

API - Programmierung Allgemein

Dienstag, 4. Mai 2021 13:14

Lernjournal API Programierung Allgemein[vm]

Datum	Kompetenz, Thema	Gelernte Inhalte, Vorgehen, Probleme & Lösungen, Bemerkungen
2021-mm-dd		
2021-09-13	Git	<p>Was ist Git?</p> <p>Git ist eine Software zur Versionsverwaltung, welche jeder Programmier früher oder später benützt. Mit Git können mehrer Leute an einem Quellcode arbeiten und am Ende ihre Ergebnisse mittels "merge" zusammenführen.</p> <p>Workflow</p> <p>Ein Git-Workflow ist eine Empfehlung wie man Git verwendet oder auch ein "Rezept, also eine Art und Weise wie man Git verwenden soll. Hier zum Beispiel werden Workflows verglichen: https://www.atlassian.com/de/git/tutorials/comparing-workflows</p> <p>Commit und Push</p> <p>Commit's sind Snapshots vom Projekt, also die erfassen den Status des Projektes vom aktuellen Projekt.</p> <p>Nachdem man etwas Lokal an einem Projekt geändert hat, kann man via Push den anderen Teammitglieder die Änderungen freigeben.</p> <p>Git-Befehle</p> <p>Installierte Version überprüfen git --version</p> <p>Nutzername und Email-Adresse hinzufügen</p> <p>Nutzername hinzufügen git config --global user.name "IHR_BENUTZERNAME"</p> <p>Nutzername anzeigen git config --global user.name</p> <p>Email-Adresse hinzufügen git config --global user.email "adresse@example.com"</p> <p>Email-Adresse anzeigen git config --global user.email</p> <p>Alle Infos anzeigen git config --global --list</p> <p>Mit Projekten arbeiten</p> <p>Repository herunterladen git clone SSH / URL_HIER_EINFUEGEN</p> <p>Mit dem status-Befehl werden alle Dateien angezeigt, bei denen im Vergleich zum letzten Commit eine Änderung registriert wurde. git status</p> <p>Alle Änderungen zum Index hinzufügen git add.</p> <p>Mit dem Befehl: "git branch -a" kann man alle Verzweigungen der Repository anzeigen git branch -a</p> <p>Mit dem Befehl: "git log" kann man eine Liste der letzten Commits anzeigen, dann sieht man genau, wer, wann etwas gemacht hat. git log</p> <p>3 Befehle um Änderungen aus dem entfernten Repository und mit ihren lokalen Dateien zu synchronisieren.</p> <p>fetch holt Änderungen aus dem entfernten Repository, aber wendet diese nicht auf Ihren Code an. git fetch</p> <p>merge synchronisiert per fetch abgeholte Änderungen mit dem Workspace.</p> <p>pull führt fetch und merge aus. Dadurch werden Änderungen aus dem entfernten Repository abgeholt und mit dem Workspace synchronisiert git pull PFAD BRANCH</p> <p>Eine neue Datei Repository pushen (hochladen)</p> <p>Als Erstes muss man die Datei dem Index hinzufügen dies macht man mit diesem Befehl: git add index.html</p> <p>Als Zweites mnuss man festhalten was genau neu ist, dies macht man mit diesem Befehl: git commit -m "Commit Nachricht"</p> <p>Als aller Letztes muss man folgenden Befehl ausführen: git push</p> <p>Working-Tree</p> <p>Ein Working-Tree ist dafür da, mehrere Arbeitsverzeichnisse zu managen, welche alle zu einem Repository gehören. Wenn man mit einem Working-Tree arbeiten, dann kann man an mehrere Ordner arbeiten.</p>

Frage:

Für folgende Frage: [Führen Sie die git status und git branch -a](#), bekam ich folgendes Ergebnis:

Wenn ich "git branch -a" eingebe, bekomme ich folgenden Output:

```
* develop
main
remotes/origin/HEAD -> origin/main
remotes/origin/develop
remotes/origin/main
```

Aufgabe 2c Branches and Checkout

Frage 1:

Wenn ich "git status" eingebe, bekomme ich folgenden Output:

On branch develop

Your branch is up to date with 'origin/develop'.

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git checkout -- <file>..." to discard changes in working directory)

Wenn ich "git branch -a" eingebe, bekomme ich folgenden Output:

```
* develop
main
remotes/origin/HEAD -> origin/main
remotes/origin/develop
remotes/origin/main
```

Frage 2:

Mit dem Befehl: "git checkout develop" komme ich zum develop branch, dies ist dann der Output:

Switched to branch 'develop'

M index.html

Your branch is up to date with 'origin/develop'.