

**git\_comments:****git\_commits:**

1. **summary:** [FLINK-19671][codestyle] Revert .editorconfig change violating our coding style  
**message:** [FLINK-19671][codestyle] Revert .editorconfig change violating our coding style > Rules about breaking the long lines: > >  
 Break the argument list or chain of calls if the line exceeds limit or earlier if you believe that the breaking would improve the code readability > If you break the line then each argument/call should have a separate line, including the first one > Each new line should have one extra indentation (or two for a function declaration) relative to the line of the parent function name or the called entity

**github\_issues:****github\_issues\_comments:****github\_pulls:****github\_pulls\_comments:****github\_pulls\_reviews:****jira\_issues:**

1. **summary:** Update EditorConfig file to be useful  
**description:** We should update our `{.editorconfig}` file to format Java code according to a style that passes our checkstyle rules and also applies our import ordering. This will greatly simplify development because developers can just "re-format code" in IntelliJ and be done with it. See the ML discussion for more background:  
<https://lists.apache.org/thread.html/r481bb622410718cd454d251b194c7c1ad358e5d861fdacc9a4be3b5b%40%3Cdev.flink.apache.org%3E>

**jira\_issues\_comments:**

1. master: ea599af7f871a7b79c4428db0ce87512a76f88c2 release-1.11: 586a56abb6b06c26a6aa821a4644c23afc11181a release-1.10: c09e959cf55c549ca4a3673f72deeb12a34e12f5
2. Hi [~aljoscha], I found the continuation indentation is set to 8, which will result in 2 extra tabs in new line when breaking long lines. This breaks the rule: `{code:java}` Each new line should have one extra indentation (or two for a function declaration) relative to the line of the parent function name or the called entity `{code}`
3. I have used 8 for a long time because we don't have any style enforcement in our CI and I don't think it makes sense to manually review code for style in PR reviews. Regarding 4 vs. 8, it's a bit tricky since some constructs in the Flink code base use 4 as continuation indent and some use 8. For example, parameters in a method definition use 8 but a lot of people use only 4 when calling a method. I went with 8 because it's uniform and also because the IntelliJ EditorConfig additions don't allow configuring different indentation for different constructs. By the way, the Google Java Code style and `{{google-java-format}}` also use a uniform continuation indent for all constructs, except that the Google style uses 2 and 4 instead of 4 and 8. I think it's better to have some uniform style that is easy to apply automatically by everyone and EditorConfig is the best I've found so far if we don't want to go further and use a tool like `{{google-java-format}}`. (See also <https://lists.apache.org/thread.html/rfb079ec4cfd35bcb93df9c2163aaa121e392282f0f3d9710c8ade811%40%3Cdev.flink.apache.org%3E>)
4. I agree that we should have a rule which is easy to apply automatically. However, I think we need to keep EditorConfig to be aligned with the rules in Flink code-style guide[1]. Otherwise it may still be valid that one overrides the code style settings to be aligned with the code-style guide. So maybe we should have an agreement in the ML about the new continuation indentation to apply? [1] <https://flink.apache.org/contributing/code-style-and-quality-formatting.html>
5. I'm afraid the ML discussion has fizzled out, as these discussions often do. I tried to just make some progress with a tool that can apply a style automatically. What EditorConfig does is the closest we can get without using Spotless/google-java-format. By adding it to the code base I'm trying to force the issue but I think not many people even use EditorConfig.
6. I've just stumbled across this issue. Suddenly IntelliJ was inserting everywhere double indentations, when it was not supposed to be doing. It was super annoying for me to modify any pre-existing code with single indentations (It took me quite a bit of time to figure what has happened and found this as the root cause). Before I figured out and disabled `{{.editorconfig}}` changes in the IntelliJ, I've spent a lot of time manually reverting all of the changes, and still [~AHeise] found some more unintentional changes... <https://github.com/apache/flink/pull/13741/commits/1e26fb076f97cb27397a1d76d26aff11790e438f#r511927651>  
[https://github.com/apache/flink/pull/13741#discussion\\_r511925956](https://github.com/apache/flink/pull/13741#discussion_r511925956) [~aljoscha] after reading the mailing list discussion about using this "ratchet" thing, I didn't find anywhere where we agreed upon changing the continuation indent from 4 to 8. It looks like it's invalid with our coding style, could we maybe revert it? If we want to change our coding style, and automate the formatting, eh fine +0.1 from my side (I don't like braking the `{{git blame}}`/IntelliJ's annotations). But I agree with [~chesnay]. -1 for any gradual change. And -1 for a change like this, that's only affecting the exact particular lines of code that I touched, even unknowingly, via copy pasting or refactoring/changing signatures etc. I fear this would create a lot of noise in the upcoming PRs for very long time.
7. I've had similar issues with line wrapping, and by now disabled the editor config support :/
8. Those are more arguments for having an opinionated, enforced coding style. You shouldn't have to worry about coding style/formatting at all in a PR if you have a good tool for that. Unfortunately, the discussion about that seems to have died. Without that, I'll keep using the EditorConfig setup because it's the closest thing I can get to a good formatter with configuration in a text file that I can version with code. Plus it pleases our Checkstyle and I won't worry about a style guide that requires humans to check/apply it by hand.
9. But: 1. there was no discussion about changing the the code style, (this is what you are effectively doing), just that you will prepare `{{.editorconfig}}` compliant with the checkstyle 2. there was -1 from [~chesnay] for a gradual change I honestly think this will do more harm than good, blowing up sizes of the PRs in any foreseeable future.
10. There was this discussion where I presented the EditorConfig file that I wanted to add: <https://lists.apache.org/thread.html/r481bb622410718cd454d251b194c7c1ad358e5d861fdacc9a4be3b5b%40%3Cdev.flink.apache.org%3E>. People either didn't care or were +1 in that thread.
11. I've reverted the contested change. As we discussed offline, let's aim to change the codestyle and reformat the codebase after the 1.12 feature freeze. master e2dd0b6a89 release-1.11 c685368b20 release-1.10 bbeecc3e20

