We, the NumFOCUS Code of Conduct Enforcement Committee, issue a public apology to Jeremy Howard for our handling of the JupyterCon 2020 reports. We should have done better. We thank you for sharing your experience and we will use it to improve our policies going forward.

We acknowledge that it was an extremely stressful experience, being summoned to an interview with several members of a committee, after a week had passed, and without knowing the nature of the complaint. We apologize for causing this stress and will work to improve our process to avoid this from happening in the future.

To clarify a crucial miscommunication that we take responsibility for: At the time of the interview, the committee had not determined that there was a violation of the code of conduct, only that there were two complaints filed and being examined. We apologize for not communicating that clearly from the beginning. We have not recommended any enforcement actions. We had asked to postpone the posting of the talk to the JupyterCon shared space until the complaints are resolved. We realize now that we used overly charged language and miscommunicated the stage of the investigation when discussing the complaints, i.e. saying a violation occurred. We should have been clearer saying multiple complaints have been made and the alleged violation investigation had not been resolved.

Data structures in Python don't actually contain objects. They references to objects (aka "pointers").
In this episode, I worked on the form that will send invites to users for the new social network app that I'm building. We built the view, the form, and the tests and wired a button to the new view.

The first thing that we do was talk through the new changes since the last stream. After discussing the progress, I took some time to cover the expected budget for the application to get it to an MVP.

Once we covered the budget, I talked about different strategies for sending invite emails and the tradeoffs between sending email in a request and response cycle versus using background workers.

You probably have heard of the bestselling Python book, *Automate the Boring Stuff with Python.* What are the next steps after starting to dabble in the Python basics? Maybe you've completed some tutorials, created a few scripts, and automated repetitive tasks in your life. This week on the show, we have author Al Sweigart to talk about his new book, *Beyond the Basic Stuff with Python: Best Practices for Writing Clean Code.*

A dictionary in Python is a collection of items that stores data as key-value pairs. In Python 3.7 and later versions, dictionaries are sorted by the order of item insertion. In earlier versions, they were unordered.

Let's have a look at how we can sort a dictionary on basis of the values they contain.