Context

This is a coding course for 3rd year Information Management students, aimed at improving student's programming skills and improving their knowledge on the fundamentals of programming.

Interactive Document is used to get students engaged with the material before class. The instructor uses a flipped classroom approach and adds inline questions and discussions to the study material. The annotations possibilities give students the opportunity to ask questions related to exercises.

In the real world developers don't play solo, they usually work on teams. Hence, the teacher aimed to activate students, stimulate collaboration and foster knowledge sharing between students.

Constructive alignment

Learning objectives

- Students are able to build good and solid software
- Students are able to understand and be comfortable with CLI commands

Learning activities

Students learn about coding principles and good practices every developer should know. This course is mainly based on exercises and laboratories that will need to be completed both inside and outside the classroom. Learners are asked to read through the study material before class and then answer the coding exercises in-class on their laptops. When executing the command line exercises students get the opportunity to collaborate with their peers.

These learning activities address the following levels of Bloom's Taxonomy:

- **Remember** - Memorising coding principles and good practices
- **Apply** - Executing code exercises
- **Create** - Programming and representing code in a graphical way using UML
Assessment of learning outcomes

- In order to assess whether students understand the coding principles an individual assignment and exam is graded by the instructor. The FeedbackFruits activity is not graded as it is only set up to flip the classroom.
- Class participation is evaluated based on the quality (not quantity) of participation in class.
- The instructor was able to evaluate the level of interactivity and collaboration by reviewing the comments and questions raised in the Interactive Document.

Quote from the instructor

“Students were not explicitly encouraged to reply and help each other out in the learning process, but fortunately this happened organically.”

Notable outcomes

- Students were more engaged compared to prior years, the formative activity had a 100% participation rate.
- Students enjoyed working with the tool and would like to see the application in other courses as well.
- It was accepted that the instructor did not have time to answer all the questions in the document. Fortunately, the students helped each other out by themselves.

The role of the instructor

- Creating inline exercises in the Interactive Document.
- Checking the analytics table to evaluate students’ progress and specifically whether students worked together (commented on each other)
- The most important factor is to give clear instructions of what is expected from the students.
- Next to that, it is important to make online interactions safe and encouraging – ask for constructive peer feedback only and remind them there is no such thing as stupid questions.
- The instructor has the ability to clarify instructions that are not clear by annotating and commenting.

Added value of technology

- Helps with the monitoring of student activity and fosters collaboration in an otherwise individual task (coding).

Possible variation

Using a summative Interactive Document that includes practice questions could allow instructors to assess participation, collaboration and correctness of answers for a final grade.

To help students to improve their coding, it could be interesting to add a Peer Review activity so students can learn from each other.