Ruby on Rails Tutorial – Notizen

# Neue Applikation anlegen

## Applikation erstellen

### Applikation anlegen

* cd */path/to/rails\_projects*
* rails new *appname*

### prepare the Gemfile & install gems

* edit /path/to/rails\_projects/appname/Gemfile
* bundle install

### .gitignore anpassen

.bundle

db/\*.sqlite3\*

log/\*.log

\*.log

/tmp/

doc/

\*.swp

\*~

.project

.DS\_Store

# Versionskontrolle mit Git

## Globale Konfiguration

|  |  |
| --- | --- |
| * git config –-global user.name “Firstname Lastname” * git config –-global user.email admin@railstutorial.org | Globales Setup (nur einmal nach der Installation nötig) |

## Repository anlegen

|  |  |
| --- | --- |
| * git init | Neues Repository im aktuellen Verzeichnis erstellen |
| * git remote add origin [git@github.com:<username>/<appname>.git](mailto:git@github.com:%3cusername%3e/%3cappname%3e.git) | Remote Repository festlegen |

## Repository verwenden

|  |  |
| --- | --- |
| * git add . | Alle Dateien des aktuellen Verzeichnisses dem Repository hinzufügen |
| * git commit -am "Comment" | Commit aller Änderungen |
| * git push origin master | Master-Branch zum Remote-Repository pushen (Nur beim 1. Mal) |
| * git push | Aktuellen Branch zum Remote-Repository pushen |
| * git pull origin master | Letze Änderungen vom Repository laden |
| * git checkout –f | Alle Änderungen rückgängig machen |
|  |  |
| * git mv oldfile newfile | Datei umbenennen |

## branching & merging

|  |  |
| --- | --- |
| * git checkout –b *new-branch-name* | Neuen Branch anlegen |
| * git checkout *existing-branch-name* | Zu vorhandenem Branche wechseln |
| * git branch –d branch-name | Branche löschen (sicher, prüft ob alles gemerged ist) |
| * git branch –D branch-name | Branche löschen (forced, löscht nichtgemergte Änderungen ohne Rückfrage |
| * git checkout master * git merge *branch-name* | Branch in den Master mergen (vorher committen) |

## Basisworkflow Beispiel

|  |  |
| --- | --- |
| Neuen Branch anlegen | * git checkout –b *new-branch-name* |
| Änderungen vornehmen, testen usw. | |
| Neue Dateien hinzufügen | * git add . |
| Commit aller Änderungen | * git commit -am "Comment" |
| zum Remote-Repository pushen | * git push |
| Branch in den Master mergen & löschen | * git checkout master * git merge *branch-name* * git merge –d *branch-name* |

## Nützliche Befehle

|  |  |
| --- | --- |
| * git status | Status des aktuellen Repositories anzeigen |
| * git branch | Liste der branches anzeigen |
| * git log | Liste der Commits |