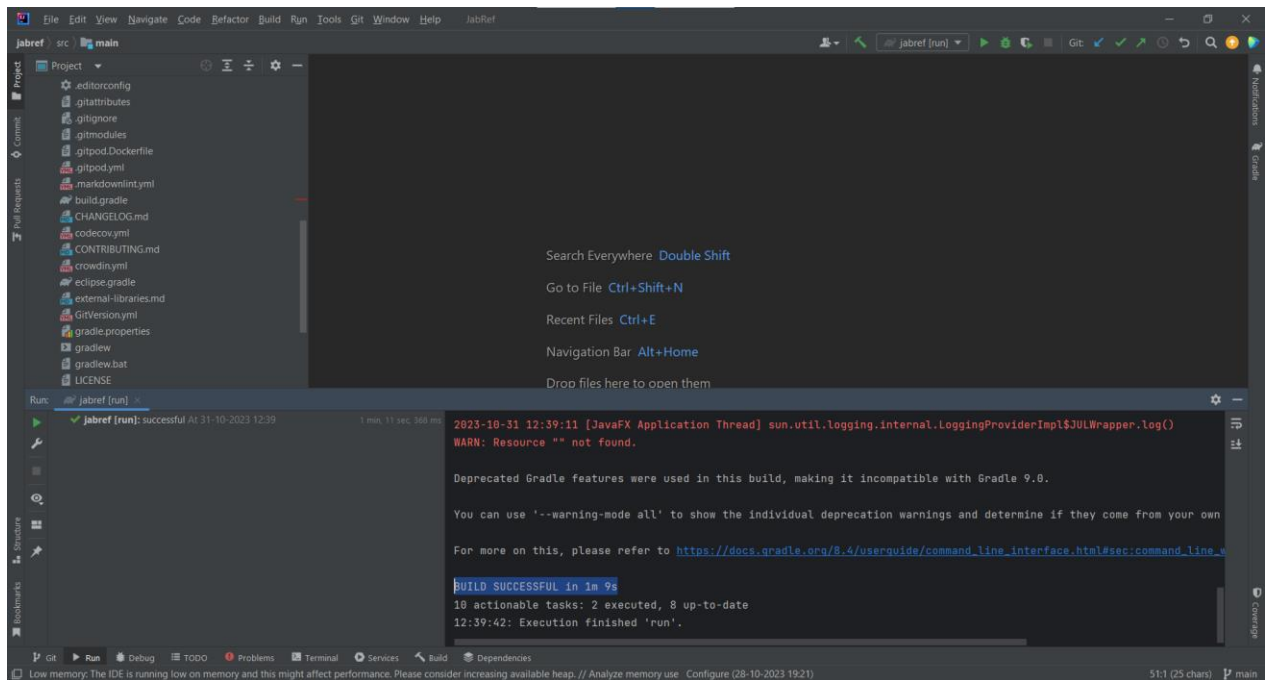
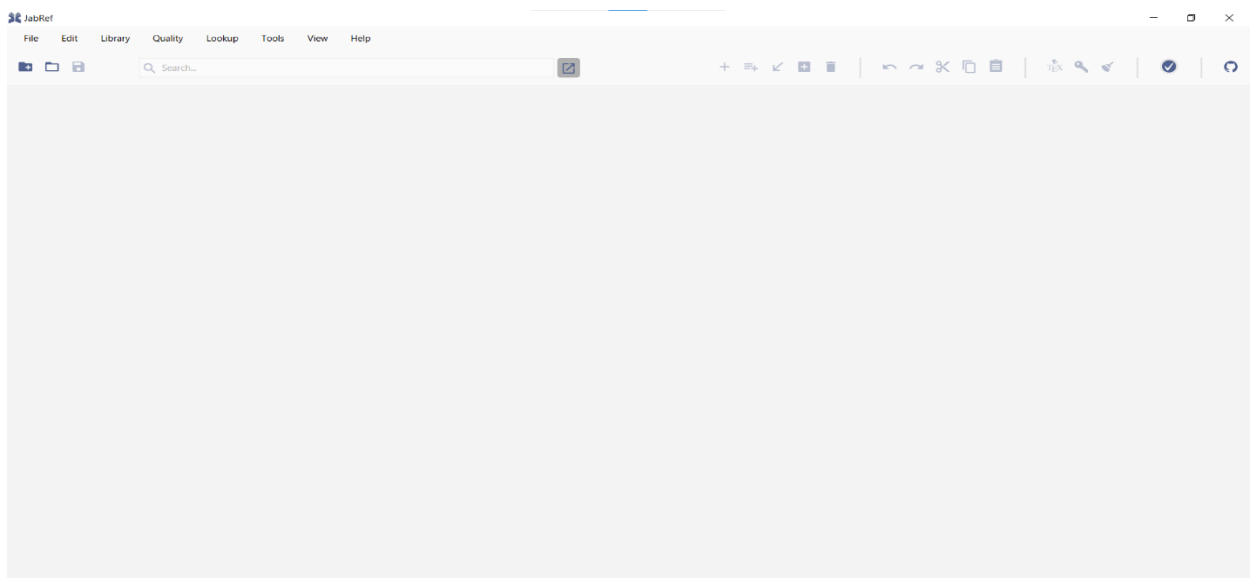


CSC 6110 : Software Engineering
Assignment 2
Fall 2023
Student name: Kanchan Chopde
Based on: Software Build and Testing

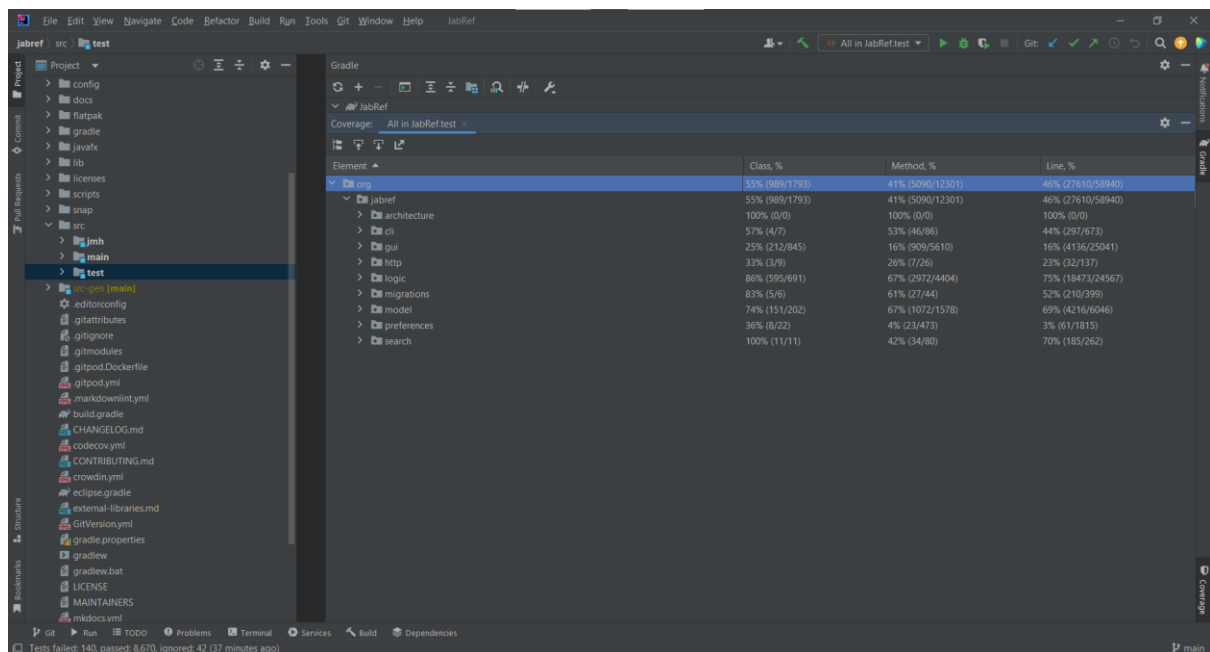
- Successful Build



- Jabref UI after Run:



Generated Coverage report:



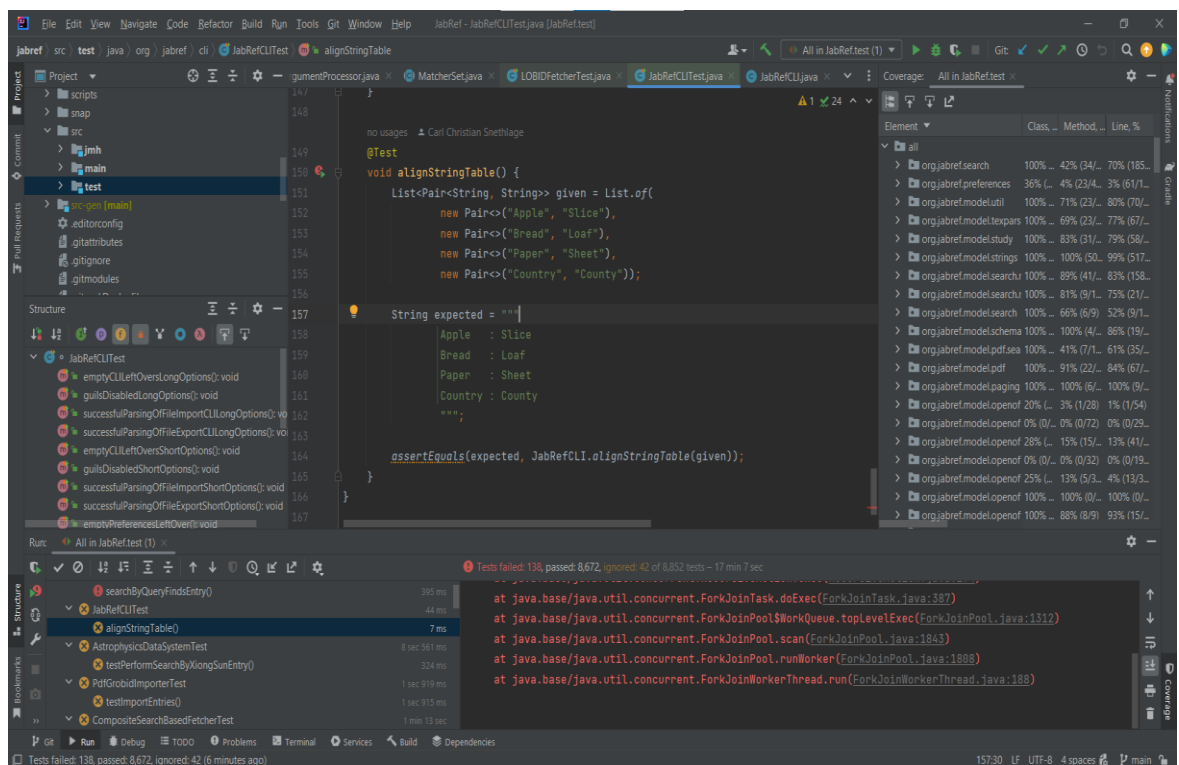
- Explore the filed test cases and determine the common reasons for the failures of any 10 failed cases. What could be done to fix these testcase failures?

For the test case failures below are the listed common reasons and probable fixes:

1) alignStringTable() test case has failed.

Reason: In expected result there is added line separator(space) causing issue.

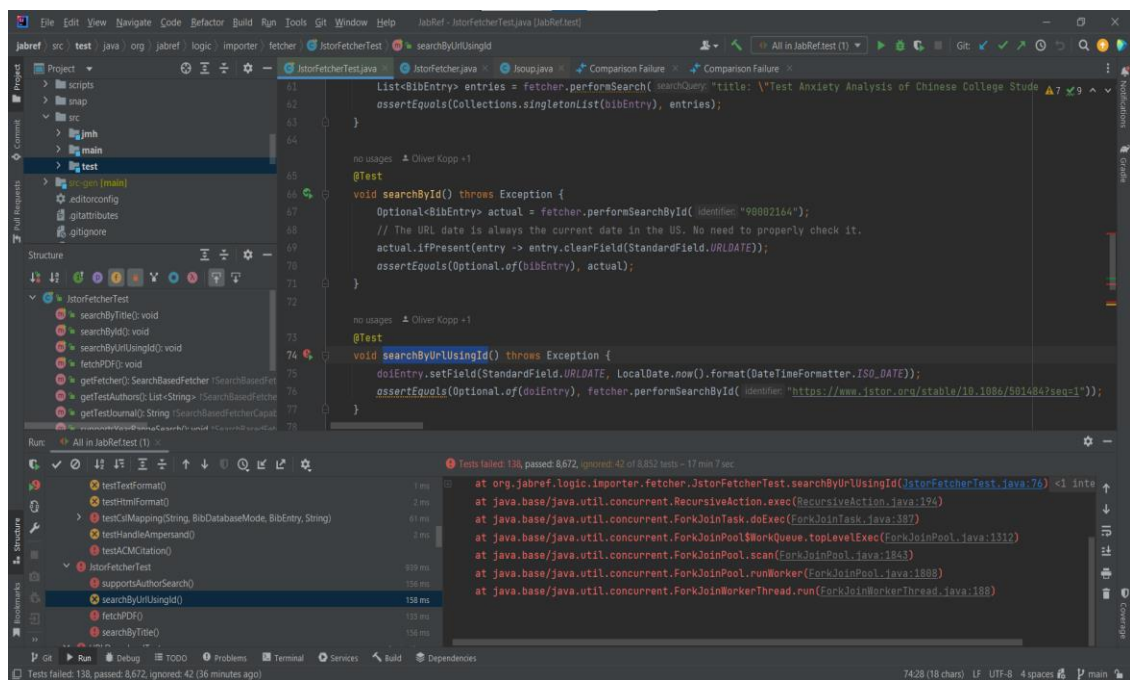
Fix: remove the space added.



2) searchByUrlUsingId()

Reason: expected doi entry is not matching with actual doi entry which is getting fetched by `performSearchbyId()` method. It is returning empty.

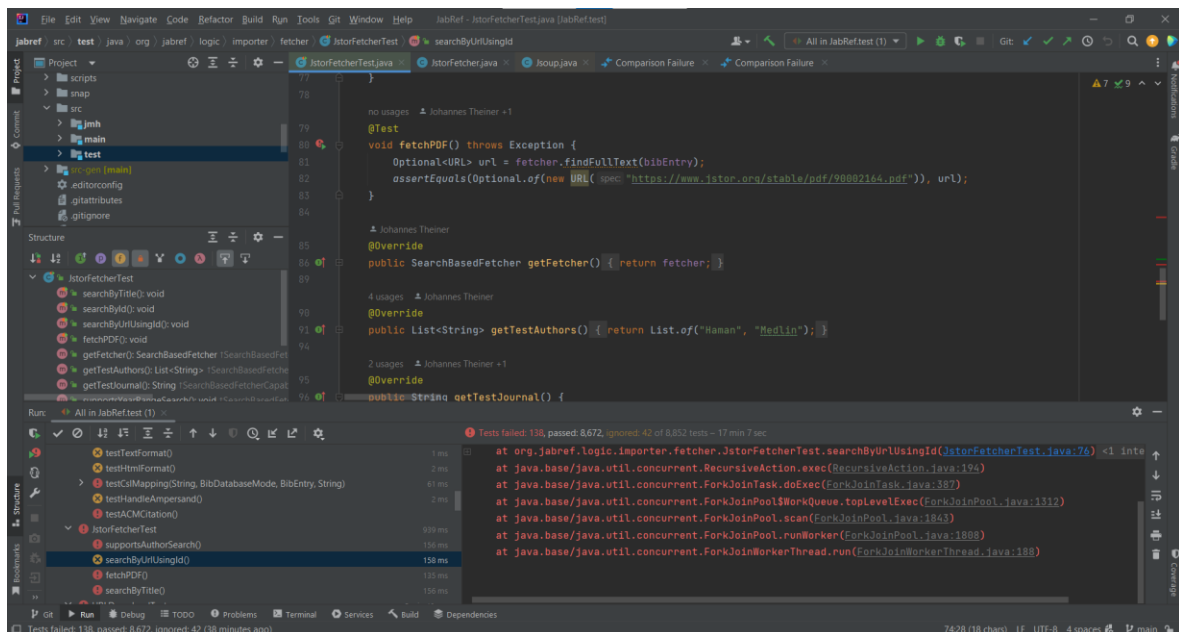
Fix: check if `fetcher.performSearchbyId()` gets correct entry.



3) fetchPDF()

Reason : This method calls findFullText(bibEntry), in this method valid URL is successfully extracted from the HTML element. If not it throws error.

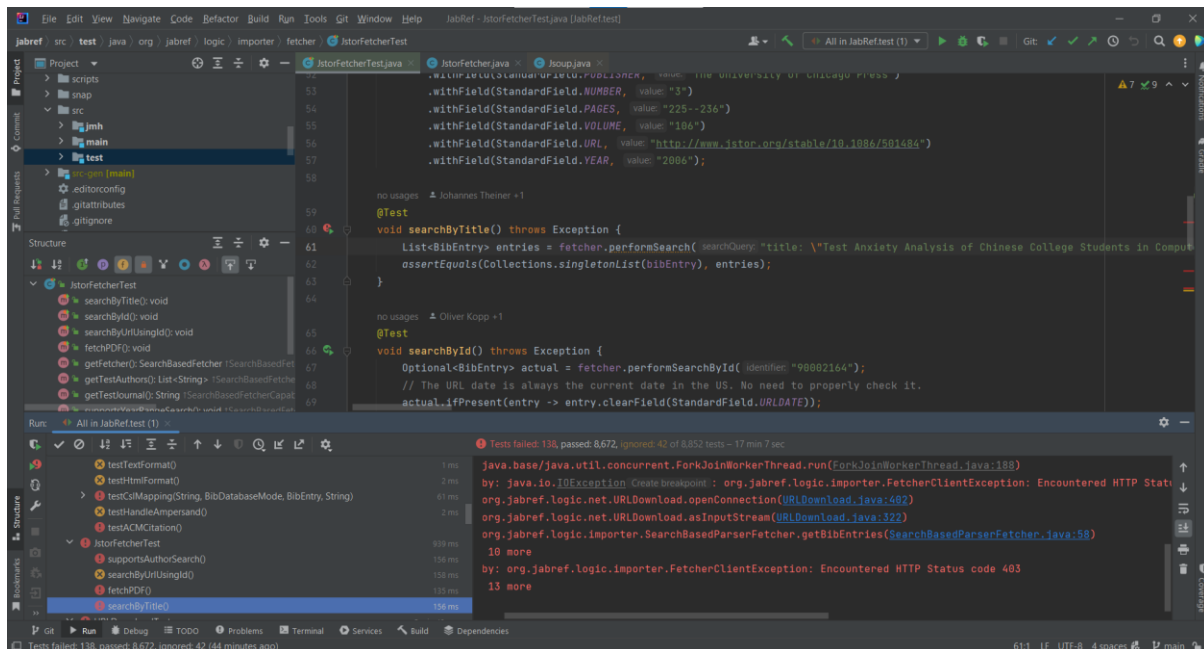
Fix: check if fetcher.findFullText(bibEntry) gets correct entry.



4) SearchByTitle()

Reason : An HTTP 403 response code means that a client is forbidden from accessing a valid URL.

Fix: need to check if `fetcher.performSearch(url)` is able to get value.

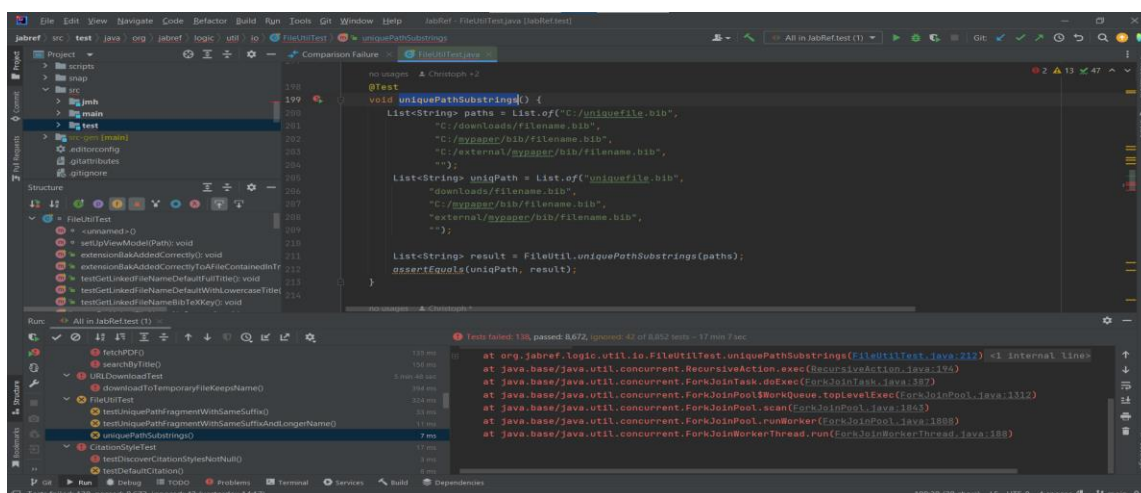


The screenshot shows an IDE with a Java project. The main editor displays the `JstorFetcherTest.java` file. The `searchByTitle()` test method is highlighted, showing a call to `fetcher.performSearch()` with a specific title. The test fails with an `IOException` due to an `Encountered HTTP Status code 403`. The error message is: `java.io.IOException: Encountered HTTP Status code 403`. The stack trace shows the error originates from `org.jabref.logic.importer.FetcherClientException` and `org.jabref.logic.net.URLDownload`. The test results pane at the bottom shows 136 tests failed, 8,672 passed, and 42 ignored.

5) UniquePathsSubstring()

Reason: The actual value here which is coming from “paths” variable does not match with `uniqPath` as this value has fields where “C:/ ” is omitted for few entries.

Possible Fix: The function `uniquepathSubtring` is returning the minimal unique path for substring but it is not trimming `C:\` and returning same substring which is passed.

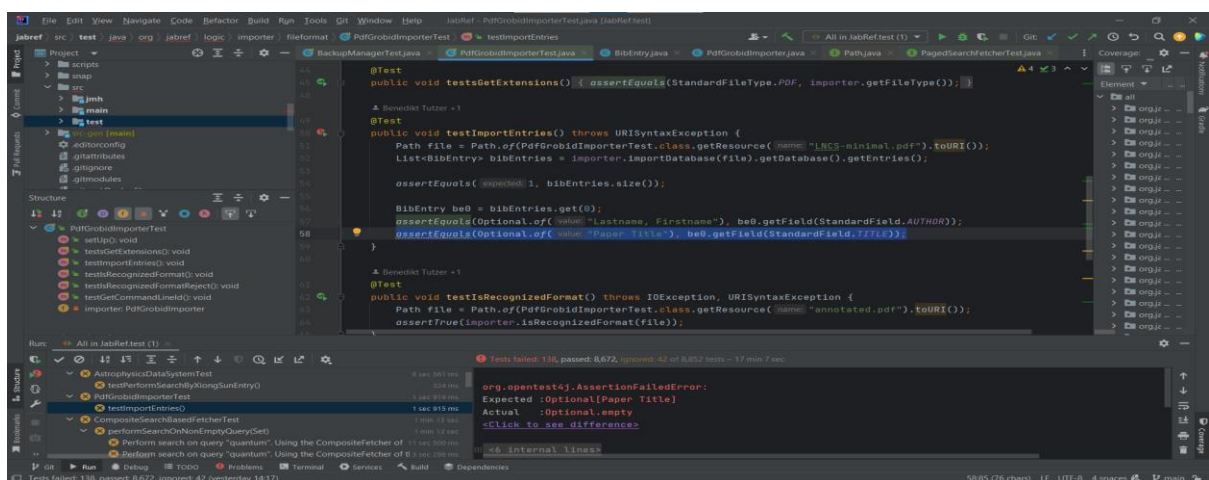


The screenshot shows an IDE with a Java project. The main editor displays the `FileUtilTest.java` file. The `uniquePathSubstrings()` test method is highlighted, showing a call to `FileUtil.uniquePathSubstrings()` with a list of paths. The test fails with an `IOException` due to an `Encountered HTTP Status code 403`. The error message is: `java.io.IOException: Encountered HTTP Status code 403`. The stack trace shows the error originates from `org.jabref.logic.util.io.FileUtilTest` and `org.jabref.logic.net.URLDownload`. The test results pane at the bottom shows 136 tests failed, 8,672 passed, and 42 ignored.

6) testImportEntries()

Reason: assertion fails for Title field as it does not matches the expected title i.e. "Paper Title".

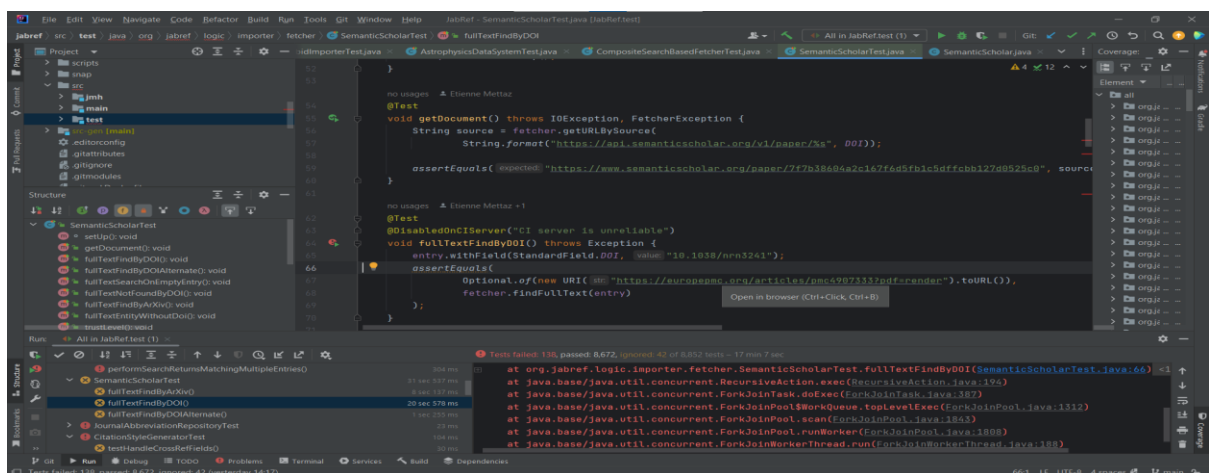
Probable Fix: Need to check for passed URI entry i.e. LNCS-minimal.pdf what is the value for Title field and modify accordingly.



7) fullTextFindByDOI()

Reason: There is error for pdf lookup with DOI. It is returning status code 403. Hence assertion is failing here.

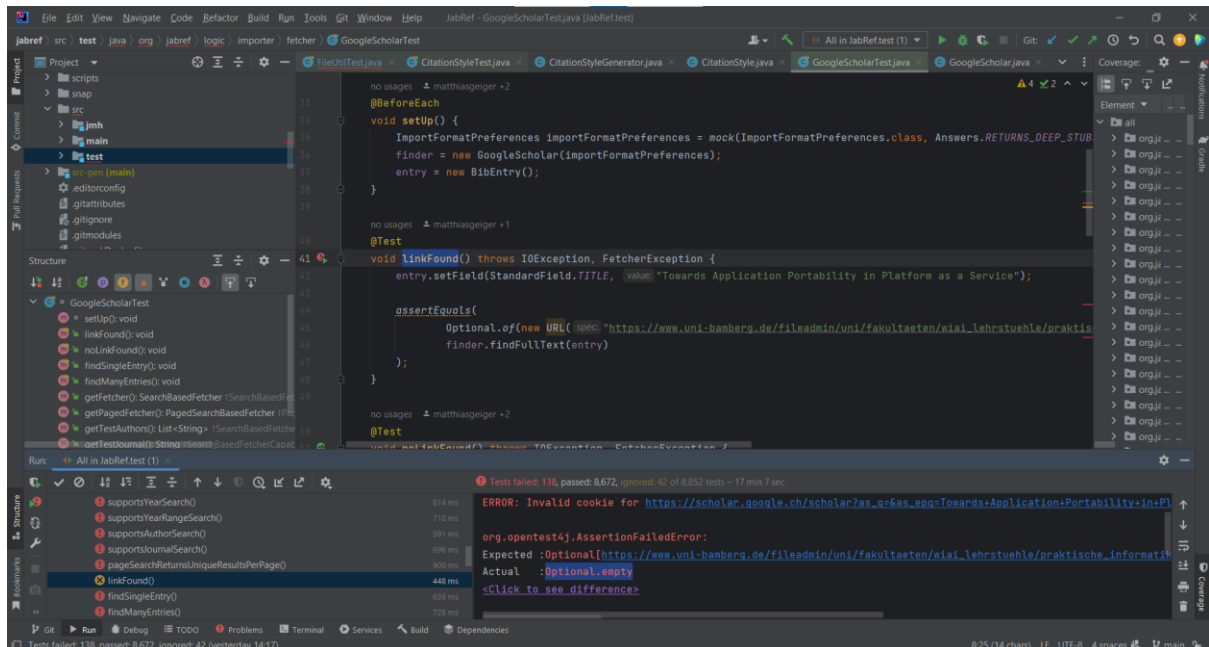
Probable Fix: need to check why is it returning 403 forbidden access.



8) linkfound()

Reason: Actual result is returning empty link and hence AssetEquals fails.

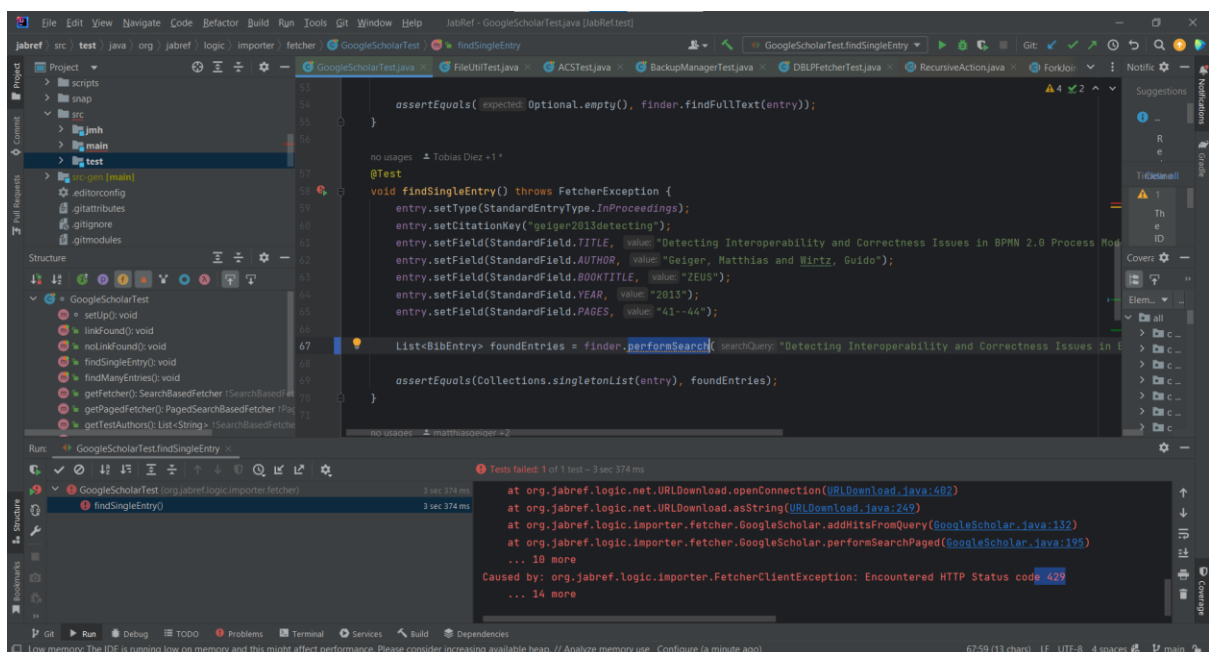
Probable Fix: Check entry title if properly provided.



9) findSingleEntry()

Reason : return status 429 code.

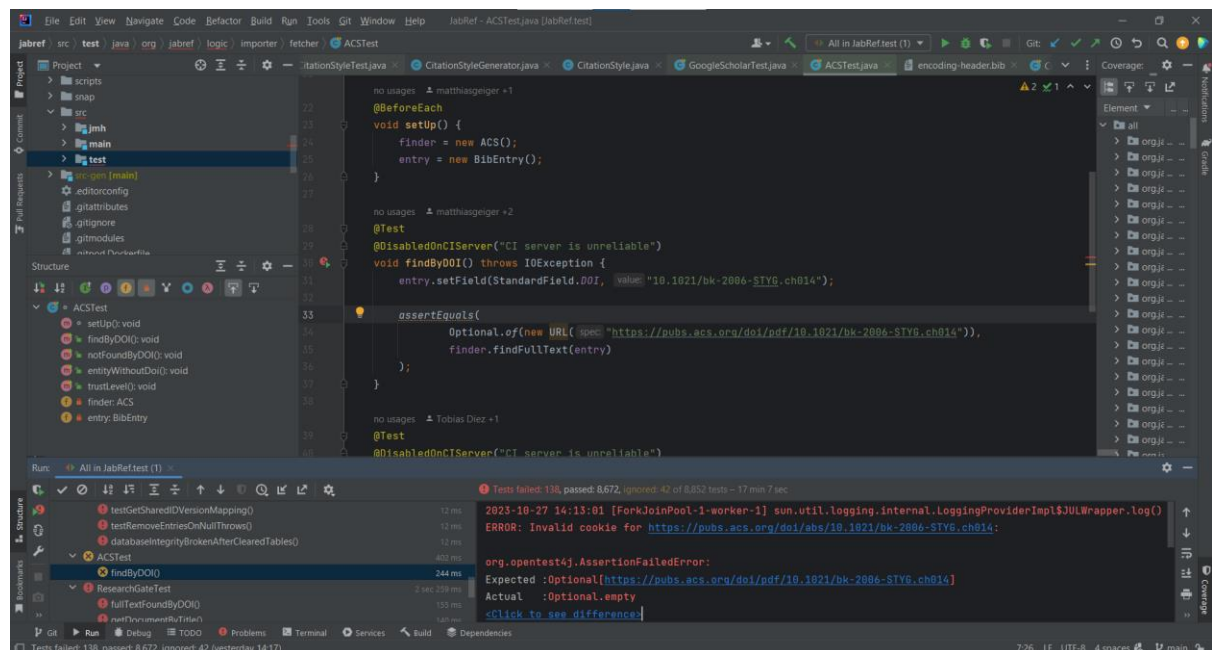
Probable Fix: 429 is response from server. May be it has no adequate resources to make request.



10)findbyDOI()

Reason :DOI value does not match the expected value. Actual value obtain is empty. Hence AssertEqual is failing.

Probable Fix: Need to check if DOI is set properly to DOI field and entry passed to findFullText is getting expected results.



- Explore the coverage report. Which modules are well covered by automated tests? Which modules have least coverage. Assume, all the folders in org/jabref are modules. Do you see any pattern among the well-covered and least covered areas?

Modules well covered by automated test:

100% by class %

- Search
- Gui.Actions
- Gui.ExternalFileType
- Gui.Prefrences.keybindings.presets
- Gui.Search.Rules.Describer
- Gui.util.comparator

• Logic.auxparser
• Logic.Bibtext
• Logic.bst
• Logic.bst.util
• Logic.bst.citationkeypattern
• logic.cleanup
• logic.crawler
• logic.database
• logic.externalfiles
• logic.formatter
• logic.formatter.bibtextfields
• logic.formatter.casechanger
• logic.formatter.minifier
• logic.git
• logic.groups
• logic.help
• importer.fileformat.medline
• importer.fileformat.mods
• logic.integrity
• logic.journal
• logic.msbib
• logic.pdf.search.indexing
• logic.preview
• logic.protectedterms
• logic.remote
• logic.remote.client
• logic.remote.server
• logic.search
• logic.shared.securty
• logic.texparser
• logic.util.io
• logic.util.strings
• model
• model.database
• model.entry

• model.entry.event
• model.entry.field
• model.groups.event
• model.metadata
• model.metadata.event
• model.openoffice
• model.paging
• model.pdf
• model.pdf.search
• model.schema
• model.search
• model.search.matchers
• model.search.rules
• model.strings
• model.study
• model.texparser
• model.util
<p>Modules covered between (97.5%-6.2%) Class %</p> <p>org.jabref.logic.importer.fileformat.citavi</p> <p>org.jabref.logic.importer.fileformat</p> <p>org.jabref.logic.importer.fetcher</p> <p>org.jabref.logic.importer.fetcher.transformers</p> <p>org.jabref.model.groups</p> <p>org.jabref.model.entry.types</p> <p>org.jabref.logic.exporter</p> <p>org.jabref.logic.layout.format</p> <p>org.jabref.logic.importer</p> <p>org.jabref.logic.citationstyle</p> <p>org.jabref.model.entry.identifier</p> <p>org.jabref.logic.xmp</p> <p>org.jabref.logic.bibtex.comparator</p> <p>org.jabref.logic.l10n</p> <p>org.jabref.gui.icon</p> <p>org.jabref.migrations</p> <p>org.jabref.gui.autocompleter</p>

org.jabref.model.database.event
org.jabref.gui.mergeentries.newmergedialog.fieldsmerger
org.jabref.logic.importer.util
org.jabref.logic.layout
org.jabref.gui.preferences.journals
org.jabref.gui.importer.actions
org.jabref.logic.util
org.jabref.logic.undo
org.jabref.logic.pdf.search.indexing
org.jabref.logic
org.jabref.gui.util.component
org.jabref.gui.preferences.keybindings
org.jabref.gui.libraryproperties.constants
org.jabref.logic.openoffice.style
org.jabref.gui.theme
org.jabref.logic.pdf
org.jabref.cli
org.jabref.gui.externalfiles
org.jabref.logic.preferences
org.jabref.logic.net
org.jabref.logic.importer.fetcher.isbntobibtex
org.jabref.http.dto
org.jabref.gui.libraryproperties.contentsselectors
org.jabref.gui.journals
org.jabref.gui.edit.automaticfiededitor.renamefield
org.jabref.gui.edit.automaticfiededitor.editfieldcontent
org.jabref.gui.edit.automaticfiededitor.copyormovecontent
org.jabref.gui.commonfxcontrols
org.jabref.gui.autosaveandbackup
org.jabref.gui.edit
org.jabref.logic.shared
org.jabref.gui.keyboard
org.jabref.gui.mergeentries.newmergedialog
org.jabref.gui.desktop.os
org.jabref.gui.util
org.jabref.gui.search
org.jabref.gui.edit.automaticfiededitor
org.jabref.preferences

org.jabref.http.server
org.jabref.gui.menus
org.jabref.gui.importer.fetcher
org.jabref.gui.documentviewer
org.jabref.gui.undo
org.jabref.model.openoffice.style
org.jabref.gui.slr
org.jabref.gui.sidepane
org.jabref.gui.exporter
org.jabref.gui.maintable
org.jabref.gui.groups
org.jabref.model.openoffice.ootext
org.jabref.gui.auximport
org.jabref.gui
org.jabref.gui.entryeditor
org.jabref.model.openoffice.util
org.jabref.gui.preview
org.jabref.gui.entryeditor.fileannotationtab
org.jabref.gui.duplicationFinder
org.jabref.gui.util.uithreadaware
org.jabref.gui.texparser
org.jabref.gui.specialfields
org.jabref.gui.mergeentries
org.jabref.gui.mergeentries.newmergedialog.cell
org.jabref.gui.fieldeditors
org.jabref.gui.importer
org.jabref.model.openoffice.uno

Modules with least coverage:

0% by class %

- org.jabref.model.openoffice.uno

- org.jabref.model.openoffice.rangesort
- org.jabref.logic.shared.prefs
- org.jabref.logic.shared.listener
- org.jabref.logic.shared.exception
- org.jabref.logic.shared.event
- org.jabref.logic.preview
- org.jabref.logic.openoffice.frontend
- org.jabref.logic.openoffice.backend
- org.jabref.logic.openoffice.action
- org.jabref.logic.openoffice
- org.jabref.logic.net.ssl
- org.jabref.logic.logging
- org.jabref.http
- org.jabref.gui.shared
- org.jabref.gui.remote
- org.jabref.gui.push
- org.jabref.gui.preferences.xmp
- org.jabref.gui.preferences.websearch
- org.jabref.gui.preferences.table
- org.jabref.gui.preferences.protectedterms
- org.jabref.gui.preferences.preview
- org.jabref.gui.preferences.network
- org.jabref.gui.preferences.nameformatter
- org.jabref.gui.preferences.linkedfiles
- org.jabref.gui.preferences.groups
- org.jabref.gui.preferences.general
- org.jabref.gui.preferences.externalfiletypes

- org.jabref.gui.preferences.external
- org.jabref.gui.preferences.export
- org.jabref.gui.preferences.entryeditor
- org.jabref.gui.preferences.entry
- org.jabref.gui.preferences.customimporter
- org.jabref.gui.preferences.customexporter
- org.jabref.gui.preferences.customentrytypes
- org.jabref.gui.preferences.citationkeypattern
- org.jabref.gui.preferences.autocompletion
- org.jabref.gui.preferences
- org.jabref.gui.openoffice
- org.jabref.gui.mergeentries.newmergedialog.toolbar
- org.jabref.gui.mergeentries.newmergedialog.diffhighlighter
- org.jabref.gui.mergeentries.newmergedialog.cell.sidebuttons
- org.jabref.gui.maintable.columns
- org.jabref.gui.logging
- org.jabref.gui.linkedfile
- org.jabref.gui.libraryproperties.saving
- org.jabref.gui.libraryproperties.preamble
- org.jabref.gui.libraryproperties.keypattern
- org.jabref.gui.libraryproperties.general
- org.jabref.gui.libraryproperties
- org.jabref.gui.integrity
- org.jabref.gui.help
- org.jabref.gui.fieldeditors.journalinfo
- org.jabref.gui.fieldeditors.identifier
- org.jabref.gui.fieldeditors.contextmenu

- org.jabref.gui.errorconsole
- org.jabref.gui.entryeditor.citationrelationtab.semanticscholar
- org.jabref.gui.entryeditor.citationrelationtab
- org.jabref.gui.dialogs
- org.jabref.gui.desktop
- org.jabref.gui.copyfiles
- org.jabref.gui.collab.stringrename
- org.jabref.gui.collab.stringdelete
- org.jabref.gui.collab.stringchange
- org.jabref.gui.collab.stringadd
- org.jabref.gui.collab.preamblechange
- org.jabref.gui.collab.metedatachange
- org.jabref.gui.collab.groupchange
- org.jabref.gui.collab.entrydelete
- org.jabref.gui.collab.entrychange
- org.jabref.gui.collab.entryadd
- org.jabref.gui.collab
- org.jabref.gui.cleanup
- org.jabref.gui.citationkeypattern
- org.jabref.gui.bibtexextractor
- org.jabref.gui.backup
- org.jabref.build.xjc

Pattern between them:

- Both well covered and least covered test cases includes base modules such as Gui, Logic, Model, Http.

- In well covered modules, most of the Logic modules and Model modules are covered. Seems like all essential functionalities are covered.
- Where as , most of the Gui modules are least covered.