

Oppgaver 1

September 2, 2024

1 Oppgaver 1

1.1 Åpne RStudio. Hva er konsollen og hva er miljøvinduet? Bruk konsollen som kalkulator og gjør noen regnestykker.

```
[ ]: 1+1
```

1.2 I konsollen: definer objektet `mitt_tall` hvor du lagrer yndlingstallet ditt. Hva skjer i miljøvinduet når du gjør dette?

```
[ ]: mitt_tall <- 1
```

1.3 Skriv `mitt_tall` i konsollen. Hva skjer?

```
[ ]: mitt_tall
```

1.4 Definer et annet objekt (finn på navnet selv) og gjør noen regnestykker med de to objektene du nå har definert.

```
[ ]: nytt_tall <- 2  
  
mitt_tall+nytt_tall
```

1.5 Definer et tredje objekt som summen av de to første objektene,

```
[ ]: sum_tall <- mitt_tall+nytt_tall
```

1.6 Trykk på feiekosten i miljøvinduet. Hva skjer?

```
[ ]:
```

1.7 Skriv `mitt_tall` i konsollen nå. Hva skjer?

```
[ ]: mitt_tall
```

1.8 Opprett et nytt R skript fra nedtrekksmenyen.

[]:

1.9 I skriptet definerer du de samme objektene som i oppgavene 2-4 (mitt_tall + egendefinert objekt). Kjør skriptet. Hva skjer?

[]:

1.10 Lagre skriptet som my_first_Rskript.R eller noe lignende.

[]:

1.11 Lukk skriptet, trykk på feiekosten i miljøvinduet. Kall på de definerte objektene i konsolen. Hva skjer?

[]:

1.12 Åpne skriptet du nettopp lagde og kjør det på nytt.

[]:

1.13 Definer et objekt med fire desimaler.

[]:

```
tall_desimaler <- 1.2345
```

1.14 Bruk funksjonen round() til å runde av dette objektet til ingen, én og to desimaler

[]:

```
round(tall_desimaler, digits = 0)
round(tall_desimaler, digits = 1)
round(tall_desimaler, digits = 2)
```

1.15 Hva skjer hvis du skriver og kjører `3 > 2` i scriptet? Hva med `3 == 2`? Og `3 < 2`? Hvilken type objekt har disse? (`class()`)

[]:

```
3 > 2
3 == 2
3 < 2

class(3 > 2)
```

1.16 Hva skjer hvis du skriver $3 = 2$?

```
[ ]: # 3 = 2
```

1.17 Lagre et objekt som er en vektor med tre tall.

```
[ ]: vektor_tall <- c(1, 2, 3)
```

1.18 Hent det tredje elementet i vektorene.

```
[ ]: vektor_tall[3]
```

1.19 Finn lengden på vektoren ved å bruke funksjonen `length()`

```
[ ]: length(vektor_tall)
```

1.20 Hva skjer hvis du anvender funksjonen `sum()` på vektoren?

```
[ ]: sum(vektor_tall)
```

1.21 Opprett en annen vektor, også denne med tre tall.

```
[ ]: vektor_tall_2 <- c(4, 5, 6)
```

1.22 Gjør noen regneoperasjoner med vektorene. Hva skjer?

```
[ ]: vektor_tall_2+1  
vektor_tall_2*2  
  
sum(vektor_tall_2)+1
```

1.23 Definer en vektor med fire tall. Hva skjer hvis du prøver å legge sammen vektoren med fire tall med en av de vektorene med tre tall?

```
[ ]: vektor_tall_4 <- 1:4  
dd  
# vektor_tall_4+vektor_tall_2
```

1.24 Kombiner de to vektorene til én vektor

```
[ ]: c(vektor_tall_2, vektor_tall_4)
```

1.25 Opprett vektoren statsminister med koden nedenfor. Bruk deretter funksjonene class(), length(), unique() og table() på dette objektet. Hva forteller disse?

```
[ ]: statsminister <- c("Brundtland", "Willoch", "Brundtland", "Syse", "Brundtland",  
  ↪ "Jagland", "Bondevik", "Stoltenberg", "Bondevik", "Stoltenberg", "Solberg",  
  ↪ "Støre")  
  
class(statsminister)  
length(statsminister)  
unique(statsminister)  
table(statsminister)
```

1.26 Hva forteller denne? length(unique())

```
[ ]: length(unique(statsminister))
```

1.27 Definer en vektor som heter vektor med seks tall du velger. Hva gjør koden vektor > 5?

```
[ ]: vektor <- 1:6  
vektor_sjekk <- vektor > 5  
vektor_sjekk
```

1.28 Ofte kan det være lurt å gjøre om boolske verdier til tall. F.eks. 1*(vektor_sjekk). Hva forteller sum(1*vektor_sjekk) deg?

```
[ ]: 1*(vektor_sjekk)  
sum(1*(vektor_sjekk))
```

1.29 Sjekk om Vedum har vært statsminister ved å bruke %in%

```
[ ]: "Vedum" %in% vektor_sjekk
```

1.30 Tell antall statsministerperioder Gro Harlem Brundtland hadde ved sum(1*...)

```
[ ]: sum(1*(statsminister == "Brundtland"))
```

1.31 Sett sammen vektorene parti, kvinne og statsminister til en data.frame og gi denne et passende navn.

```
[ ]: parti <- c("AP", "H", "Ap", "H", "Ap", "Ap", "Krf", "Ap", "KrF", "Ap", "H",  
  ↪ "Ap")
```

```
kvinne <- c(TRUE, FALSE, TRUE, FALSE, TRUE, FALSE, FALSE, FALSE, FALSE, FALSE,
↪TRUE, FALSE)
```

```
[ ]: statsminister_kjonn <- data.frame(parti, kvinne, statsminister)
```

1.32 Print ut, sjekk hvordan ser ut (kan bruke View() også). Hvordan ser du hva slags objekttype de ulike kolonnene er?

```
[ ]: statsminister_kjonn
```

1.33 Hent ut kolonnene fra data framen ved hjelp av \$. Bruk funksjonene nrow() og ncol() for å telle antall rader og kolonner i datasettet.

```
[ ]: statsminister_kjonn$parti
statsminister_kjonn$kvinne
statsminister_kjonn$statsminister

nrow(statsminister_kjonn)
ncol(statsminister_kjonn)
```