

Zadanie 1. (Termin wykonania: 2018-10-28)

Do każdego z wykonanych zadań należy wykonać sprawozdanie (plik tekstowy) w postaci skopiowanych odpowiednich fragmentów tekstów z terminala, zawierających odpowiednie wykonane polecenia wraz z wyświetlonymi przez nie komunikatami, oraz zawartości przetwarzanych plików (można używać polecenia `'cat'` do wyświetlania ich na terminalu).

Sprawozdania mają być umieszczone w dodatkowym katalogu w repozytorium svn (będą wykorzystywane przy prezentacji rozwiązania).

Używając polecenia `'svn'`, należy wykonać następujące pod-zadania na przydzielonym sobie repozytorium svn:

A) (0.5 p.) Utworzyć dwie kopie robocze własnego repozytorium. Na przykładowym pliku tekstowym doprowadzić do konfliktu, a następnie ręcznie rozwiązać kolizję.

W sprawozdaniu zawrzeć, oprócz wykonywanych poleceń, zawartości plików ze znacznikami kolizji.

B) (0.5 p.) Utworzyć we własnym repozytorium poddrzewa katalogów o różnych głębokościach, a następnie utworzyć kopie robocze, w których zostanie zademonstrowane wykorzystanie `"sparse directories"` do sprowadzenia różnych wybranych fragmentów podkatalogów.

Do wyświetlania drzew katalogów do sprawozdania można użyć polecenia `'tree'`.

C) (0.5 p.) Umieścić w repozytorium przykładowy plik HTML a następnie sprawdzić jaki ma wpływ ustawienie własności `'svn:mime-type'` na wartość `"text/html;charset=UTF-8"` (albo inny charset).

D) (1 p.) Utworzyć we własnym repozytorium podkatalog (np. o nazwie `"externals"`) i ustawić na nim własność (property) `'svn:externals'`, tak aby w odpowiednim podkatalogu znalazła się kopia robocza repozytorium zewnętrznego:

<http://156.17.7.16/info/>

Sprawdzić na dodatkowych kopiach roboczych jak działają `"svn up"`, `"svn co"` bez opcji `"--ignore-externals"` oraz z tą opcją.

E) (1 p.) Przetestować zakładanie, usuwanie, zrywanie i

podkradanie blokady pliku ('svn lock' i związane z nim polecenia).

F) (1 p.) W następujących po sobie kolejnych "commitach":

- dodać dwa pliki o różnych zawartościach: a.txt i b.txt (svn add ...)
- usunąć b.txt (svn delete ...)
- dopisać wiersz tekstu do a.txt i przenieść a.txt do b.txt (svn move ...)
- dopisać wiersz tekstu do b.txt i przenieść b.txt do c.txt (svn move ...)
- dodać nowy plik b.txt (svn add ...)

Wyświetlić historię każdego obiektu i historię całego podkatalogu (svn log -v ...). Wykorzystując "peg-revision" i "operative revision" wyświetl zawartości pliku c.txt i wszystkich jego wcześniejszych wersji.

G) (1 pkt.) ZADANIE "TECHNICZNO-ADMINISTRACYJNE" (To zadanie musi być wykonane, gdyż umożliwi dostęp do własnego repozytorium GIT na serwerze.)

Przy pomocy polecenia `ssh-keygen` wygenerować parę kluczy:

klucz prywatny i klucz publiczny.

Klucz publiczny (ten z końcówką '.pub') należy skopiować do pliku o nazwie `<numer-indeksu>.pub` i umieścić w katalogu głównym **własnego repozytorium svn na serwerze 156.17.7.16.**

Nie pomylić z kluczem prywatnym, którego nie wolno nikomu ujawniać!

Klucz prywatny musi się znajdować w podkatalogu `.ssh` katalogu domowego, na prywatnych komputerach studenta, z których będzie on mieć dostęp do repozytoriów GIT na serwerze.

UWAGA: WIELU STUDENTÓW DOSTARCZAŁO KLUCZ W NIEODPOWIEDNIM FORMACIE.

NAJLEPIEJ WYKONAĆ POLECENIE: `ssh-keygen` BEZ ŻADNYCH PARAMETRÓW.

W PODKATALOGU `.ssh/` KATALOGU `$HOME`

ZOSTANĄ UTWORZONE PLIKI: `id_rsa id_rsa.pub`

PLIK: `id_rsa.pub` WYGLĄDA PODOBNIIE JAK TO:

`ssh-rsa`

AAAAB3NzaC1yc2EAAAADAQABAAQCTJbx5zETMKOd9dMqv1NngWtPMrX4eQLxhe
o3HSOyKBcmfSY1ib8zrEEBK8fzJrnlRILq1aE/iEgqTgXsCNx7MYRMpYsZbMEfOAlxXLxp0
O8II5yjaYrchH1lheaEM/PZsok8co8L92dGIPCTwRE+VI809+SWWG93RHJVK8Mfx/d2UWjpv
p218NMqjQX7aReTav3vys9pIld7G2HIWWskjsiVvrOGYTI/0GPeqP29TkwwEegEPEllx63m
epIb+EAwJElFFI6o9PUgwSENDXT2hyHWHs9Lx3C3peHxu9scj6tidk/hGNN4+5c3w2gNB
mEn3Mg2APE7kBB26I0f kik@kik-Latitude-D600

(JEDNA DŁUUUUUGA LINIJKA)

NALEŻY SWÓJ PLIK [id_rsa.pub](#) UMIEŚCIĆ W KATALOGU **GŁÓWNYM** SWOJEGO REPOZYTORIUM.

NALEŻY TEŻ USUNĄĆ Z KATALOGU GŁÓWNEGO SWOJEGO REPOZYTORIUM INNE PLIKI I KATALOGI O KOŃCÓWCE NAZWY: [.pub](#)*