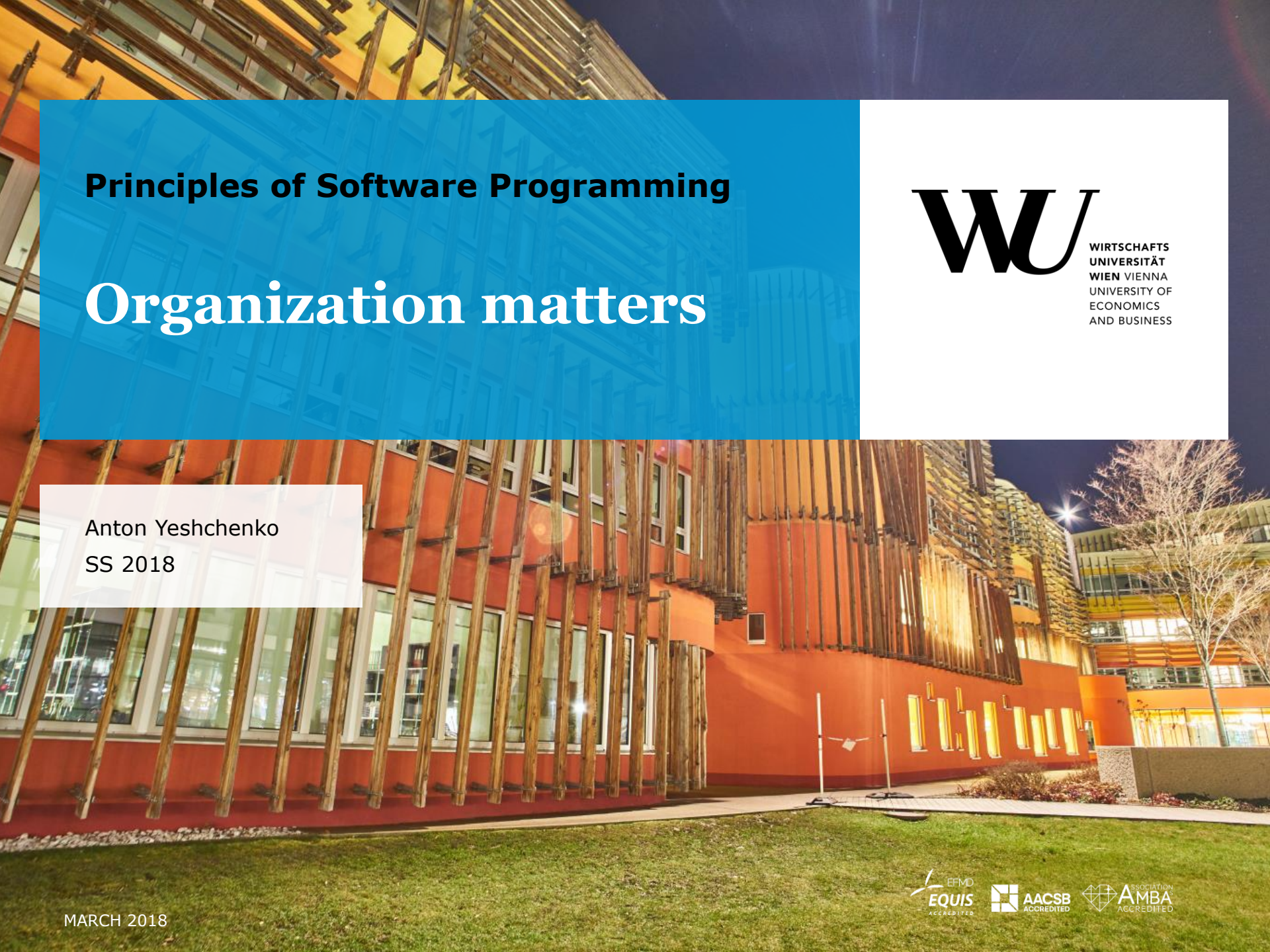


Principles of Software Programming

Organization matters

Anton Yeshchenko

SS 2018

WU WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

EFMD EQUIS ACCREDITED · AACSB ACCREDITED · ASSOCIATION AMBA ACCREDITED

MARCH 2018

About me

WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

Anton Yeshchenko, MSc.

PhD student / Researcher at WU

Researcher at department of Information Systems and Operations
Institute for Information Business

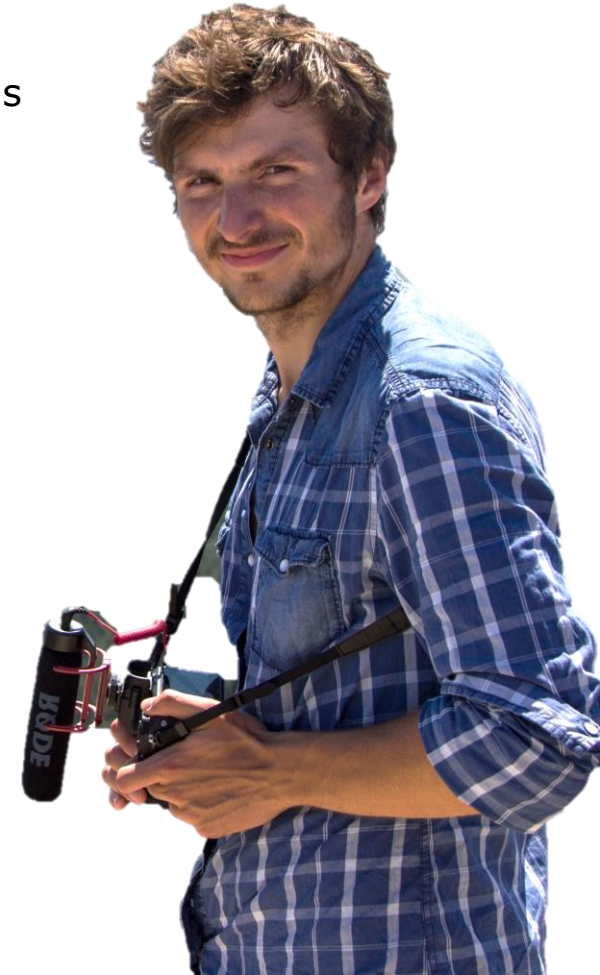
Building D2, 3rd Floor.
Email: **anton.yeshchenko at wu.ac.at**

Interests:

- Process mining
- Predictive monitoring
- NLP
- Machine learning

Hobbies:

- Filmmaking
- Triathlon
- Travelling journalism



About you?

1. Name
2. One hobby
3. Programming experience
4. Why here (in programming class)?

Technical goal - Exam

Running semester dates:

- **12.05**.2018 - 12:00 - 13:30
- **28.06**.2018 - 14:00 - 15:30

Knowledge goals

- Understand:
  - How computer **actually** works?
  - How **"program"** actually works in computer?
  - **How to think** in terms of **programmable solutions** about practical problems?

Knowledge goals

- Understand:
  - How computer **actually** works?
  - How **"program"** actually works in computer?
  - **How to think** in terms of **programmable solutions** about practical problems?

- Be able to:
  - **Implement** (design and code) **solutions** for problems
  - **Read** programs written by others
  - **Make** critical **judgements** on the quality of different *implementations*

Content of the course

- From *hardware* to *code (software)*
- Variables, data types
- Control flow: branching, iteration, functions
- Basic data structures, algorithms
- Object oriented programming:
  - Classes, objects
  - Inheritance, method overriding
- Information hiding
- Recursion

Extra (Wahlfach):

- Exceptions,
- Dynamic data structures,
- algorithms

PICTURE:HTTP://WADLER.BLOGSPOT.CO.AT/2010/06/WHY-UNDERGRADUATES-SHOULD-LEARN.HTML

Learning ways

8 + 1 lectures

(non bothering)
1. Quizzes

Learning ways

8 + 1 lectures

(non bothering)
1. Quizzes

(hopefully interesting)
2. Theory

Learning ways

8 + 1 lectures

(non bothering)
1. Quizzes

(hopefully interesting)
2. Theory

(collaborative and engaging)
3. In class exercises

Learning ways

8 + 1 lectures

(non bothering)
1. Quizzes

(hopefully interesting)
2. Theory

(collaborative and engaging)
3. In class exercises

(not ignored?, proactive)
4. Home (practical) exercises

Resources for the course

- Course page with
  - exam topics
  - lecture slides (uploaded after the lecture)
  - exercises
  - https://learn.wu.ac.at/dotlrn/classes/pool/6088.18s
- Alternativelly:
  - https://github.com/yesanton/Principles-of-Programming-6088-S18-Course-materials

- Python notebook created for you
  - https://programming.ai.wu.ac.at/6088/notebooks/$STUDENTID

Resources on internet (to learn python)

- https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/
- https://www.coursera.org/learn/learn-to-program
- https://imgur.com/gallery/3wSHJ
- http://www.python-course.eu/python3_course.php
- http://docs.python-guide.org/en/latest/
- http://diveintopython.net
- https://pythontips.com/2016/02/27/learning-python-for-data-science/
- https://talkpython.fm
- https://github.com/jupyter/jupyter/wiki/A-gallery-of-interesting-Jupyter-Notebooks
- https://learnpython.wordpress.com
- http://www.pythontutor.com
- https://www.codecademy.com/learn/python
- http://interactivepython.org/courselib/static/thinkcspy/index.html
- https://github.com/gregmalcolm/python_koans