



Woraus zusammensetzen? Zum Beispiel:

1. Grundlage ist ein *Window* mit Client-side-Fensterdekoration.
2. Darin sowohl eine *HeaderBar* als auch ein *ScrolledWindow*.
3. Die *HeaderBar* enthält einen *Button* zum Öffnen und einen *Button* zum Speichern.
4. Das *ScrolledWindow* enthält eine *TextView*.

*Hinweis:*

<https://pelzflorian.de/gui-prog-gtk-2017/text-view-utils.tar.gz>

Dieser Code liest eine Datei an einem Stück ein. Solange die Datei geladen wird, wird das Programm blockiert. Für große Dateien wäre das also nicht akzeptabel. Echte Programme zur Textverarbeitung würden Bibliotheken wie „Tepl“ benutzen.

Statt eines einfachen `gtk_main()` kann man zur besseren Integration in das Betriebssystem bzw. die Benutzeroberfläche eine *Application* benutzen. Für weitere Features – wie die Abfrage, ob Menüleisten im Fenster (wie bei Windows) gezeigt werden sollen oder nicht (bei Ubuntu's Unity, Mac OS X, GNOME) – sollte dann auch ein *ApplicationWindow* statt eines *Window* verwendet werden.

- Eine Anwendung läuft nur einmal gleichzeitig.
- Wiederholtes Öffnen aktiviert nur die bereits bestehende Anwendung, welche dann z.B. ein neues Fenster öffnet.
- D.h. die `main()`-Methode erzeugt nur noch eine *Application*, bindet ein `activate`-Signal und startet die Anwendung per `g_application_run()`-Aufruf. `gtk_main()` ist Teil der *Application* und wird nicht separat aufgerufen.
- Zudem sollten Funktionen des Anwendungsmenüs, der Menüleiste und Tastenkürzel über die *ActionMap*-Eigenschaften der *Application* bereitgestellt werden.

Beispiele für den Einbau in den Editor vorher:

- <https://pelzflorian.de/git/gui-prog-gtk-2016/tree/bin/text2/text.c>
- <https://developer.gnome.org/gtk3/stable/ch01s04.html>
- Dokumentation für *GtkApplication*, *GtkApplicationWindow*

Weitere Ideen:

- An Stelle des *Buttons* zum Öffnen kann auch wie bei *gedit* ein *Menu-Button* mit *Popover* benutzt werden. Darin kann dann z.B. ein *Recent-Chooser* und der normale *Öffnen-Button* platziert werden.
- Die Bibliothek „*GtkSourceView*“ bietet ein spezialisiertes *TextView*-Widget für Texteditoren.
- Analog kann auch ein *Bildbetrachter*, *PDF-Betrachter*, *Web-Browser* u.s.w. geschrieben werden. Neben der Entwicklung des Endnutzerprogramms liegt die Hauptarbeit bei solchen Programmen beim Bereitstellen eines passenden Widgets.