

# Error Handling in JabRef

## Throwing and Catching Exceptions

Principles:

- All exceptions we throw should be or extend `JabRefException`; This is especially important if the message stored in the Exception should be shown to the user. `JabRefException` has already implemented the `getLocalizedMessage()` method which should be used for such cases (see details below!).
- Catch and wrap all API exceptions (such as `IOExceptions`) and rethrow them
  - Example:

```
try {  
    // ...  
} catch (IOException ioe) {  
    throw new JabRefException("Something went wrong...",  
        Localization.lang("Something went wrong...", ioe);  
}
```

- Never, ever throw and catch `Exception` or `Throwable`
- Errors should only be logged when they are finally caught (i.e., logged only once). See **Logging** for details.
- If the Exception message is intended to be shown to the User in the UI (see below) provide also a localizedMessage (see `JabRefException`).

(Rationale and further reading: <https://www.baeldung.com/java-exceptions>)

## Outputting Errors in the UI

Principle: Error messages shown to the User should not contain technical details (e.g., underlying exceptions, or even stack traces). Instead, the message should be concise, understandable for non-programmers and localized. The technical reasons (and stack traces) for a failure should only be logged.

To show error message two different ways are usually used in JabRef:

- showing an error dialog
- updating the status bar at the bottom of the main window

