Code Smells

Uma imagem com céu noturno

Descrição gerada automaticamente

Done by

Paula Lopes 58655

**Long Parameter List**

**Code snippet:** writeMetadatatoPdfByFileNames(BibDatabaseContext databaseContext, BibDatabase dataBase, Vector<String> fileNames, Charset encoding, FilePreferences filePreferences, XmpPdfExporter xmpPdfExporter, EmbeddedBibFilePdfExporter embeddedBibFilePdfExporter, boolean writeXMP, boolean embeddBibfile){…}



**Location:** src/main/java/org/jabref/cli/ArgumentProcessor.java

**Explanation:** A long parameter list code smell indicates that a new structute should be created or the method is doing stuff that it shouldnt (creating other code smells like Inappropriate Intimacy). As seen in the code snippet, the method ‘writeMetadatatoPdfByFileNames’ has so many arguments the screenshot is barely readable xD.

**Refactoring proposal:** A refactoring proposal for this code smell would be to limit it to have a maximum of 4-7 parameters. Let’s say we created two classes one that stored all the information about the dataBase and it’s context (BibData class) and a second class that stores the files preferences as well as it’s pdf exporter types (File class). This way we could access the information of each argumente inside it’s class. For exemple, BibData class should have a method that returns the databaseContext and another one that returns the dataBase.

writeMetadatatoPdfByFileNames(BibData data, Vector<String> fileNames, Charset encoding, File f, boolean writeXMP, boolean embeddBibfile){…}

**Switch Statement**

**Code snippet:**

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

**Location:** src/main/java/org/jabref/gui/externalfiletype/ExternalFileTypes.java

**Explanation:** Switch/ifs statements should be simple and have a low complexity, otherwise the code can be unreadable and difficult to mantain. Therefore, the cognitive complexity of any method should not be high.

**Refactoring proposal:** A refactoring proposal for this code smell would be to create a search method to remove an external file based on the entry type given as a parameter.

**Uma imagem com texto

Descrição gerada automaticamente**

**Uma imagem com texto

Descrição gerada automaticamente**

Otherwise, without this method, this two sections of the code could also be considered as duplicate code.

**Empty Methods**

**Code snippet:**

Uma imagem com texto

Descrição gerada automaticamente

**Location:** src/main/java/org/jabref/gui/commonfxcontrols/FieldFormatterCleanupsPanelViewModel.java

**Explanation:** Methods should never be empty. However, there are a few cases where a method might not have a body. For exemple: It is not yet or never will be supported or the method is a intentionally blank override ( probably because of another code smell called failed request ). Exceptions for this code smell are empty methods in abstract classes.

**Refactoring proposal:** A refactoring proposal for this code smell is as simples as adding a nested comment explaining why the method needs to be empty or throwing and exception.

Public FieldFormatterCleanupsPanelViewModel() {

throw new UnsupportedOperationException();

}