

Jupyter Notebooks as tools for interactive learning of Concepts in Structural Geology and efficient grading of exercises.

Jan Niederau (1), Florian Wellmann (2), Jannik Maersch (2,3), and Janos Urai (3)

(1) Institute for Applied Geophysics and Geothermal Energy, E.ON Energy Research Center, RWTH Aachen University, Germany , (2) Numerical Reservoir Engineering, Graduate School AICES, RWTH Aachen University, Germany, (3) Structural Geology, Tectonics and Geomechanics, Energy and Mineral Resources Group (EMR), RWTH Aachen University, Germany

Programming is increasingly recognised an important skill for geoscientists – however, the hurdle to jump into programming for students with little or no experience can be high. We present here teaching concepts on the basis of Jupyter notebooks that combine, in an intuitive way, formatted instruction text with code cells in a single environment. This integration allows for an exposure to programming on several levels: from a complete interactive presentation of content, where students require no or very limited programming experience, to highly complex geoscientific computations. We consider these notebooks therefore as an ideal medium to present computational content to students in the field of geosciences.

We show here how we use these notebooks to develop digital documents in Python for undergrad-students, who can then learn about basic concepts in structural geology via self-assessment. Such notebooks comprise concepts such as: stress tensor, strain ellipse, or the mohr circle.

Students can interactively change parameters, e.g. by using sliders and immediately see the results. They can further experiment and extend the notebook by writing their own code within the notebook. Jupyter Notebooks for teaching purposes can be provided ready-to-use via online services. That is, students do not need to install additional software on their devices in order to work with the notebooks.

We also use Jupyter Notebooks for automatic grading of programming assignments in multiple lectures. An implemented workflow facilitates the generation, distribution of assignments, as well as the final grading. Compared to previous grading methods with a high percentage of repetitive manual grading, the implemented workflow proves to be much more time efficient.