**Git Grundlagen 2**

Git benachrichtigt Sie auch darüber, dass Sie zu einem anderen Zweig gewechselt sind: "Zu Zweig 'Benutzerprofil' gewechselt". Lassen Sie uns noch einmal "git branch" ausführen, um zu beweisen, dass:

|  |  |
| --- | --- |
|  | Meister |
|  | \* Benutzerprofil |
|  |  |

**Nach Abschluss der Entwicklung eines Features dieses von diesem Entwicklungszweig auf den Master verschieben? Muss ich kopieren und einfügen? Anonymer Entwickler**

Bevor wir die Fragen beantworten, werfen wir zunächst einen Blick auf den Ablauf beim Hinzufügen neuer Zweige:

1. Erstellen Sie eine neue Verzweigung, um mit "git branch <Verzweigungsname>" eine neue Funktion zu entwickeln.
2. Wechseln Sie mit "git checkout <branch-name>" vom Hauptzweig zum neuen Zweig.
3. Entwickeln Sie die neue Funktion.
4. Du steckst im dritten Schritt fest. Was solltest du als nächstes tun? Die Antwort ist einfach: Sie müssen den Befehl "merge" verwenden. Um einen sekundären Zweig in den Hauptzweig (der ein Master-, Entwicklungs- oder Feature-Zweig sein kann) einzufügen, wechseln Sie zunächst zurück in den Hauptzweig. In unserem Fall sollten wir den Hauptzweig auschecken:

|  |
| --- |
| $ git Checkout Master |
|  | \* Meister |
|  | Benutzerprofil |

Der aktuelle Zweig wird jetzt in master geändert und der Benutzerprofilzweig kann mit dem Befehl "merge" zusammengeführt werden:

$ git merge user-profile

Denken Sie daran, dass Sie sich in der Hauptniederlassung befinden und eine andere Niederlassung in der Hauptniederlassung zusammenführen - nicht umgekehrt! Da sich die Benutzerprofilfunktion nun im Hauptzweig befindet, wird der Benutzerprofilzweig nicht mehr benötigt. Führen wir also den folgenden Befehl aus:

|  |  |
| --- | --- |
|  | $ git branch -d Benutzerprofil |

Mit der Option "-d" können wir das nun nicht mehr benötigte "Benutzerprofil" löschen. Übrigens, wenn Sie versuchen, den Zweig zu entfernen, in dem Sie sich befinden, lässt Git Sie nicht:

|  |  |
| --- | --- |
|  | Fehler: Der Zweig ' Benutzerprofil ', in dem Sie sich gerade befinden, kann nicht gelöscht werden. |

Anstatt zwei Befehle auszuführen, können Sie nur einen ausführen:

|  |  |
| --- | --- |
|  | $ git checkout -b Admin-Panel |

**Branch auf github über terminal löschen:**

git push origin --delete branchName