Habits of Efficient Developers

Some time posts:

[01:45](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=105s) : 1. Focus Interruptions: 15-20 minutes to get back a. Own control: Email notifications b. Not under your control XXXX [02:30](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=150s) : Email notifications are the worst! [03:08](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=188s) : 1a. Disable ALL notifications [04:58](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=298s) : 1b. Pair Programming 05:AA secs: 1c. Sleep over coffee (?) [05:50](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=350s) : 1d. One at a time [06:31](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=391s) : 2. Master your IDE [06:57](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=417s) : 2a. Functionality 2b. Shortcuts [07:47](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=467s) : 3. No menial work [08:55](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=535s) : 3a. Write programs (for yourself) [09:30](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=570s) : Bash [09:57](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=597s) : Demo (Bash) [10:39](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=639s) : Format JSON content with 'jq' [14:59](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=899s) : XKCD 1205 [15:04](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=904s) : 3b. Avoid GUIs [15:38](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=938s) : 3c. Automate testing For every new bug: First add a test to reproduce it (for regression testing) [17:10](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1030s) : 3d. Repeatable Developer Environment [17:42](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1062s) : 'docker-compose up' [18:19](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1099s) : Docker compose file example with three Docker containers Also allow easy experimentation with newer/other versions of development tools, etc. [19:22](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1162s) : 4. Feedback Find the shortest and tightest feedback loop [19:56](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1196s) : 4a. Test-driven development (TDD) [24:45](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1485s) : 4b. REPL For exploring unknown systems and APIs [25:03](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1503s) : REPL demo (using a remote REPL server, Clojure) [29:24](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1764s) : End of REPL demo. [29:31](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1771s) : 4c. Code reviews [29:58](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1798s) : Tweet: <twitter DOT com/iamdevloper/status/397664295875805184> "10 lines of code = 10 issues. 500 lines of code = "looks fine." Code reviews." [31:20](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1880s) : Asking the product owner to create a refactoring story! [31:50](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1910s) : 4c. Continuous Code reviews [33:05](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=1985s) : End, with a summary [33:27](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=2007s) : Never stop learning! [33:29](https://www.youtube.com/watch?v=9-cyC6O81Bk&t=2009s) : Presenter: twitter DOT com/DanLebrero DLebrero AT gmail DOT com danlebrero DOT com/ twitter DOT com/Akvo akvo DOT org/