

Nutzen Sie für folgende Übung als Input:

VL-Skript svnbook.pdf

Die Übung nutzt eine hosting plattform für das svn Repository.
Sie können auch einen eigenen svn Server lokal installieren und die Übung mit diesem Server durchführen.

Nutzen Sie dafür die Beschreibung in den Vorlesungsfolien.

SVN Übung

1. Legen Sie einen oder mehrere Accounts bei einer Hosting Plattform für svn an. (Bsp: <https://riouxsvn.com/>)
2. Loggen Sie sich mit dem erzeugten Account auf der Plattform ein.
3. Legen Sie ein svn repository auf der Hosting Plattform an. Nutzen Sie die Möglichkeit, mit dem Erstellen gleich die Struktur für trunk, tags und branches mit anzulegen.
4. Installieren Sie einen lokalen svn Client bei sich am Rechner.
5. Konfigurieren Sie den Client so, dass er auf das erstellte Repository zugreift. (nicht nötig bei Kommandozeilen Client)

6. Erstellen Sie ein lokales Arbeitsverzeichnis.
7. Checken Sie die Daten aus dem svn repository in Ihr lokales Verzeichnis aus.
8. Erstellen Sie neue Dateien und Verzeichnisse in dem Arbeitsverzeichnis und bringen Sie diese in das svn Repository.
9. Löschen Sie eine Datei und übertragen Sie diese Information ins repository.
10. Legen Sie einen zweiten Arbeitsbereich unabhängig vom ersten an.

11. Checken Sie die Daten des Repositories in den zweiten Arbeitsbereich aus.
12. Testen Sie das Zusammenspiel der beiden Arbeitsbereiche, indem Sie Dateien in einem Bereich ändern, ins repository bringen, im anderen Bereich aus dem Repository holen. Erzeugen Sie Konflikte durch gleichzeitige Änderungen an Dateien und lösen Sie diese Konflikte.

SVN Übung - howto

Die Übung nutzt eine hosting plattform für das svn Repository. Sie können auch einen eigenen svn Server lokal installieren und die Übung mit diesem Server durchführen. Nutzen Sie dafür die Beschreibung in den Vorlesungsfolien.

1. Legen Sie einen oder mehrere Accounts bei einer Hosting Plattform für svn an. (Bsp: <https://riouxsvn.com/>)

Folgen Sie den Anweisung auf der Plattform. Für diese Übung ist es nicht nötig, mehrere Accounts anzulegen. Wenn Sie die Rechtevergabe testen wollen, benötigen Sie mehr als einen Account.

2. Loggen Sie sich mit dem erzeugten Account auf der Plattform ein.
Erklärt sich selbst.

SVN Übung - howto

3. Legen Sie ein svn repository auf der Hosting Plattform an. Nutzen Sie die Möglichkeit, mit dem Erstellen gleich die Struktur für trunk, tags und branches mit anzulegen.

Folgen Sie dafür den Möglichkeiten der Hosting Plattform.

4. Installieren Sie einen lokalen svn Client bei sich am Rechner.

Wenn Sie svn von der Apache Seite <https://subversion.apache.org/packages.html> installieren, ist der Kommandozeilen Client bereits enthalten. Ansonsten installieren Sie einen beliebigen svn Client.

5. Konfigurieren Sie den Client so, dass er auf das erstellte Repository zugreift.

Beim Kommandozeilenclient nicht nötig.

6. Erstellen Sie ein lokales Arbeitsverzeichnis.

Selbsterklärend

7. Checken Sie die Daten aus dem svn repository in Ihr lokales Verzeichnis aus.

In der Kommandozeile in dem kreierte lokalen Verzeichnis:

svn checkout <repository url>

8. Erstellen Sie neue Dateien und Verzeichnisse in dem Arbeitsverzeichnis und bringen Sie diese in das svn Repository.

- *Neue Dateien und Verzeichnisse lokal erstellen*
- *svn add <Dateiname> <Dateiname> ...*
- *svn commit <Dateiname> <Dateiname> ...*

9. Löschen Sie eine Datei und übertragen Sie diese Information ins repository.

- *Datei lokal löschen.*
- *Auf Verzeichnis svn commit <Verzeichnisname>*

10. Legen Sie einen zweiten Arbeitsbereich unabhängig vom ersten an.

Selbsterklärend

11. Checken Sie die Daten des Repositories in den zweiten Arbeitsbereich aus.

In der Kommandozeile in dem kreierte lokalen Verzeichnis:

svn checkout <repository url>

Svn Übung - howto

12. Testen Sie das Zusammenspiel der beiden Arbeitsbereiche, indem Sie Dateien in einem Bereich ändern, ins repository bringen, im anderen Bereich aus dem Repository holen. Erzeugen Sie Konflikte durch gleichzeitige Änderungen an Dateien und lösen Sie diese Konflikte. Nutzen Sie die svn Befehle update, add, delete, status, move