

# Which Grade Does Your Data Science Code Get?

written by Melanie Remmels

last edited March 2, 2019



## TODO

Data science is already a complex topic from the ground up. How can this topic be programmatically implemented so that it does not require rocket science to understand its implementation? Maybe a similar topic could help answering this question:

- Are Data Scientists Software Developers? - Yes
- Do Data Scientist write code? - Yes
- Is Data Science Code easy to understand? - Sometimes
- Do approaches and tools exist to improve "classic" software code? - Yes
- Can the approaches for clean code be transferred to Data Science? - Let's see!

## Clean Code Developer

If you want to (re)use the code of another developer, you have to understand it. You hope that the other developer has structured his code well, that variables have unique and meaningful names, and that there are some comments which explain the purpose of the whole algorithm.

In other words, it is hoped that the developer has followed basic, unofficial programming rules. These rules are even more important when it comes to data science and machine learning.

The Clean Code Initiative [3] is about promoting more professional software development. In my opinion, this initiative should also be applied to machine learning, as it increases the comprehensibility of code and programming. Comprehensible code not only helps others, but also oneself, because it helps you to understand your own ideas expeditiously after a certain period of time.

## Available Grades

There are thousands of different programming paradigms and principles. The Clean Code initiative [3] has divided these into different ranks. In the following you get a little insight into each grade: TODO: Jeden einzelnen Gürtel beschreiben

**Grade 0 - Black:** You are interested in the topic, but you don't yet apply it in your everyday work

**Grade 1 - Red:** The absolute essentials. You should build a general attitude to write clean code. Easy principles like DRY (Don't repeat yourself!) and KISS (Keep it simple, stupid!) are applied. The basic rule behind it is: Leave the code cleaner than you found it.

**Grade 2 - Orange:**

**Grade 3 - Yellow:**

**Grade 4 - Green:**

**Grade 5 - Blue:**

**Grade 6 - White:**

As mentioned on [2]:

Clean Code Developer is not a status but a progression.

The idea is to internalize the values, not to learn the rules by heart. For this reason, someone who reached all grades should start again from the beginning.

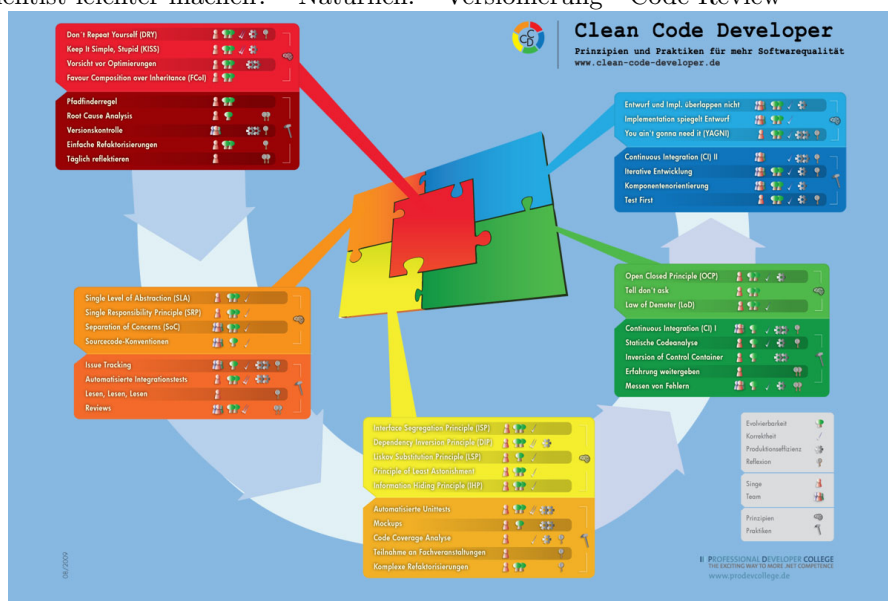
- Gürtel bauen aufeinander auf - Man sollte Gürtel nicht als Prüfung sehen, sondern versuchen sich immer weiter zu verbessern

- Zitat: "Wer einen weißen Gürtel hat, soll wieder von vorne anfangen"

## Wie kann man es trainieren

Katas Lesen wie andere vorgehen

Kann man noch weitere Sachen von der Software entwicklung übernehmen, die das Leben eines Data Scientist leichter machen? - Natürlich: - Versionierung - Code Review



[1] [2]

[3] For further information and training how to programm clean, have a look here: [4]. The examples are for "normal" code, not data science specific. But for a first hands-on they are helpful.

## References

- [1] S. Savage, "Data science and test driven development." <https://www.linkedin.com/pulse/data-science-test-driven-development-sam-savage/>, August 2015. Accessed on 2019-02-21.
- [2] R. Westphal and S. Lieser, "Clean code developer." <https://clean-code-developer.com/grades/>, 2015. Accessed on 2019-02-21.
- [3] R. Westphal and S. Lieser, "Clean code developer." <https://clean-code-developer.com/>, 2015. Accessed on 2019-02-14.
- [4] C. C. D. School, "Coding dojo - clean code developer school." <https://ccd-school.de/coding-dojo/>, 2018. Accessed on 2019-02-14.