

L05 Documentation

Markus Raab

Institute of Information Systems Engineering, TU Wien

This work is licensed under a Creative Commons
“Attribution-ShareAlike 4.0 International” license.



Introduction

- 1 Introduction
- 2 Generate
- 3 Verify
- 4 Meeting
 - Recapitulation
 - Assignments
 - Preview

Learning Outcomes

After successful completion of L05
students will be able to use techniques to

- remember basics
- generate documentation
- verify documentation

Correctness

Documentation tends to be:

- outdated
- incorrect
- not helpful

Make Sure That

- documentation gets reviewed
- documentation is in source code management
- small distance to code
- avoid redundant information

Different Types of Documentation

tutorials for learning

how-to solving a problem

reference searching for details
e.g. man pages, API docu

explanations in doc/dev “How?”

decisions for background information “Why?”

examples for copy&paste

Different People

beginners never forget everybody starts as beginner

advanced understanding how to improve

expert learn how to teach others,
improve upon what the software is doing

L05 Documentation

Markus Raab

Institute of Information Systems Engineering, TU Wien

This work is licensed under a Creative Commons
“Attribution-ShareAlike 4.0 International” license.



Generate

1 Introduction

2 **Generate**

3 Verify

4 Meeting

- Recapitulation
- Assignments
- Preview

Markdown

Minimal formatting abilities but implemented by many tools

- [Link](/linktarget) or [Link](relative/link)
- *italics* and **bold**
- 'inline code' or ' ' code fences
- - item
 - item
 - item

Views

The same (markdown) file can be viewed via:

- directly viewing the source `doc/help/kdb.md`
- website `https://www.libelektra.org/manpages/kdb` (rendered by marked)
- API docu
`https://doc.libelektra.org/api/master/html/doc_help_kdb_md.html`
(rendered by doxygen)
- `man kdb` (rendered by roff)
- `kdb -help` or `kdb help <command>`
- GitHub `https://master.libelektra.org/doc/help/kdb.md`
- From qt-gui (rendered by discount)

→ 5 different markdown renderer involved

Plugins

For plugins, documentation even changes the build process:

```
1 -_infos=_Information_about_the_toml_plugin_is_in_keys_below
2 -_infos/author=_Jakob_Fischer_<jakobfischer93@gmail.com>
3 -_infos/licence=_BSD
4 -_infos/provides=_storage/toml
5 -_infos/needs=_base64
6 -_infos/recommends=_type
7 -_infos/placements=_getstorage_setstorage
8 -_infos/status=_experimental_unfinished
9 -_infos/metadata=_order_comment/#_comment/#/start_comment/#/s
10 -_infos/description=_This_storage_plugin_reads_and_writes_TOML
```

Conclusion

- reuse of documentation by generation
- avoids duplication and errors
- avoids CI checks for inconsistencies
- generation by CI

L05 Documentation

Markus Raab

Institute of Information Systems Engineering, TU Wien

This work is licensed under a Creative Commons
“Attribution-ShareAlike 4.0 International” license.



Verify

- 1 Introduction
- 2 Generate
- 3 Verify**
- 4 Meeting
 - Recapitulation
 - Assignments
 - Preview

Goals

Not every documentation-related task can be generated:

- in the text we want to refer to the behavior
- we want to verify if given examples are correct

Problems with Unit Tests

- ① difficult to read
- ② code cannot be directly copied (asserts)
- ③ cannot be easily integrated in tutorials

→ specialized verification language for documentation

```
1 0000 ' ' sh
2 0000 kdb_set_user:/tests/something
3 0000 #_RET:_0
4
5 0000 kdb_get_user:/tests/something
6 0000 ' ' '
```

Syntax

- ① starts with `'''sh`
- ② comments introduce checks
- ③ otherwise is shell code to be executed
- ④ `#>` verifies stdout output
- ⑤ `# RET:` verifies return code (if not 0)

Conventions

- ① Test data in /tests.
- ② Generate temporary files if needed or use [HERE](#).

Conclusions

- ① If possible, generate.
- ② Otherwise, verify.
- ③ Keep user and purpose in mind.

L05 Documentation

Markus Raab

Institute of Information Systems Engineering, TU Wien

This work is licensed under a Creative Commons
“Attribution-ShareAlike 4.0 International” license.



Meeting

- 1 Introduction
- 2 Generate
- 3 Verify
- 4 **Meeting**
 - Recapitulation
 - Assignments
 - Preview

Recapitulation.



Feedback

- Feedback Talk



L05 Documentation