

Diese Leseprobe haben Sie beim
 **edv-buchversand.de** heruntergeladen.
Das Buch können Sie online in unserem
Shop bestellen.
[Hier zum Shop](#)

Auf einen Blick

1	Git in zehn Minuten	13
2	Learning by Doing	23
3	Git-Grundlagen	77
4	Datenanalyse im Git-Repository	157
5	GitHub	181
6	GitLab	217
7	Azure DevOps, Bitbucket, Gitea und Gitolite	245
8	Workflows	267
9	Arbeitstechniken	291
10	Git in der Praxis	323
11	Git-Probleme und ihre Lösung	355
12	Kommandoreferenz	377

Inhalt

Vorwort	9
1 Git in zehn Minuten	13
1.1 Was ist Git?	13
1.2 Software von GitHub herunterladen	16
1.3 Programmieren lernen mit Git-Unterstützung	18
2 Learning by Doing	23
2.1 git-Kommando installieren	23
2.2 GitHub-Account und -Repositories einrichten	31
2.3 Mit dem Kommando »git« arbeiten	37
2.4 Authentifizierung	50
2.5 Git spielerisch lernen (Githug)	61
2.6 Entwicklungsumgebungen und Editoren	63
2.7 An einem fremden GitHub-Projekt mitarbeiten	72
2.8 Synchronisation und Backups	74
3 Git-Grundlagen	77
3.1 Nomenklatur	77
3.2 Die Git-Datenbank	82
3.3 Commits	86
3.4 Commit-Undo	94
3.5 Branches	102
3.6 Merge	108
3.7 Stashing	116
3.8 Remote Repositories	118
3.9 Merge-Konflikte lösen	129
3.10 Rebasing	136
3.11 Tags	142
3.12 Referenzen auf Commits	147
3.13 Git-Interna	152

4	Datenanalyse im Git-Repository	157
4.1	Commits durchsuchen (git log)	157
4.2	Dateien durchsuchen	168
4.3	Fehler suchen (git bisect)	173
4.4	Statistik und Visualisierung	175
5	GitHub	181
5.1	Pull-Requests	182
5.2	Actions	187
5.3	Paketmanager (GitHub Packages)	195
5.4	Automatische Sicherheits-Scans	201
5.5	Weitere GitHub-Funktionen	205
5.6	GitHub CLI	210
5.7	Codespaces	214
6	GitLab	217
6.1	On Premises versus Cloud	218
6.2	Installation	219
6.3	Das erste Projekt	226
6.4	Pipelines	228
6.5	Merge-Requests	238
6.6	Web-IDE	241
6.7	GitPod	242
7	Azure DevOps, Bitbucket, Gitea und Gitolite	245
7.1	Azure DevOps	245
7.2	Bitbucket	250
7.3	Gitea	252
7.4	Gitolite	263
8	Workflows	267
8.1	Anweisungen für das Team	267
8.2	Solo-Entwicklung	268
8.3	Feature-Branches für Teams	270
8.4	Merge/Pull-Requests	277

8.5	Long-running Branches – Gitflow	281
8.6	Trunk-based Development	286
8.7	Welcher Workflow ist der Richtige?	289
9	Arbeitstechniken	291
9.1	Hooks	291
9.2	Prägnante Commit-Messages	297
9.3	Submodule und Subtrees	304
9.4	Mehr Komfort in Bash und Zsh	315
9.5	Zwei-Faktor-Authentifizierung	317
10	Git in der Praxis	323
10.1	Etckeeper	324
10.2	Dotfiles mit Git verwalten	327
10.3	Zugriff auf Subversion mit git-svn	334
10.4	Von SVN zu Git migrieren	338
10.5	Ein Blog mit Git und Hugo	343
11	Git-Probleme und ihre Lösung	355
11.1	Git-Fehlermeldungen (Ursache und Lösung)	356
11.2	Leere Verzeichnisse speichern	363
11.3	Merge für eine einzelne Datei	364
11.4	Dateien permanent aus Git löschen	365
11.5	Ein Projekt aufteilen	373
11.6	Commits in einen anderen Branch verschieben	374
12	Kommandoreferenz	377
12.1	git-Kommando	377
12.2	Revisionssyntax	409
12.3	git-Konfiguration	410
	Index	417