This document explains the changes to formatter preferences in the patch submitted for eclipse 
bug 303519.

**Newly Added Preferences:**

New alignment preferences are added to customize alignment for different annotations.
1. `FORMATTER_ALIGNMENT_FOR_ANNOTATIONS_ON_PACKAGE`
2. `FORMATTER_ALIGNMENT_FOR_ANNOTATIONS_ON_TYPE`
3. `FORMATTER_ALIGNMENT_FOR_ANNOTATIONS_ON_FIELD`
4. `FORMATTER_ALIGNMENT_FOR_ANNOTATIONS_ON_METHOD`
5. `FORMATTER_ALIGNMENT_FOR_ANNOTATIONS_ON_PARAMETER`
6. `FORMATTER_ALIGNMENT_FOR_ANNOTATIONS_ON_LOCAL_VARIABLE`

Following preferences are added for the alignment of the types which was not present earlier.
7. `FORMATTER_ALIGNMENT_FOR_GENERIC_TYPE_ARGUMENTS`
8. `FORMATTER_ALIGNMENT_FOR_CASCADING_METHOD_INVOCATION_WITH_ARGUMENTS`
9. `FORMATTER_ALIGNMENT_FOR_LOCAL_VARIABLE_DECLARATION`
10. `FORMATTER_ALIGNMENT_FOR_FIELD_DECLARATION`
11. `FORMATTER_ALIGNMENT_FOR_FOR_STATEMENT`
12. `FORMATTER_ALIGNMENT_FOR_NEW_ANONYMOUS_CLASS`

Some other preferences
13. `FORMATTER_INSERT_NEW_LINE_AFTER_ANNOTATION_HAVING_ARGUMENTS`
14. `FORMATTER_INSERT_SPACE_AFTER_COMMENT_PREFIX`
15. `FORMATTER_FORCE_IF_ELSE_STATEMENT_BRACE`

New wrapping preferences.
16. `FORMATTER_WRAP_NON_SIMPLE_PACKAGE_ANNOTATION`
17. `FORMATTER_WRAP_NON_SIMPLE_TYPE_ANNOTATION`
18. `FORMATTER_WRAP_NON_SIMPLE_MEMBER_ANNOTATION`
19. `FORMATTER_WRAP_NON_SIMPLE_PARAMETER_ANNOTATION`
20. `FORMATTER_WRAP_NON_SIMPLE_LOCAL_VARIABLE_ANNOTATION`
21. `FORMATTER_WRAP_COMMENT_INLINE_TAGS`
22. `FORMATTER_WRAP_PREFER_TWO_FRAGMENTS`

Preferences for sorting the members of the alignment.
23. `FORMATTER_SORT_PACKAGE_ANNOTATIONS`
24. `FORMATTER_SORT_TYPE_ANNOTATIONS`
25. `FORMATTER_SORT_MEMBER_ANNOTATIONS`
26. `FORMATTER_SORT_PARAMETER_ANNOTATIONS`
27. `FORMATTER_SORT_LOCAL_VARIABLE_ANNOTATIONS`

Preference to have new line after `<p>` tag in javadoc
28. `FORMATTER_COMMENT_NEW_LINE_AT_START_OF_HTML_PARAGRAPH`

Please refer file `DefaultCodeFormatterConstants.java` for these preferences.

**Fragment count dependent preferences:**

In order to retain both forward and backwards compatibility all alignment preferences will have
a corresponding new preferences with **suffix 'count_dependent'** appended to the preference name.

The new format `FragmentDependentAlignment` for these preferences is tuple of three integers. It allows custom alignment based on the number of fragments. e.g. Consider the preference `alignment_for_parameters_in_method_declaration` which decides alignment preference for parameters in method declaration. Lets say the value of this pref is 16|4|48 which is basically COMPACT_SPLIT| 4 | ONE_PER_LINE_SPLIT.
So if the number of parameters in method declaration, they would be aligned in compact way else they would be split into one per line.

### Deprecated Preferences:
The following preferences are deprecated as the desired effect can be achieved using the corresponding alignment preference.

1. `FORMATTER_INSERT_NEW_LINE_AFTER_ANNOTATION_ON_FIELD`
2. `FORMATTER_INSERT_NEW_LINE_AFTER_ANNOTATION_ON_METHOD`
3. `FORMATTER_INSERT_NEW_LINE_AFTER_ANNOTATION_ON_PACKAGE`
4. `FORMATTER_INSERT_NEW_LINE_AFTER_ANNOTATION_ON_TYPE`
5. `FORMATTER_INSERT_NEW_LINE_AFTER_ANNOTATION_ON_PARAMETER`
6. `FORMATTER_INSERT_NEW_LINE_AFTER_ANNOTATION_ON_LOCAL_VARIABLE`
7. `FORMATTER_INSERT_NEW_LINE_BEFORE_CLOSING_BRACE_IN_ARRAY_INITIALIZER`