Exercise Set 2

Bash Scripting

University of Oslo - IN3110/IN4110

Fall 2021

Note: These exercises are not mandatory. You don't have to put your work in your Github repository.

Exercise 1: A Simple Script

Create a script that takes a name, and an age. Output these values, along with how many words and letters are in the name. Also add ten years to the age, and comment on whether or not the person will be old in ten years.

Some useful commands are `wc`, and perhaps `bc` or `expr` for calculations. There are several other ways to do calculations.

Exercise 2: Checking for Network Connection

`ping` is a program that sends a simple request to a network host and tells us the time it took to get a response. We are going to use this to monitor our web connection.

Create a script that uses `ping` to send a request to `google.com` (or any other website), and based on the output tells us whether or not we are connected to the internet. (Hint: use the `$?` variable to get the status of the previously run command). Like most other programs `ping` returns 0 when it runs successfully, i.e. when we have a connection.

By default, ping sends requests until you tell it to stop. However, you can also tell it to send only one request. To avoid polluting your terminal with output from `ping`, redirect both stdout and stderr to `/dev/null`, which will discard the output.

Use `watch` to make your script run once every second, and try to disconnect and connect to the internet to see if your script works.

Exercise 3: A Simple Journal

Some people find it useful to make small journal entries throughout the day, to keep track of how they spend their time. Create a script that takes a string MESSAGE and appends the following line to a file.

```
YYYY-MM-DD HH:MM:SS - MESSAGE
```

The first part of the string is the date and time for when the log was written.

Also, have a log command that makes your script go through the file and count the number of entries for each day.

Some useful programs you can use in this script are `date`, `cut` and `uniq`.

Thumbs up to yourself - You just completed week 2!