first exercise is under Part-1 (Emoticon) at:**
  - https://programming-23.mooc.fi/part-1/1-getting-started
- A single part's exercises are available at first, and the other parts become available for download once you have completed 70% of the exercises of the previous part. However, it is good to aim for 100% as the skills you learn might be needed in subsequent exercises.
- It is advisable to conduct the exercises in sprints (to aid memorizing)
- Group discussions with peers are also encouraged, but try to solve the exercises first yourself! It is the only way to learn coding.

Step 7: Doing the coding exercises

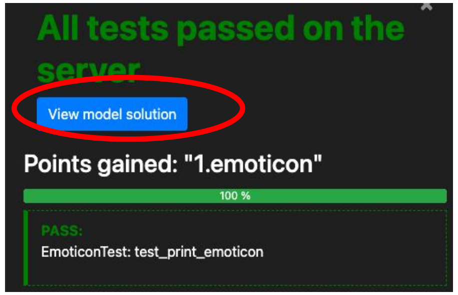


AALTO-BIZ-PROGRAMMING-I-2023 (WORKSPACE)
- .tmc
  - TMC-Readme.md
- part01-01_emoticon
  - src
    - emoticon.py

- **Open *VS Code* and the programming course**, if not open already
- **In the sidebar on the left…**
  - – **expand** with **>** the first exercise package (part01-01_emoticon), if not expanded
    - *Exercise_name* ➤ src ➤ *Exercise_name*.py
  - – Click the ***Exercise_name.py*** file to open the exercise template on the main window on right
- **In the main window…**
  - – fill in the exercise template with your answers (find task instructions at https://programming-23.mooc.fi)
  - – run your program with ▷ to make sure it works, check the error messages in bottom of screen if not
  - – submit the exercise to the TMC server to be checked with the eye icon TMC
  - – click 'Submit to server' if all TMC tests pass
    - – *note that small deviations in text may matter!*
  - – you can view the model solution and compare it to your own solution to learn more! There are often various ways to program the task ☺



PASSED



All tests passed on the server

Points gained: "1.emoticon"

PASS:
EmoticonTest: test_print_emoticon

- You can check your points earned from https://tmc.mooc.fi/ and clicking your account email on top right
- **Continue to the next exercise and repeat the above**

General tips

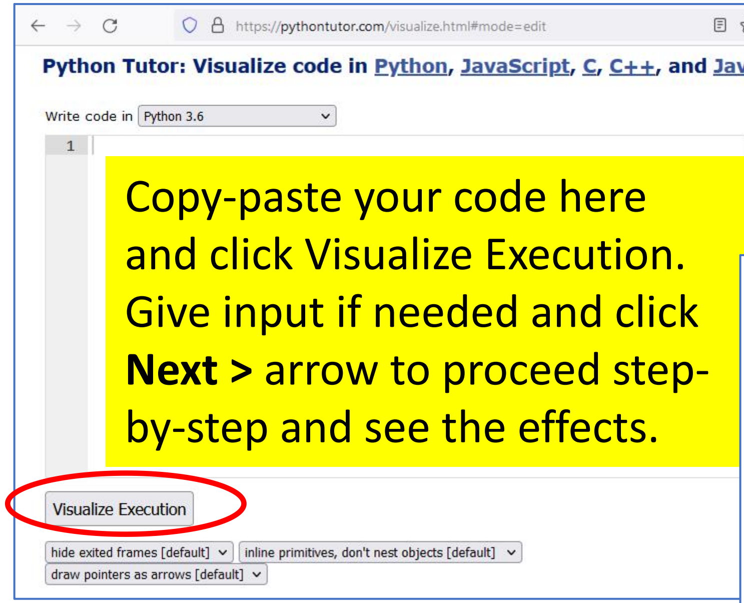
- Keep the code as clean as you can
  - use proper indentation! This will be checked when running the program.
  - use descriptive names for variables, functions, everything
  - keep all functions short, including main
  - write functions that only do one thing
  - remove all copy-pasted code by refactoring* code (or don’t copy-paste code in the first place!)
  - replace “bad” and unclear code with clean, easy to read code

* Refactoring means to rewrite existing source code in order to improve its readability, reusability or structure without affecting its meaning or behavior.

3. Troubleshooting and getting help during the course

Visualize your code to understand how it works or to find out bugs!

LINK: https://pythontutor.com/visualize.html#mode=edit



https://pythontutor.com/visualize.html#mode=edit

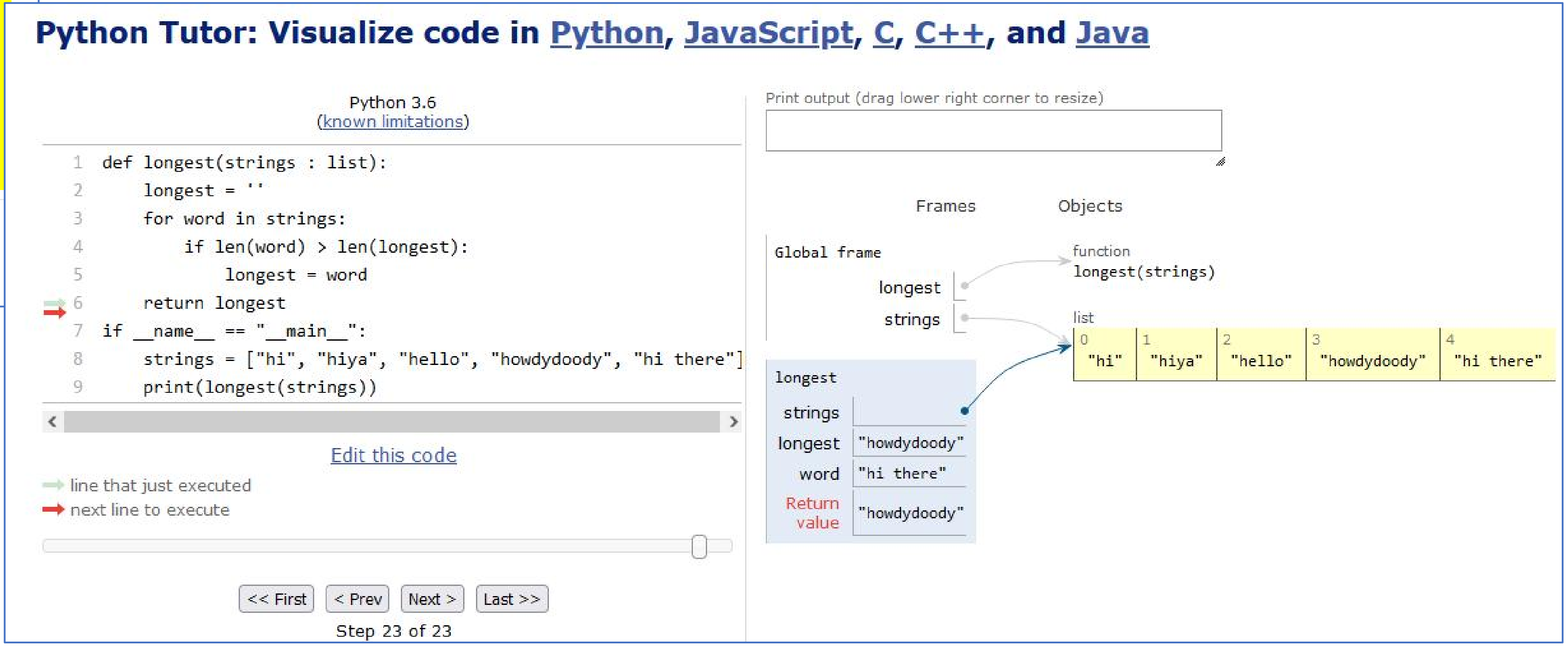
Python Tutor: Visualize code in Python, JavaScript, C, C++, and Java

Write code in Python 3.6

Copy-paste your code here and click Visualize Execution. Give input if needed and click **Next >** arrow to proceed step-by-step and see the effects.

Visualize Execution

hide exited frames [default] | inline primitives, don't nest objects [default] | draw pointers as arrows [default]



Python Tutor: Visualize code in Python, JavaScript, C, C++, and Java

Python 3.6
(known limitations)

```
def longest(strings : list):
    longest = ''
    for word in strings:
        if len(word) > len(longest):
            longest = word
    return longest
if __name__ == "__main__":
    strings = ["hi", "hiya", "hello", "howdydoody", "hi there"]
    print(longest(strings))
```

Edit this code

line that just executed
next line to execute

<< First | < Prev | Next > | Last >>

Step 23 of 23

Print output (drag lower right corner to resize)

Frames | Objects

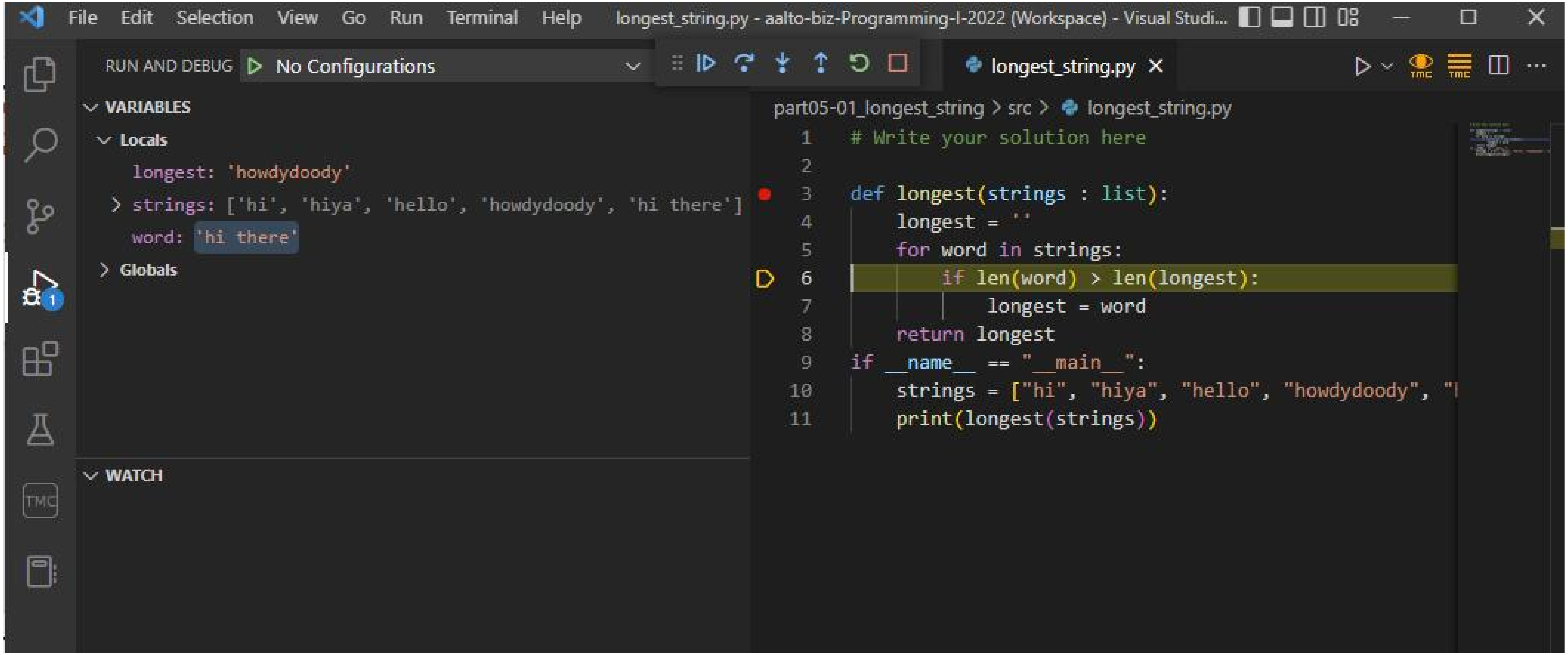
Global frame: longest → function longest(strings); strings → list

list: 0 "hi", 1 "hiya", 2 "hello", 3 "howdydoody", 4 "hi there"

longest

strings	
longest	"howdydoody"
word	"hi there"
Return value	"howdydoody"

VS Code has also a built-in debugger that can be used (its use is advised in Part 04*)


File
Edit
Selection
View
Go
Run
Terminal
Help
longest_string.py - aalto-biz-Programming-I-2022 (Workspace) - Visual Studi...
RUN AND DEBUG
No Configurations
longest_string.py
part05-01_longest_string > src > longest_string.py
VARIABLES
Locals
Globals
WATCH
TMC

```
longest: 'howdydoody'
strings: ['hi', 'hiya', 'hello', 'howdydoody', 'hi there']
word: 'hi there'
```

```python
# Write your solution here

def longest(strings : list):
    longest = ''
    for word in strings:
        if len(word) > len(longest):
            longest = word
    return longest
if __name__ == "__main__":
    strings = ["hi", "hiya", "hello", "howdydoody", "
    print(longest(strings))
```

* https://programming-23.mooc.fi/part-4/1-vscode, “The built-in debugger”

Use the course's Zulip chat forum for programming related issues and installation problems

Link to Zulip chat forum: **https://pythonmooc23.zulip.aalto.fi/**

Also, desktop and mobile applications for Zulip are available. You can also allow notifications from the discussions.

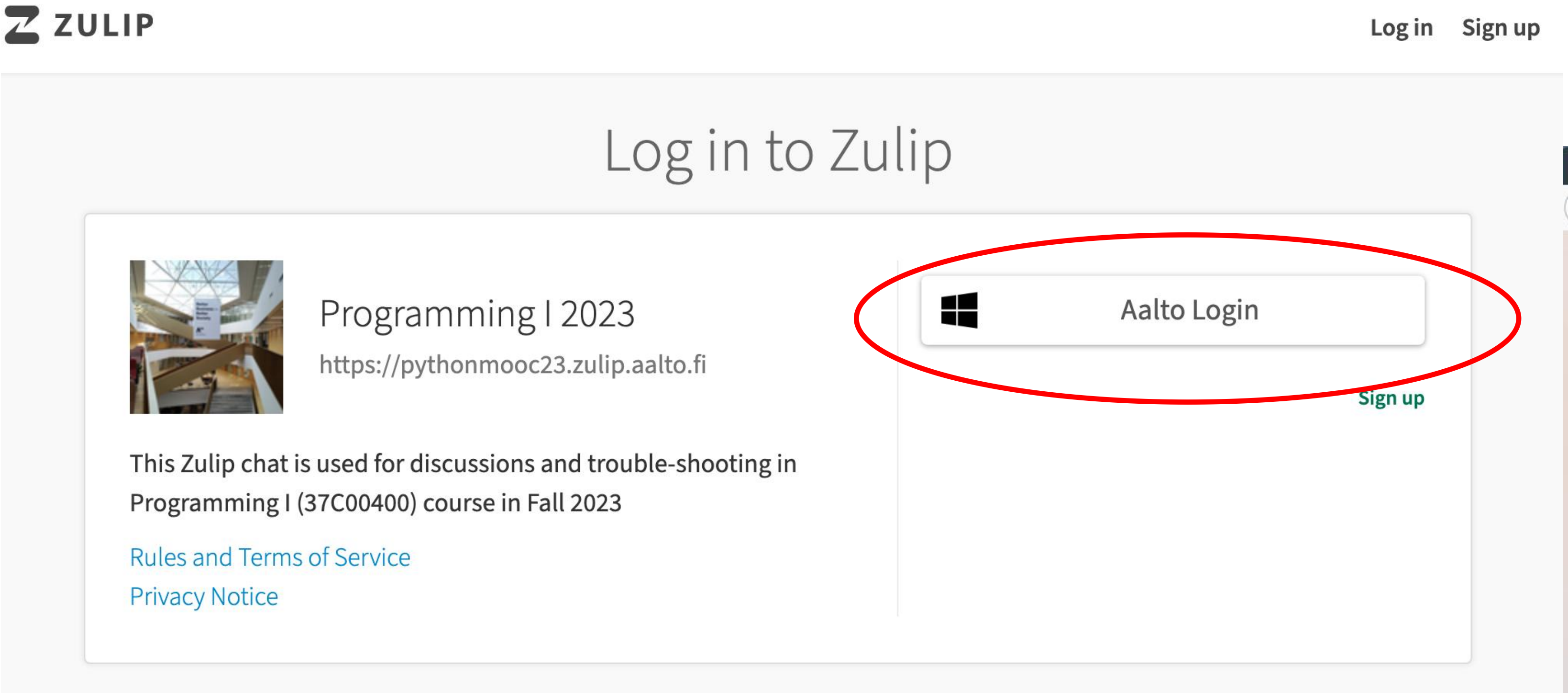
You may change your visible name to a nickname, but the teachers can see the emails of the participants.

Students are encouraged to help each other in the Discussion Forum!

Active participation in terms of peer support in the discussion forum is taken into account in the final grade, e.g. when the grade is close to the next one.

You may also use the general MOOC's support channel in Discord (see link to it at https://programming-23.mooc.fi/support-and-assistance), search in there is often fruitful.

Instructions for signing in the to course chat forum at https://pythonmooc23.zulip.aalto.fi/ (1/2)



Log in to Zulip

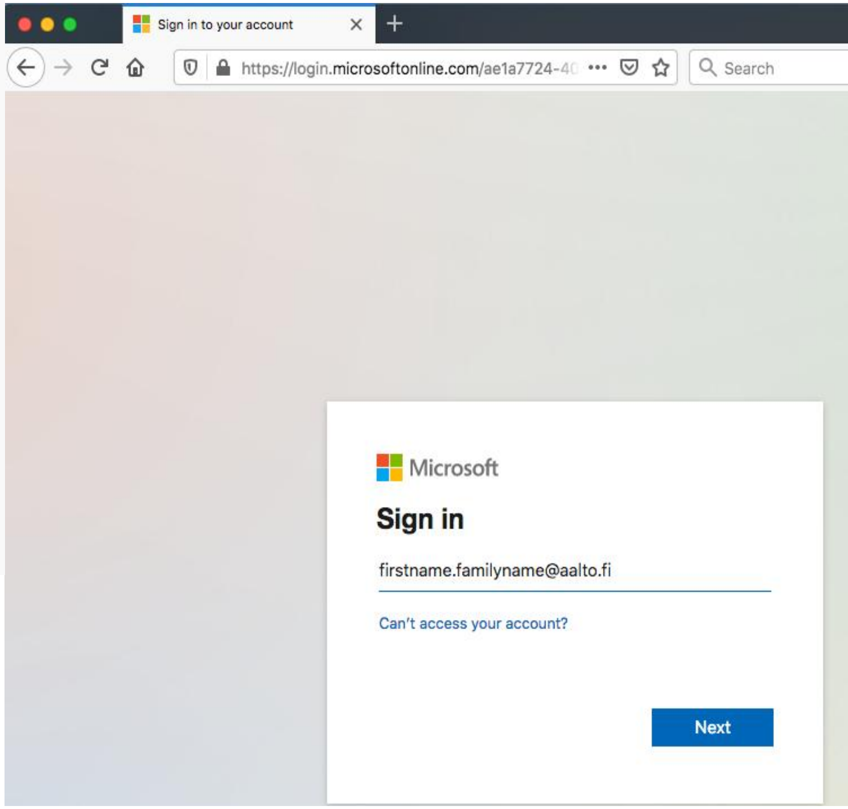
Programming I 2023

https://pythonmooc23.zulip.aalto.fi

This Zulip chat is used for discussions and trouble-shooting in Programming I (37C00400) course in Fall 2023

Rules and Terms of Service

Privacy Notice

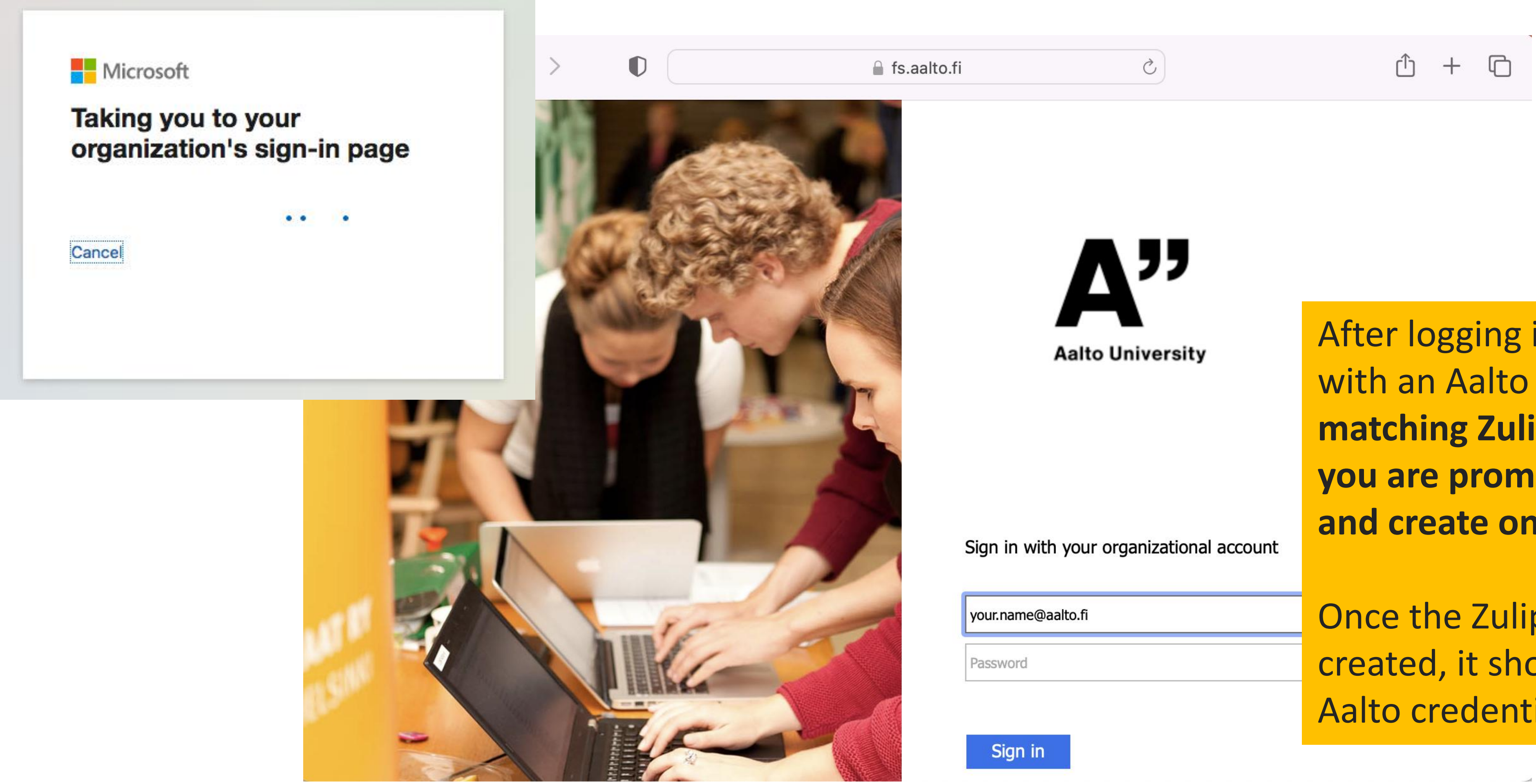


Sign in

firstname.familyname@aalto.fi

Can't access your account?

Instructions for signing in to course Chat forum at https://pythonmooc23.zulip.aalto.fi/ (2/2)



Taking you to your organization's sign-in page

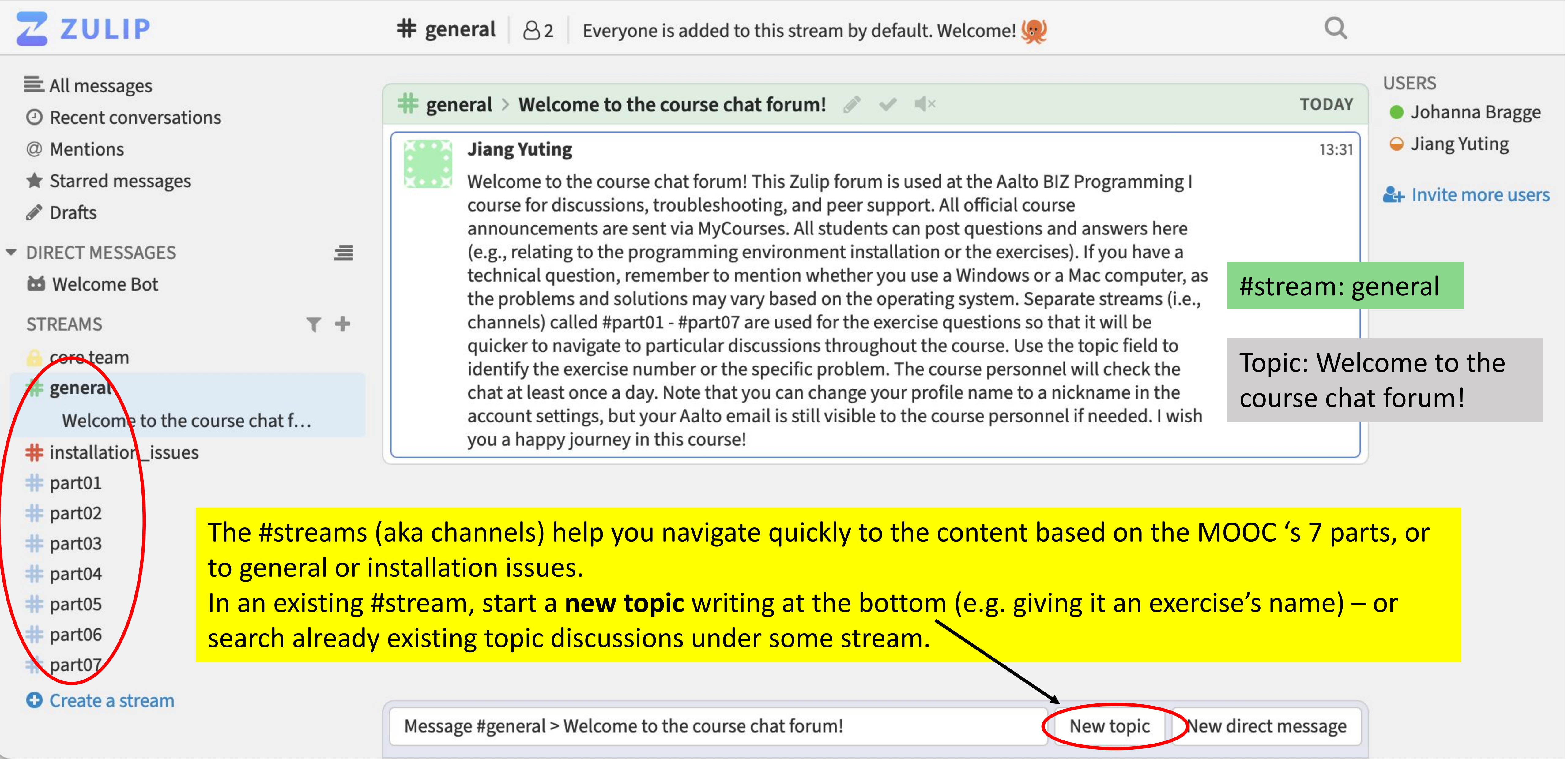
Sign in with your organizational account

your.name@aalto.fi

After logging in for the **first time** with an Aalto account, **if no matching Zulip account was found, you are prompted to "Register" and create one.**

Once the Zulip account has been created, it should be linked to your Aalto credentials.

Chat forum https://pythonmooc23.zulip.aalto.fi/



Jiang Yuting 13:31

Welcome to the course chat forum! This Zulip forum is used at the Aalto BIZ Programming I course for discussions, troubleshooting, and peer support. All official course announcements are sent via MyCourses. All students can post questions and answers here (e.g., relating to the programming environment installation or the exercises). If you have a technical question, remember to mention whether you use a Windows or a Mac computer, as the problems and solutions may vary based on the operating system. Separate streams (i.e., channels) called #part01 - #part07 are used for the exercise questions so that it will be quicker to navigate to particular discussions throughout the course. Use the topic field to identify the exercise number or the specific problem. The course personnel will check the chat at least once a day. Note that you can change your profile name to a nickname in the account settings, but your Aalto email is still visible to the course personnel if needed. I wish you a happy journey in this course!

#stream: general

Topic: Welcome to the course chat forum!

The #streams (aka channels) help you navigate quickly to the content based on the MOOC 's 7 parts, or to general or installation issues.
In an existing #stream, start a **new topic** writing at the bottom (e.g. giving it an exercise's name) – or search already existing topic discussions under some stream.

Frequent errors in code

- Use correct capitalization
  - variable names are case-sensitive
- Remember the colon ":"
- Close parentheses ( ), brackets { } [ ] and quotation marks " "
- Check also proper indentation, or you will be prompted to correct it when running the exercise.
- Check whether the output is correct. The python interpreter does not report an error does not mean that the program is error-free.

Advice when stuck or running an infinite loop

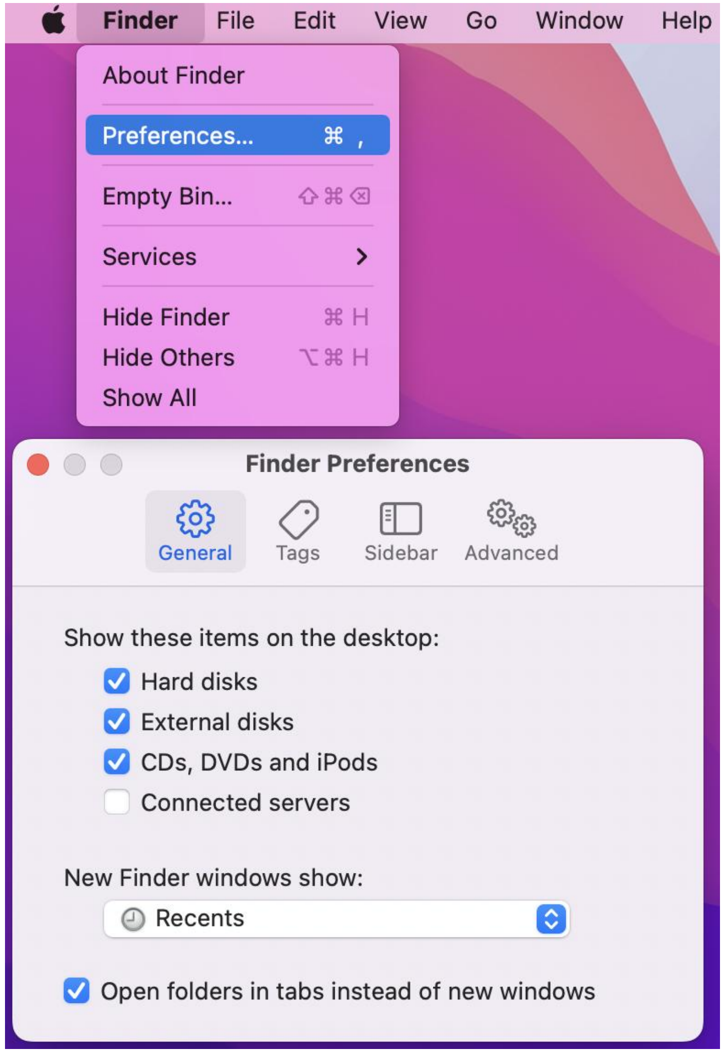
“Sometimes [in VS Code] it happens that some program is left running in the background, maybe **waiting for input** from the user or stuck in an **infinite loop**, without you realising it. You might only notice this when trying to execute the next program, which simply will not run because the previous execution is still hogging up resources. A quick fix is pressing on the keys **Control+C** at the same time, which stops the execution of any ongoing process. The next program should now run fine.”

https://programming-23.mooc.fi/part-4/1-vscode, “Executing code”

Troubleshooting hints in Macs

- If you need to check installations on your Mac’s hard drive and the disk is hidden (by default), open the Finder > open finder Preferences (or Settings in newer OS versions) from the upper menu, and **tick Hard disks**

- In case you are not seeing a folder or file in some program folder, click **Command + Shift + period** (piste .) to see also possible hidden files



Finder Preferences

Show these items on the desktop:

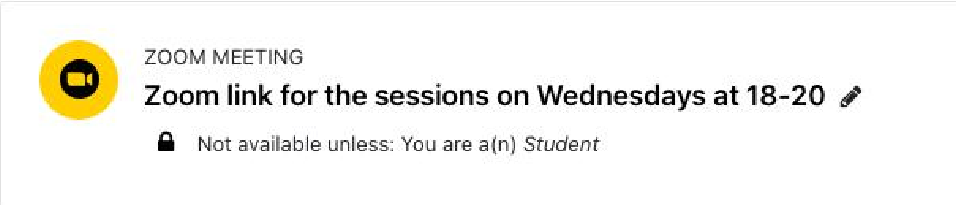
☑ Hard disks
☑ External disks
☑ CDs, DVDs and iPods
☐ Connected servers

New Finder windows show: Recents

☑ Open folders in tabs instead of new windows

Online help sessions on Wednesdays at 18.00 o’clock start at October 25!

- Recurring zoom link for online sessions (link also at course main page): https://aalto.zoom.us/j/63264231398?pwd=MXlpR3ZjZEt2R3kvNnJzeHNqUUhpUT09


ZOOM MEETING

Zoom link for the sessions on Wednesdays at 18-20

Not available unless: You are a(n) *Student*

- Responsible teacher of course: **Yuting Jiang** (yuting.jiang@aalto.fi)
  - use the subject **“Programming-I” in emails**

- Supporting teacher: Johanna Bragge (johanna.bragge@aalto.fi)
  - e.g. Mac OS related tech issues

Main takeaways

- Please make sure to add your **student ID *number*** in the TMC system in the **Organizational Identifier** field.
- Register to and select in VS Code the **Aalto BIZ course instance (‘/*aalto-biz-Programming-I-2023*’)**
- Submit the exercises right from the beginning (Part01) using the **VS Code editor’s TMC extension**, so that we can see your points earned!
  (in Aalto we can’t see them if you conduct the exercises in the browser, which is possible for parts 1-3 in the general MOOC web page).
- Start doing the **exercises regularly** – they take time!
  - 2 deadlines for them: November 15 (parts 1-4) and December 11 (parts 5-7)
    - You get 90% of points from parts 1-4 if you submit them late between November 16-December 11.

MOOC developers’ advice on the exercises

> “Becoming a proficient programmer **requires a lot of practice**, sometimes even quite mechanical practice. It also involves developing problem solving skills and applying intuition. This is why there are a lot of exercises of different kinds on this course. Some of them ask you to quite straightforwardly apply what you have learnt in the material, but some of them are intentionally more challenging and open-ended.
>
> **The exercises are not in any specific order of difficulty.** Each section usually introduces some new programming concepts, and these are then practised with both simpler and more complicated exercises. **If you come across an exercise that feels too difficult, move on to the next one.** You can always come back to the more difficult exercises if you have time later.
>
> When the going inevitably gets tough, a word of consolation: a task that seems impossibly difficult this week will likely feel rather easy in about four weeks' time.”

Excerpt from: https://programming-23.mooc.fi/part-5/1-more-lists “About the exercises on this course”