

Oppgaver: Bash-scripts

Viktig: Kjør scriptene kun i en VM (Virtual Machine), for å være sikker på at viktige filer ikke blir påvirket av scriptene.

Oppgave 1: Passord-generator

Lag et bash-script som genererer et passord, med en lengde oppgitt av brukeren.

Krav:

- ☐ Bruker skal oppgi lengde på passordet generert
 - En passende feilmelding burde oppgis om brukeren oppgir noe annet enn et tall
- ☐ Passordet skal være tilfeldig generert, og bestå av tall [0-9], småbokstaver [a-z], storebokstaver [A-Z]
 - Bonus: tegn [@, \$, (, etc]
- ☐ Vis det genererte passordet i terminalen

Bonus funksjon:

- ☐ Spør brukeren om passordet skal lagres til fil
- ☐ Om ja:
 - Krypter passordet med ccrypt og lagre det i et `passord.txt.cpt` dokument
- ☐ Om nei:
 - Skriv en passende melding som varsler brukeren at passordet ikke er blitt lagret

Levering:

- Last opp til et Github-repo

- Gi en kort beskrivelse av hvordan koden fungerer i en [README.md](#)-fil

Oppgave 2: Liste opp og sortere mapper

- ☐ Lag et bash-script som tar banen (path) til en mappe som argument
- ☐ Scriptet skal så liste opp alle filer og undermapper som ligger der, med navnet på den originale mappen i toppen

Ekstra utfordring:

- ☐ Aksepter 'a' eller 'd' som et argument, som sorterer listen økende (ascending) eller synkende (decending)
- ☐ La bruker oppgi mapp-banen til flere mapper, og list opp for alle de gitt mappene

Levering:

- Last opp til et Github-repo
- Gi en kort beskrivelse av hvordan koden fungerer i en [README.md](#)-fil

Oppgave 3: Skriv og les fra fil

Lag et Bash-script som spør brukeren om de ønsker å skrive eller lese fra en .txt-fil

Skriv til fil:

- ☐ Bruker starter scriptet med argument 'w'
- ☐ Brukeren blir promptet til å skrive hva som skal lagres til fil
- ☐ Brukeren blir bedt om å oppgi et navn på filen, filen lagres i mappen terminalen kjører fra
- ☐ **Ekstra utfordring:** Filen lagres i en mappe oppgitt av brukeren

Les fra fil:

- ☐ Bruker starter scriptet med argument 'r' og fil-bane til .txt-filen som skal leses
- ☐ Innholdet skal skrives ut til skjerm

Levering:

- Last opp til et Github-repo
- Gi en kort beskrivelse av hvordan koden fungerer i en [README.md](#)-fil

Oppgave 4: System-overvåkning

Lag et Bash-script som overvåker system-ressurser

Krav:

- ☐ Logg dato og tidspunkt
- ☐ Logg CPU-bruk
- ☐ Logg minne-bruk
- ☐ Logg disk-plass
- ☐ Skriv resultatene til en fil

Ekstra utfordring:

- ☐ Scriptet skal kjøre jevnlig hver 30 minutt
- ☐ Ny logg-data skal ikke overskrive eksisterende data i fil

Levering:

- Last opp til et Github-repo
- Gi en kort beskrivelse av hvordan koden fungerer i en [README.md](#)-fil