Sammendrag av innhold fra gruppetime 1

Frøya Lydersen

skagel@ifi.uio.no

Innhold

1. Terminal
2. Python

Terminal

På samme måte som filutforsker, kan terminalen “være” et sted på filsystemet.
Derfor trenger vi noen kommandoer for å navigere rundt på filsystemet.

Vanlige kommandoer

- `ls`, eller `ls -l` - list innhold i mappen som terminalen er i.
- `cd` 🧀 - gå til en annen mappe.
- `pwd` - se absoluttsti til der du er på filsystemet.
- `python` 🧀 - kjør et pythonprogram

Her har jeg erstattet stier og filnavn med 🧀 oster.

Variabler/tilordninger & utskrivning

Vi kan skrive ut tekst fra vår kode ved å kalle på `print`:

```
print("Jeg liker ikke ostepop !")
```

Vi kan “tilordne” en verdi (f.eks. tallet 5, eller teksten “frosk”) til en variabel:

```
navn = "Frøya"
alder = 21
```

Lese inn fra bruker

Vi kan lese inn tekst fra den som kjører koden vår med `input`, og så tilordne det til en variabel:

```
interesse = input("Hva er din interesse? ")
```

Så kan vi skrive det tilbake ut:

```
print(interesse)
```

Kommentarer

Hvis vi starter en linje med tekst med `#` så blir det til en “kommentar”. Kommentarer har ingen bæring på kjøringen av programmet, og er så langt dere trenger å bry dere, noe mennesker skriver kun for at andre mennesker skal lese.

```
# yo yo jeg heter frøya og jeg progga denne episke koden
print("Jeg liker ikke ostepop !") # bruker print for å skrive ut...
# 5u39jgdlk"U)("%U328dsjgmwfanai&u awjhglasgdhsd...
```

Programmet ovenfor gjør nøyaktig det samme som det første programmet som ble vist. Dvs. kommentarene har ingen effekt.

Slå sammen strenger

Vi kan bruke `+` for å slå sammen to strenger til en streng:

```
x = "ost"
y = "pepsi"
z = x + y  # "ostpepsi" <-- merk ingen mellomrom
```

Merk at `+` slår sammen strengene uten hensyn til hvordan mennesker leser teksten, altså legger den ikke inn mellomrom. Vi som programmerere må her være ekstremt presis i hva vi ønsker. Ønsker vi mellomrom må vi enten legge inn mellomrom i slutten av x eller starten av y, eller så må vi legge til en streng som bare inneholder et mellom, så det blir: `z = x + " " + y`

Slå sammen strenger

Vi kan også få `print` til å slå strengene sammen for oss:

```
x = "ost"
y = "pepsi"
print(x,y)  # "ost pepsi" <-- merk mellomrom
```

Dette funker på en helt annen måte enn `+`. Her er `,` bare skillet mellom to forskjellige verdier som vi "gir" til `print`, som da putter et mellomrom i mellom hver ting den mottar, før den skriver det ut.