In this passage, the writer talks about how important it is for programmers to review each other's code to learn and improve. They're going to look at David Gilbert's SerialDate class and make suggestions for how it could be better, not to show off or be mean, but to help everyone get better at coding. They appreciate Gilbert sharing his code with others, showing how teamwork is key in coding. By going through this process, the writer hopes to make it easier for people to understand and fix common problems in Java programming, like dealing with dates.

First, Make It Work

The passage talks about testing and fixing code. It starts by saying that when they checked the unit tests using Clover, they found some areas that weren't tested well enough. So, they made more tests to cover those areas better, even though they weren't activated at first. These extra tests helped find problems like case sensitivity and features that weren't supported. Fixing these issues was pretty simple. They also found mistakes in how the code was written by looking at where the tests failed. This meant they had to keep tweaking the code until everything worked smoothly. In the end, they say that after all these tests and fixes, SerialDate works correctly.

Then Make It Right

The passage talks about how to make software easier to understand and work with. It says it's important to keep code clear and simple by using good formatting and naming. It suggests using tools like Javadoc to help with this. It also says that names for things in the code should make sense and describe what they do. It's important to keep names consistent throughout the project. The passage also mentions ways to improve code structure, like using design patterns, and emphasizes the importance of testing to make sure everything works correctly.